

2 ej.
20



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

**CONTROL DE VENTAS E INVENTARIOS DE
UNA FARMACIA POR MEDIO DE UNA
MICROCOMPUTADORA**

TESIS PROFESIONAL

**QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACION**

P R E S E N T A N :

**ORLANDO ZALDIVAR ZAMORATEGUI
JOSE ALBERTO RAUL PARRA ESPADIN**

DIR. DR. ADOLFO GUZMAN ARENAS

MEXICO, D. F.

1986



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

I N D I C E

RESUMEN

CAPITULO

I. INTRODUCCION.	1
I.1 COMO SE SOLUCIONAN LOS PROBLEMAS ACTUALMENTE.	2
I.2 QUE PROBLEMAS RESUELVE.	13
II. METODO PROPUESTO.	17
II.1 INTRODUCCION PARA EL DESARROLLO DEL METODO PROPUESTO.	18
II.1.1 ETAPAS.	18
II.2 PANORAMA TEORICO PARA EL DESARROLLO DEL SISTEMA.	21
II.2.1 INVESTIGACION.	21
II.2.2 ANALISIS.	23
II.2.3 DISENO.	26
III. DESCRIPCION DE PROGRAMAS.	34
III.1 INTRODUCCION A LA DESCRIPCION DE PROGRAMAS.	35
III.1.1 SUBSISTEMA DE ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS.	35
III.1.2 SUBSISTEMA DE CONTROL DE VENTAS.	34
III.1.3 DESCRIPCION DEL CONTENIDO DE LOS ARCHIVOS DEL SISTEMA.	38

III.2	DESCRIPCION GENERAL DE LA DOCUMENTACION DE LOS PROGRAMAS.	42
III.2.1	DESCRIPCION DE LAS SECCIONES QUE INTEGRAN LA DOCUMENTACION DE LOS PROGRAMAS.	43
III.2.2	DOCUMENTACION DE LOS PROGRAMAS DEL SUBSISTEMA DE ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS.	47
III.2.2.1	DOCUMENTACION DEL MODULO PROGRAMA PRINCIPAL (SABADO).	48
III.2.2.2	DOCUMENTACION DEL MODULO SUGPED.	45
III.2.2.3	DOCUMENTACION DEL MODULO LISPED.	93
III.2.2.4	DOCUMENTACION DEL MODULO PEDXTEM.	103
III.2.2.5	DOCUMENTACION DEL MODULO CADUCAR.	113
III.2.2.6	DOCUMENTACION DEL MODULO BAJA.	136
III.2.2.7	DOCUMENTACION DEL MODULO ESCRIBE.	117
III.2.2.8	DOCUMENTACION DEL MODULO LISTAR.	167
III.2.2.9	DOCUMENTACION DEL MODULO ACTUAL.	102
III.2.2.10	DOCUMENTACION DEL MODULO CARGPED.	201
III.2.2.11	DOCUMENTACION DEL MODULO LEERAN.	211
III.2.3	DOCUMENTACION DE LOS PROGRAMAS DEL SUBSISTEMA DE CONTROL DE VENTAS.	220
III.2.3.1	DOCUMENTACION DEL MODULO PROGRAMA PRINCIPAL (PROY1)	221
III.2.3.2	DOCUMENTACION DEL MODULO DUPLICAR.	251
III.2.3.3	DOCUMENTACION DEL MODULO LEECOPIA.	245
III.2.3.4	DOCUMENTACION DEL MODULO FACTURAR.	279
III.2.3.5	DOCUMENTACION DEL MODULO CORTE.	311
III.2.3.6	DOCUMENTACION DEL MODULO REGRESAR.	332
III.2.3.7	DOCUMENTACION DEL MODULO TRANSFERIR.	346
III.2.3.8	DOCUMENTACION DEL MODULO ACTUALI7.	355

IV.	MANUAL DE OPERACION DEL SISTEMA.	378
IV.1	MANUAL DE OPERACION DEL SUBSISTEMA DE ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS.	379
IV.1.1	MANUAL DE OPERACION DEL MODULO SABADO.	380
IV.1.2	MANUAL DE OPERACION DEL MODULO SUGPED.	385
IV.1.3	MANUAL DE OPERACION DEL MODULO LISPED.	393
IV.1.4	MANUAL DE OPERACION DEL MODULO PEDXTEN.	396
IV.1.5	MANUAL DE OPERACION DEL MODULO CADUCA.	400
IV.1.6	MANUAL DE OPERACION DEL MODULO BAJA.	406
IV.1.7	MANUAL DE OPERACION DEL MODULO ESCRIBE.	410
IV.1.8	MANUAL DE OPERACION DEL MODULO LISTAR.	413
IV.1.9	MANUAL DE OPERACION DEL MODULO ACTUAL.	419
IV.1.10	MANUAL DE OPERACION DEL MODULO CARGPED.	423
IV.1.11	MANUAL DE OPERACION DEL MODULO LEERAN.	428
IV.2	MANUAL DE OPERACION DEL SUBSISTEMA DE CONTROL DE VENTAS.	433
IV.2.1	MANUAL DE OPERACION DEL MODULO PROY1.	434
IV.2.2	MANUAL DE OPERACION DEL MODULO DUPLICAR.	442
IV.2.3	MANUAL DE OPERACION DEL MODULO LEECOPIA.	449
IV.2.4	MANUAL DE OPERACION DEL MODULO FACTURAR.	456
IV.2.5	MANUAL DE OPERACION DEL MODULO CORTE.	470
IV.2.6	MANUAL DE OPERACION DEL MODULO REGRESAR Y TRANSFERIR.	477
IV.2.7	MANUAL DE OPERACION DEL MODULO ACTUALIZAR.	483

V. LISTADOS FUENTES DE LOS PROGRAMAS DEL SISTEMA. 492

V.1	LISTADOS FUENTES DE LOS PROGRAMAS DEL SUBSISTEMA DE ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS.	493
V.1.1	LISTADO FUENTE DEL PROGRAMA SABADO.	494
V.1.2	LISTADO FUENTE DEL PROGRAMA MOD1.	504
V.1.3	LISTADO FUENTE DEL PROGRAMA MOD2.	508
V.1.4	LISTADO FUENTE DEL PROGRAMA MOD3.	514
V.1.5	LISTADO FUENTE DEL PROGRAMA MOD4.	516
V.1.6	LISTADO FUENTE DEL PROGRAMA MOD5.	523
V.1.7	LISTADO FUENTE DEL PROGRAMA MOD6.	529
V.1.8	LISTADO FUENTE DEL PROGRAMA MOD7.	536
V.1.9	LISTADO FUENTE DEL PROGRAMA MOD8.	542
V.1.10	LISTADO FUENTE DEL PROGRAMA MOD9.	552
V.1.11	LISTADO FUENTE DEL PROGRAMA MODA.	556
V.1.12	LISTADO FUENTE DEL PROGRAMA MODB.	561
V.1.13	LISTADO FUENTE DEL PROGRAMA MODC.	562
V.1.14	LISTADO FUENTE DEL PROGRAMA MODD.	568
V.2	LISTADOS FUENTES DE LOS PROGRAMAS DEL SUBSISTEMA DE CONTROL DE VENTAS.	571
V.2.1	LISTADO FUENTE DEL PROGRAMA PROY1.	572
V.2.2	LISTADO FUENTE DEL PROGRAMA MODU1.	581
V.2.3	LISTADO FUENTE DEL PROGRAMA MODU2.	585
V.2.4	LISTADO FUENTE DEL PROGRAMA MODU3.	589
V.2.5	LISTADO FUENTE DEL PROGRAMA MODU4.	598
V.2.6	LISTADO FUENTE DEL PROGRAMA MODU5.	603
V.2.7	LISTADO FUENTE DEL PROGRAMA MODU6.	607
V.2.8	LISTADO FUENTE DEL PROGRAMA MODU7.	610

VI. DESCRIPCION DE EJEMPLOS GENERADOS POR LA EJECUCION DE LOS PROGRAMAS DEL SISTEMA.	617
VI.1 DESCRIPCION DE EJEMPLOS GENERADOS POR LA EJECUCION DE LOS PROGRAMAS DEL SUBSISTEMA DE ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS.	618
VI.2 DESCRIPCION DE EJEMPLOS GENERADOS POR LA EJECUCION DE LOS PROGRAMAS DEL SUBSISTEMA DE CONTROL DE VENTAS.	661
VII. CONCLUSIONES.	696
VII.1 CONCLUSIONES: SISTEMA Y METODOLOGIA.	697
VII.2 VENTAJAS Y DESVENTAJAS.	699
VII.3 MEJORAS.	701
VII.4 SUGERENCIAS PARA TRABAJOS FUTUROS.	701
VIII. BIBLIOGRAFIA.	702

RESUMEN

En la actualidad se ha venido dando un impulso cada vez más grande al Desarrollo de Sistemas, con la finalidad de llegar a automatizar, la mayor parte de los servicios con que contamos hoy en día.

Esta automatización se ha acelerado, debido en gran parte a que los actuales sistemas con que se cuenta, ya no satisfacen las necesidades que de ellas se demanda, en lo referente a la rapidez y veracidad en el manejo de información.

De esta necesidad que se tiene en la actualidad, de manejar la información, de una manera más rápida y eficiente, así como también la de disponer de ella en cualquier instante que se desee, nos ha hecho abordar este tema, considerando que en la actualidad tiene gran importancia este manejo de información, ya que para cualquier tipo de empresa o negocio, estos lentos manejos de información, le pueden significar retraso de proyectos, de inversiones, pérdida de confianza de sus clientes etc., que a la larga le podrían traer graves consecuencias, por falta de exactitud y rapidez en cuanto a resultados.

Por lo expuesto anteriormente, hemos pretendido que con este trabajo, se muestre la importancia y el beneficio que le puede representar para los dueños de negocios, los sistemas automatizados y su implementación en máquinas pequeñas (Micros), sobre todo con respecto a las ventajas que le pueden producir estos sistemas en comparación a los sistemas manuales con los que cuenta actualmente, mostrándole además, que, con estos primeros, obtendría la rapidez y la eficiencia que requiere en sus transacciones comerciales con sus clientes, ayudándole a obtener un mejor control sobre la información de su negocio. Debido a esto, se abordó un problema real y de uso práctico, que tocará todos aquellos problemas que se vienen presentando en los negocios medianos o pequeños, por la falta de exactitud y rapidez en el manejo de información. A su vez que se trata de solucionar estos problemas, se sigue una Metodología para el desarrollo e implementación de este Sistema, que se desarrolla en base a las necesidades del problema, tratando con esto de solucionar en gran medida los graves problemas que se presentan en la actualidad debido al poco cuidado que se tiene al analizar el Sistema antes de intentar desarrollarlo y que en ocasiones causa graves conflictos entre el usuario y el Analista de Sistemas

El propósito que se tiene en la elaboración del trabajo, es presentar el desarrollo de un Sistema desde su fase de investigación hasta la implementación del mismo, de una forma accesible y fácil de entender en cada uno de los pasos desarrollados, intentando con esto que el usuario pierda el miedo a las computadoras y que se sane la confianza que él no ha tenido debido a su poca relación directa con esta nueva tecnología, y mostrarle además con esta presentación, su fácil manejo y la gran utilidad que puede tener en su negocio.

El tema de este trabajo trata sobre el desarrollo de un sistema y su implementación en una Micro, denominado, 'CONTROL DE VENTAS E INVENTARIOS EN UNA FARMACIA, POR MEDIO DE UNA MICROCOMPUTADORA', este sistema fue realizado en base a las necesidades reales de algunas farmacias que fueron consultadas, sobre las técnicas o procedimientos que siguen para el manejo de su información.

El diseño del Sistema contempla el manejo de los siguientes movimientos que se llevan a cabo en una farmacia, por ejemplo: SUGERIR PEDIDOS, COMPRA DE ARTICULOS POR TEMPORADA, CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD, DAR DE BAJA ALGUN ARTICULO, DAR DE ALTA ALGUN ARTICULO, LISTAR INFORMACION, ACTUALIZAR INFORMACION, CARGA DE UNIDADES PEDIDAS, LISTA VENTAS POR TEMPORADA, LISTA DATOS DE PROVEEDORES, FACTURACION DE VENTAS PARA LOS CLIENTES Y CORTE DE CAJA DIARIO.

Como se mencionó líneas arriba, en el desarrollo de este Sistema, se consideran dos puntos fundamentales, para el éxito o fracaso de del mismo, los cuales son los siguientes; el primero se refiere a la facilidad de manejo que se le debe brindar al usuario y el segundo se refiere a la facilidad de mantenimiento que se le pueda dar al Sistema.

Esperamos que este trabajo sea de utilidad a trabajos posteriores en el desarrollo e implementación de un Sistema en una Micro, recordando que si las actividades que realiza el Sistema, se le presentan de una forma accesible al usuario, éste sabrá que puede contar con alguien, que tal vez en un determinado momento le ayude a resolver su problema de manejo de información, de una forma rápida, clara y exacta.

[1]

C A P I T U L O I

I N T R O D U C C I O N

I.1 COMO SE SOLUCIONAN LOS PROBLEMAS ACTUALMENTE.

Si el problema existe, debe buscarse una solución.

De hecho, cualquier area de las actividades productivas del país funcionan y encuentran una solución a sus problemas.

La industria farmacéutica no escapa al comentario anterior. Todas sus actividades están controladas por procesos que hasta ahora, han sido una opción positiva para obtener el mayor beneficio en el aspecto económico y funcional.

Enmarcándonos en el dominio de nuestro trabajo, procederemos a describir cómo se solucionan actualmente los problemas que se presentan en el funcionamiento de una farmacia.

Las actividades de una farmacia, se ven desde los siguientes aspectos:

A) INVENTARIOS.

Los medicamentos se encuentran colocados en una serie de estantes por orden alfabético, lo que facilita su localización y consulta.

La verificación de existencias de medicamentos se hace de manera directa.

Esta actividad se realiza en intervalos de tres meses como mínimo o de seis meses como máximo.

El encargado de la farmacia, acude personalmente y revisa, estante por estante, las existencias de cada medicamento. En su registro (una libreta, hojas sueltas, formatos comerciales, etc.) anota el nombre del producto y las unidades en existencia actuales. Esto, lo hace en cada uno de los estantes y en la bodega.

Para realizar esta actividad, el encargado invierte un promedio de 2 minutos por cada medicamento. Este tiempo, le permite verificar los siguientes datos: nombre de la medicina, unidades en existencia, fecha de caducidad y precio.

Si tomamos en cuenta que en una farmacia de tamaño regular existen de 3000 a 5000 medicamentos diferentes, el encargado necesita invertir de 4000 a 10000 minutos, que equivalen a 100 horas/hombre, para el primer caso. Si una Jornada de trabajo tiene 8 horas, entonces necesita invertir por lo menos 12 1/2 Jornadas laborales, para terminar con esta función.

Hasta aquí, parece que todo funciona normalmente.

Pero ¡oh desilusión!, cuando el encargado termina de hacer su trabajo (12 1/2 días después), todo ha cambiado, ya que, mientras él hacía la actualización del inventario, la farmacia seguía funcionando, prosiguieron sus ventas, llegaron nuevos pedidos, etc. Por lo tanto, la información que obtiene al final, no está actualizada y en consecuencia, no corresponde con la realidad.

En varios casos, el encargado toma en cuenta los nuevos pedidos, bajas, ventas, etc., pero no con la exactitud que debiera, ya que no se concentra en una sola actividad.

En otras farmacias, este trabajo lo realizan todos los empleados (6 personas por lo regular), en una Jornada de labores. Escogen un día en que la farmacia suspende todas sus actividades y no abre sus puertas al público. Se concretan única y exclusivamente a trabajar con el inventario.

En el segundo caso, aun cuando todo el personal se dedique a hacer la actualización del inventario, los resultados que se obtienen no son óptimos como puede suponerse.

En teoría, esto es lo que el encargado de la farmacia o todo el equipo debe hacer periódicamente, para verificar las existencias del inventario de mercancías.

Sin embargo, la realidad es otra.

El encargado, difícilmente cumplirá con su trabajo. Está comprobado, por los datos obtenidos del mismo personal que labora en las farmacias, que la actualización del inventario se realiza en un 70 % dejando en la incertidumbre el 30 % restante.

Es fácil de comprobar estos argumentos.

Si el encargado realmente cumpliera con su función, la farmacia siempre tendría lo que el cliente común y corriente solicitase.

La realidad es diferente.

Cuando el cliente acude a una farmacia a comprar medicamentos, no siempre encuentra lo que pide.

Es necesario aclarar que, el hecho de que no se tenga cierta medicina, no es del todo culpa del encargado. Aquí intervienen otros factores, por ejemplo: que el proveedor no surta a tiempo lo que se le pide, escasez de materias primas, problemas laborales en los laboratorios y distribuidores, etc.

Sin embargo, si el encargado ve a su farmacia como un negocio, debe buscar todas las alternativas para obtener la mayor sanancia y no dejar que factores ajenos a él mismo, sean determinantes en el funcionamiento de su establecimiento comercial.

Por lo tanto, el encargado no cumple con su función de actualizar el inventario, por las siguientes razones:

- Falta de atención y desinterés en su trabajo.
- Grandes volúmenes de información.
- Distracciones.
- Errores humanos.

Un cliente que se pierde por no tener lo que pedía.

Si se hace un cálculo muy conservador, se llega a la conclusión de que, el establecimiento pierde aproximadamente \$50,000.00 al día, por no tener lo que piden los clientes. Si a esto le agregamos el hecho de que, si usted como cliente, pide algún producto y no lo tienen, es posible que no se atreva a solicitar algún otro distinto, ya que inconscientemente piensa que no lo van a tener. Y un comentario más, se está perdiendo un futuro cliente, porque, esa persona que hoy pidió un producto y no se tuvo, difícilmente regresará, ya que primero acudirá a otra farmacia y, como último recurso, regresará a la nuestra.

Por otra parte, si el cliente pide algo y no se tiene, Usted puede ofrecerle algo parecido. Pero cómo lo va a hacer si no tiene la información necesaria para tal fin. Otro cliente que se pierde.

Es necesario aclarar que existe un Reglamento Sanitario que se antepone al último párrafo, es decir que el encargado de la farmacia no debe cambiar el medicamento solicitado por otro, a menos que esté totalmente de acuerdo el solicitante.

Analice y confirme.

Cuánto dinero se pierde por no tener una simple medicina que un cliente solicita. La cadena de pérdidas crece hasta llegar a límites realmente alarmantes.

Esto tiene significado para aquellas personas que realmente les interesa su negocio. Y a quien no.

B) PEDIDOS.

Para hacer los pedidos, el responsable de la farmacia generalmente realiza lo siguiente:

a) Basándose en la información proporcionada por los reportes de actualización de sus inventarios, solicita a su proveedor, los medicamentos que juzga convenientes.

La realización del pedido de cada medicina está determinado por el hecho de que las existencias han llegado al mínimo permitido. Es decir, si para la medicina X, el mínimo que debe existir es de 10 unidades, y existen actualmente 10 o menos de 10, entonces, se tiene que pedir esa medicina.

b) Si la existencia llegó a cero. Cuando se agotan las existencias de la medicina X, es el momento de solicitar la remesa.

c) Se lleva una relación de los medicamentos que pide el cliente y que no se tienen. Esta relación se incrementa en el momento que llega el cliente, pide la medicina X y se carece de ella. Se toma nota del nombre del medicamento y cuando llegue el momento, se solicita al proveedor.

d) A todas las medicinas se les coloca un trozo de papel en el cual se anota su nombre. Cuando se vende, se le quita el trozo de papel y se coloca en un recipiente. Llegado el momento, el encargado toma todos los papeletos e integra su relación de pedidos.

Cuando llega el momento de hacer el pedido, el encargado revisa la información que obtuvo por cualquiera de los métodos descritos anteriormente y procede a elaborarlo.

En base a su experiencia, determina el número de unidades que solicitará de cada medicamento; experiencia y basándose en los datos obtenidos.

Sin embargo, como quedó asentado líneas antes, la información que le proporciona su inventario no es del todo confiable, por lo que sus pedidos no serán los adecuados.

Además, para elaborar sus pedidos, no toma en cuenta, a menos que sea todo un profesional, las estadísticas que genera cada medicamento.

Estas estadísticas son con respecto a las ventas por temporada, por demanda, etc.

Por otra parte, al momento de hacer el pedido, no cuenta con los elementos suficientes para determinar cuál proveedor le conviene más, tomando como referencia por ejemplo, los últimos precios que le dieron cada uno de ellos.

Si tuviese la posibilidad de manejar los últimos precios que le dieron cada proveedor, entonces sería un parámetro muy importante para decir a cuál escoger.

Aquí se presentan varias opciones:

- Pedir todo a un proveedor.
- Pedir determinadas medicinas a cada proveedor.
- Elaborar óptimamente los pedidos, basándose en información relacionada con precios y estadísticas.

Si únicamente se cuenta con un proveedor, lo escrito anteriormente, carece de validez, ya que, no tiene alternativa de selección. O le compra a éste o no compra nada.

Sin embargo, los pedidos de nuevas remesas de mercancías deben ser elaborados de manera más eficiente. Un hecho muy importante, es el de adelantarse a las demandas de los clientes. Si se lleva una estadística de ventas por temporada, podemos saber de antemano, cuál va a ser la demanda de determinada medicina, durante alguna época del año. De esta forma en nuestra bodegas siempre tendremos lo que se va a vender.

C) CADUCIDAD.

La caducidad de las medicinas es uno de los problemas que más preocupan a los encargados de farmacias.

La actividad de los empleados de farmacias, está centrada en la atención directa al público.

Si tomamos como referencia, la información generada por la actualización de inventarios, entonces se determina cuáles son los medicamentos que están próximos a caducar.

En otros casos, el encargado verifica visualmente la fecha de caducidad de los productos de manera directa, acudiendo a los estantes y bodega.

Por lo general, los proveedores dan a las farmacias un plazo de 30 días, para recibir las medicinas próximas a caducar para poder cambiarlas. Así, para que el proveedor reciba de la farmacia los medicamentos para cambiarlos, estos deben tener una fecha de caducidad 30 días posterior a la fecha de devolución. En ocasiones reciben hasta con 15 días de anticipación a la fecha de caducidad. En casos muy raros, mediante algún acuerdo muy personal, los proveedores reciben medicamentos que ya caducaron.

Pero dentro de lo establecido, si el plazo es menor, entonces el proveedor se niega a recibir los medicamentos.

Esto se traduce en una PERDIDA para el negocio de la farmacia.

Para evitar este problema, el encargado revisa su información obtenida.

Hasta aquí, todo parece bien.

Pero, como ya quedó asentado, la confiabilidad de los informes generados por la actualización del inventario es de 70 %, qué sucede con el 30 % restante.

Suponiendo que del 30 % restante, una mínima parte está en periodo de caducidad, digamos un 10 %, lo que significa que este porcentaje de medicamentos fuera de control, está por caducar.

Las pérdidas generadas por medicinas que caducaron y que pasaron inadvertidas para el encargado, llegan a sumar miles de pesos y es dinero que se va a la basura.

Es común ver en las bodegas de las farmacias, cajas que contienen medicamentos caducos y que aparte de ser un estorbo, representan un peligro, ya que por descuido o con toda mala intención, pueden ser vendidos a cualquier persona y si ésta no tiene el cuidado de ver la fecha de caducidad, es muy probable que tenga un problema muy serio de salud, incluso la muerte.

Dejemos a un lado el aspecto cliente, las pérdidas para la farmacia suman miles de pesos.

Veamos algunos aspectos más sutiles al respecto.

En la farmacia, existen medicamentos de poca demanda y que por lo general son los de precio más elevado. Estos, son los más propensos a generar pérdidas por caducidad.

Además, no se lleva un control estricto de las fechas de caducidad. De hecho son muy pocas las farmacias que controlan este parámetro. Por lo que el único control que se tiene es el visual.

Las pérdidas por caducidad, no se pueden permitir en un negocio que puede controlar ésta y otras variables, las cuales generan graves problemas por el descuido de los encargados.

D) FACTURACION.

La facturación que se realiza es muy simple, pero no óptima. Se elabora una nota por cliente, de la siguiente forma:

Llega el cliente y pide la medicina X, ya sea por medio de una receta o de manera oral. El encargado, va y busca lo pedido. Regresa al mostrador y después de dar el precio al cliente, éste decide si se la lleva o no. En caso afirmativo, el encargado procede a elaborar la nota, escribiendo en la misma, el nombre de la medicina, número de unidades, el precio, el impuesto y el importe total.

En varias farmacias, el producto tiene una contraseña que es separada y guardada en una caja, con el fin de llevar un control de las ventas.

En otras, por medio de la caja registradora, se elabora un "ticket". El original es para el cliente y la copia es para la farmacia, con el fin de llevar un registro de ventas.

Una vez elaborada la nota o factura, el cliente pasa ahí mismo, en el caso de farmacias pequeñas, o en la caja registradora en farmacias medianas o grandes.

En las farmacias pequeñas, esta acción no representa problemas. Para los establecimientos medianos o grandes, se debe tener un control más estricto sobre las ventas.

Por lo general en una farmacia grande, existen varias personas que atienden al público.

Una se encarga de surtir la receta de manera directa con el cliente, acudiendo a los estantes a buscar lo solicitado. Regresa, da el precio y procede a elaborar la factura.

Otra persona, está encargada de la caja. Verifica los totales y procede a elaborar el "ticket" para el cliente.

Una vez que el cliente pasó, se dirige a la sección de entrega de mercancía, la cual le es entregada, al momento de que otra persona, verifica que se lleva lo que ampara la nota y el ticket.

E) CORTES DE CAJA.

Los cortes de las ventas obtenidas durante una sesión, son elaborados por el encargado, basándose en los datos generados durante el día de trabajo.

Para el caso más sencillo, hace el corte tomando como referencia, los papелitos identificadores de medicinas vendidas. Hace un reporte, en el cual aparecen los siguientes datos:

Nombre de la medicina, unidades vendidas y total.

Al final, cuando se termina alguna caja de medicinas, hace su balance para encontrar la ganancia neta. El cálculo de impuestos también es inexacto, ya que no tiene los elementos suficientes para su realización.

Queda en la incertidumbre mucha información. Por ejemplo: Cuáles unidades vendidas corresponden a las existencias actuales; cuáles a las del último pedido que llegó; cuál es el último precio e impuesto; etc.

Es cierto, mucha de esta información la tiene, pero no a la mano, ni en el momento exacto. Si la tiene, pero no sabe donde, ya que maneja muchos papeles sin orden.

E) ACTUALIZACION DE INFORMACION.

En una farmacia, es muy común ver que los empleados se dedican a estar reetiquetando los productos.

El encargado hace dos actualizaciones: una en su libreta y otra en la medicina.

Esto se realiza muchas veces sin un control estricto, ya que en ocasiones, en la libreta del encargado existe un precio y en las medicinas hay otro.

F) SEGURIDAD DE LA INFORMACION.

Un punto muy importante que parece quedar muy a la deriva es el relacionado con la seguridad de la información.

El encargado, es el responsable de la seguridad de la información, haciéndola confidencial y exclusiva, ya que el conocimiento externo por parte de personas ajenas al negocio, podría traer graves consecuencias.

La libreta, la puede tomar cualquier empleado y hacer consultas con intenciones desconocidas, que se puede traducir en problemas posteriores.

El encargado nunca debe permitir que la seguridad de la información se escape de sus manos.

Analizando la información anterior, relacionada con el funcionamiento actual de las farmacias, no es alentador el panorama, ya que su manejo de información es deficiente y no existe seguridad en el mismo.

Por otra parte, se observa que tienen los datos pero distribuidos en diferentes papeles de manera desordenada. En ocasiones, los administradores prefieren hacer a un lado todos los reportes de información porque no les son útiles, debido al bajo índice de confiabilidad que presentan.

Por lo que, si un negocio no tiene a la mano toda la información necesaria y a tiempo para tomar decisiones, lo más probable es que sus ganancias no sean las adecuadas.

A fin de cuentas lo que interesa es el éxito de un negocio y éste se mide en función de las ganancias que genere. O acaso existe otro parámetro más importante ?

I.2 QUE PROBLEMAS RESUELVE.

Descrito el funcionamiento actual de las farmacias y visualizando los problemas a los cuales se enfrenta, por su pésimo control en el manejo de la información, consideramos que el desarrollo de este sistema de "CONTROL DE VENTAS E INVENTARIOS DE UNA FARMACIA POR MEDIO DE UNA MICROCOMPUTADORA", le resuelve y le facilita las siguientes actividades, que en un determinado momento se le pueden presentar.

Estas actividades son las siguientes:

A) INVENTARIOS.

En esta actividad el sistema le da la facilidad de conocer en el momento que así lo desee, qué tipo y qué cantidad de medicamentos tiene en la farmacia, además, el control sobre esta información es inmediato y en cuanto a resultados, ni se diga! ya que le ahorra tiempo, dinero y esfuerzo.

B) PEDIDOS.

En esta actividad el sistema, lleva un control exacto sobre el mínimo de unidades que deben existir por cada medicamento, este porcentaje de unidades en existencia lo puede fijar usted o si lo prefiere el sistema asume, que por cada medicamento deben existir arriba del 20%, del que había al iniciarse la venta de la remesa. En base a este parámetro, el sistema detecta qué medicamentos están próximos a requerir más unidades, y es entonces cuando le susiere que realice un pedido de ese medicamento, ya que de lo contrario se pueden quedar sin existencias.

Además cuenta con un proceso que tiene como fin, llevar una estadística de las ventas de los medicamentos por temporada, para que en base a estos datos pueda prever que compras de medicamentos se pueden hacer, para la temporada que se está consultando, y así con esto poder anticiparse a la posible demanda que se le pueda presentar en esa temporada.

c) CADUCIDAD.

En lo que respecta a esta actividad, el sistema le proporciona, tomando como referencia un número de días, cuales medicamentos van a caducar. El número de días lo marca el usuario.

Dentro de la información proporcionada, se encuentra: el nombre del medicamento, clave del proveedor que la surte, cuántos días faltan para que caduque, fecha de caducidad, etc.

Estos datos corresponden a las medicinas que existen en los estantes, así como de aquellas que están en la bodega, correspondientes a la última remesa surtida por el proveedor.

d) FACTURACION.

La facturación se realiza tomando los datos que existen en el archivo, por lo que se tiene la plena seguridad que todo lo que sale de la farmacia, será tomado en cuenta para protección del negocio.

Así, Usted no tiene que andar consultando precios e impuestos, viendo si aun hay unidades en existencia; todo esto lo hace el sistema.

Por otra parte se lleva un control sobre las ventas, actualizando en ese mismo instante la información de los medicamentos. Además, la facturación incluye, a otros artículos que no están dentro del sistema y que se consideran como adicionales.

Dentro de la facturación también se toman en cuenta aquellos productos similares al pedido, pero que no se tiene en existencia.

e) CORTE DE CAJA.

Tomando como base la nueva información generada durante una sesión de ventas, el sistema procede a elaborar el corte de caja, proporcionando un reporte con todos los aspectos que a un administrador de farmacia le interesa conocer por ejemplo, cuántas unidades se vendieron de cada medicina, cuánto se va a pagar de impuestos, cuál es la ganancia neta después de descontar el valor de adquisición del medicamento, así como los impuestos, etc.

f) ACTUALIZACION.

En esta actividad el sistema, le da la posibilidad de actualizar constantemente la información de los medicamentos, permitiéndole tener al día toda su información.

Además, le permite corregir errores, en caso de que se presenten, en la mayoría de los datos de cada medicamento.

g) SEGURIDAD.

Con respecto a la seguridad, se le proporcionan dos claves que solo el que conoce de éstas puede tener acceso al sistema para hacer actualizaciones en la información.

OTROS PROBLEMAS

Además de esta actividades, el sistema le proporciona una gran variedad de reportes con la información que necesita saber para tomar decisiones con respecto a los proveedores que lo surten, así como también todo lo referente a las estadísticas de las ventas de los medicamentos por temporada, etc.

C A P I T U L O I I

M E T O D O P R O P U E S T O

II.1 INTRODUCCION PARA EL DESARROLLO DEL METODO PROPUESTO.

En base al análisis del problema y observando que no existe un método que solucione todos los problemas, ya que sería algo másico, proponemos uno que ayuda a encontrar la solución óptima de cualquier proyecto para sistemas de información.

El método que proponemos está basado en teorías que se han desarrollado alrededor de los sistemas de información, pero que permite una evaluación permante para conocer el avance del trabajo.

Por otra parte, le muestran el camino, desde un punto de vista analítico, haciendo que tenga siempre presente todo aquello que le sirve para no tomar rumbos equivocados.

Concretamente el método propuesto es el siguiente:

II.1.1 ETAPAS.

A) INVESTIGACION.

Debe acudirse al lugar de los hechos para conocer el problema real, no platicado, ni leído ni mucho menos visto por los ojos de otra persona. Esto le permite hacer una definición precisa del problema, para establecer todos los parámetros que en él intervienen y así, hacer un estudio a fondo del mismo para poder concretar en objetivos claros y precisos.

Por lo tanto, durante esta etapa debemos estudiar y deducir el sistema de información que existe actualmente, no importa que éste sea manual o automático. Primero determinelo para después buscar sus fallas y poder señalar los puntos donde es posible hacer el cambio a un sistema automático.

Al final de esta etapa debe llegar a definir perfectamente el sistema actual de información.

Consulte el capítulo correspondiente a las Soluciones Actuales.

B) ANALISIS.

La recopilación y el análisis de los datos es parte fundamental. Cómo quiere atacar algo, sino sabe de que se trata.

Establecer el mayor número de parámetros y variables que intervienen en el proceso.

Elaborar el diagrama de flujo de datos.

Consulte el resumen donde se especifica qué hace el sistema.

C) DISEÑO.

Ya encontró una alternativa de solución perfectamente bien estudiada, desarróllela.

Nuestro análisis y diseño son estructurados, siguiendo, hasta cierto límite, los lineamientos ya conocidos del análisis y diseño estructurado.

Elaborar el Diagrama Jerárquico.

Todo lo que pretende hacer escríbalo en pseudocódigo estructurado.

Consulte la documentación para cada módulo del sistema.

D) DESARROLLO.

Tomando como base la investigación, el análisis y el diseño, pase el pseudocódigo a un lenguaje de alto nivel con el cual trabajará la computadora.

Consulte los listados de programas.

E) IMPLANTACION.

La implantación debe hacerse en paralelo para que el usuario, no sea introducido a un mundo que quizá no entiende y que, a la larga puede resultarle cansado. Por lo tanto, una vez que el sistema está totalmente terminado, poco a poco se van cambiando los procedimientos manuales por los automatizados.

Consulte los manuales del usuario.

F) MANTENIMIENTO.

El mantenimiento debe ser preventivo; expansivo; en algunos casos correctivo y en los menos regenerativo.

Nuestro sistema está hecho para recibir otros módulos que complementen sus actividades.

II.2 PANORAMA TEORICO PARA EL DESARROLLO DEL SISTEMA.

II.2.1. INVESTIGACION.

DEFINICION DEL PROBLEMA.

El primer Paso que debe darse para la solución de un problema es hacer su definición precisa. Nunca se debe suponer que el problema es obvio y que todos saben de antemano en qué consiste. Deseche esta idea. Es necesario invertir tiempo y esfuerzo para dejar por escrito, la definición del problema. Este tiempo y esfuerzo, le ahorrará mucho más de lo que usted se imagina, ya que es posible, en caso de omitir esta etapa, que al final, cuando crea que todo está hecho, se dé cuenta que mucho de lo que tiene no sirve, y lo que es peor, que tendrá que iniciar todo otra vez, con las consiguientes pérdidas. Por lo tanto, primero debemos definir el QUE del problema para, posteriormente, especificar el COMO se va a resolver. El documento que hace esta definición/descripción, del problema, debe estar escrito en un lenguaje sencillo, para posteriormente elaborar otro tocando los aspectos técnicos del mismo. Entre las condiciones mínimas que debe tener este documento están: debe ser completo, preciso, claro y formal. A éstas pueden agregarse: tener la posibilidad de ser modificado y mantenible.

Una parte fundamental de esta investigación directa de campo, consiste en elaborar una gráfica del sistema de información que existe actualmente, ya sea manual, semi o automatizado. Qué sentido tiene hacer lo anterior ?

Antes de automatizar cualquier proceso, debemos introducirnos en él para conocer sus secuencias y funciones, con el fin de poder encontrar los puntos específicos, en los cuales es posible el cambio de manual a automático.

Este estudio, permite identificar los siguientes puntos del proceso:

- Elementos de entrada.
- Elementos operacionales del proceso.
- Elementos de salida.

ESPECIFICACION DE OBJETIVOS.

Tomando como base las observaciones hechas durante las entrevistas con los usuarios, en este caso los clientes, se plantean una serie de objetivos para satisfacer óptimamente todas las necesidades reales que se presentan. Iniciemos el proyecto entendiendo qué es lo que realmente quiere el cliente, para poder definir el problema y establecer objetivos. Los objetivos deben estar orientados hacia la satisfacción de las necesidades del usuario. Por lo tanto deben ser claros y precisos. La claridad radica en no dejar dudas o ambigüedades en su contenido y significado, tanto para el equipo de desarrollo como para cualquier persona que tenga algo que ver con el proyecto, como usuarios, clientes, analistas, diseñadores, ingenieros. Son precisos porque especifican todas y cada una de las acciones que realizarán, así como sus limitaciones y condiciones.

De esta manera los objetivos están íntimamente relacionados con las necesidades del usuario. Así, antes de iniciar cualquier actividad, debe saberse perfectamente qué es lo que quiere satisfacer el usuario, al automatizar sus actividades. Por lo tanto, los objetivos definen las funciones generales del sistema y además sirven de base fundamental para realizar la estructuración del mismo.

El problema debe ser estudiado a fondo para llegar a especificar los objetivos. Cuando están claros y precisos los objetivos, entonces se procede a establecer relaciones y jerarquías entre ellos. Algo que debe quedar muy claro es el hecho de que si el problema no está perfectamente definido, entonces los objetivos tampoco lo estarán.

Hasta este momento es cuando se establecen posibles métodos de solución, que se proponen para su análisis y estudio, con el fin de determinar y elegir el mejor; aquí se analiza el COMO se va a resolver el problema. Dentro del análisis se incluyen también a los objetivos.

Si el análisis es realizado tomando en cuenta los factores enunciados anteriormente, entonces se llegan a conocer las funciones propias del sistema y se determinan los métodos de solución.

II.2.2 ANALISIS.

QUE ES EL ANALISIS.

Análisis es el estudio de un problema, antes de tomar alguna acción. En el mundo de la computación, el análisis es el estudio orientado al desarrollo de nuevos sistemas. Primero debemos analizar para después implementar un sistema.

Al finalizar la etapa de análisis se debe tener un documento donde se especifique de manera precisa al sistema.

ANALISIS ESTRUCTURADO.

Hasta aquí, mucho de lo que se ha discutido corresponde a la fase de análisis clásica, sus problemas y defectos. El análisis estructurado considera lo siguiente:

- Nuevos objetivos para el análisis.

Viendo sobre los problemas de reconocimiento y sus fallas en la fase de análisis, es necesario hacer las siguientes adiciones a nuestro conjunto de objetivos de la fase de análisis:

El producto del análisis debe ser mantenible.

El problema del tamaño debe ser atendido usando un método de partición efectivo. En este caso se propone la modulación como el mejor método. Las gráficas deben usarse donde sea posible.

Debe existir una diferencia entre consideraciones físicas y lógicas; distribución de responsabilidades, basada en esta diferenciación, entre el analista y el usuario.

Tenemos que construir un modelo de sistema lógico de manera que el usuario, pueda estar familiarizado con las características del sistema antes de la implementación.

HERRAMIENTAS DEL ANALISIS ESTRUCTURADO.

El análisis estructurado hace uso de las siguientes herramientas para llegar a sus objetivos:

- Diagrama de flujo de datos. Instrumento de modelación para mostrar a los sistemas como una red integrada por varios subsistemas, relacionados por flujos de datos. Además indican dónde se usa la información y dónde se le transforma.

- Pseudocódigo estructurado. Este pseudocódigo está basado en la programación estructurada, la cual hace uso de las estructuras básicas de control. El pseudocódigo consiste en escribir la lógica de la solución en forma narrativa y de acuerdo con ciertas reglas.

- Diccionario de Datos. Para mantener el orden y comunicación y además que evalúe las interfases. Debe ser entendible y mostrar en detalle todo lo que haga el sistema.

ANALISIS ESTRUCTURADO. DEFINICION.

El análisis estructurado es el uso de algunas de estas herramientas:

Diagrama de Flujo de Datos.

Diccionario de Datos.

Inglés Estructurado.

Tablas de Decisión.

Arboles de Decisión.

Estas herramientas sirven para construir una nueva clase de especificación estructurada.

Aunque la construcción de la especificación estructurada es el aspecto más importante del Análisis Estructurado, existen otros puntos de menor importancia:

- Estimaciones heurísticas.
- Métodos que facilitan la transición del análisis al diseño.
- Ayudas para aceptar pruebas de generación.
- Técnicas de recorrido del principio al final.

QUE NO ES EL ANALISIS ESTRUCTURADO.

Existen varios aspectos en los cuales el Análisis Estructurado no tiene aplicaciones directas. Estos son algunos:

- Análisis costo-beneficio.
- Análisis de factibilidad.
- Administración de proyectos.
- Análisis de desarrollo.
- Selección de equipo.
- Consideraciones personales.
- Políticas.

El analista es el responsable de la comunicación entre todos los usuarios.

Así, el analista es la principal liga entre el área del usuario y la obra de implementación. El tiene que comunicar los requerimientos a los implementadores y los detalles de cómo esos requerimientos son satisfechos para el usuario. Participa en la determinación actual de que hacer; imagina la mejor manera de solucionar el problema y debe estar al día en los aspectos tecnológicos. Además participa en la implementación. Por lo tanto asume el rol de un arquitecto que guía la construcción del edificio.

II.2.3 DISEÑO.

A) DISEÑO ESTRUCTURADO.

El diseño estructurado usa una sistematización metódica y altamente estructurada para desarrollar el diseño de sistemas de información, estableciendo primero las funciones generales y sus objetivos, hasta llegar al nivel de especificaciones en detalle.

La primera etapa del diseño de sistemas consiste en establecer una relación entre las funciones propias del sistema y los programas de computadora. Así cada actividad que se desarrolle para satisfacer las necesidades del usuario, se hace por medio de uno o varios programas de computadora.

La arquitectura de un sistema se realiza a partir del nivel más alto de abstracción, en el cual se concibe al sistema de manera global, compuesto por subsistemas.

La arquitectura del sistema debe contener la definición de lo siguiente:

-- Tipos de Información. La información es : de entrada, salida y archivos de datos. La información de entrada es externa, capturada por medio de dispositivos periféricos tales como terminales de video o impresoras. Otro tipo de información es la de salida, como producto del sistema y ésta puede ser en forma de reportes, listados, etc. Finalmente, los archivos de datos, son los que contienen la información obtenida del exterior, generada y usada por el sistema y almacenada por él mismo.

-- Programas. En esta etapa del diseño se enuncian de manera general, los programas que integran el sistema, especificando su nombre, objetivo y actividades, relaciones, interfases y tipo de información que manejan. Esto se puede hacer por medio de un Diagrama de Estructura o Diagrama de Flujo de Datos.

B) DIAGRAMA JERARQUICO

Ahora procederemos a establecer la estructura Jerárquica de cada programa por medio de un Diagrama Jerárquico. El Diagrama está compuesto por una serie de elementos llamados módulos. Módulo es un componente o porción del sistema, perfectamente identificado en cualquier nivel Jerárquico y con una función específica. Este proyecto se realizó valiéndose de la modularidad, la cual se usa para subdividir cualquier trabajo en secciones. La modularidad permite tener un panorama del todo por sus partes, proporcionado mayor comprensión y control sobre las mismas, integrándolas en un todo funcional y Jerárquico.

El Diagrama Jerárquico nos permite ver que cada sistema está compuesto por una serie de subsistemas que a su vez contienen programas completos integrados por módulos Jerárquicos, así como las relaciones que existen entre cada uno de ellos, mostrando cuáles son los activadores y cuáles los activados.

Existen varios criterios para elaborar técnicas que permiten modular un programa, es decir, subdividirlo en módulos. A continuación se enuncian:

1. Acoplamiento. Es la medida del grado de interdependencia entre los módulos en un diseño estructurado; el total de información compartida entre dos módulos. En el diseño deben buscarse módulos independientes, con baja interdependencia, para poder usarlos en otros programas sin hacer cambios en su codificación. Ahora bien, no solo deben tomarse los módulos como tales, sino que también deben tomarse en cuenta los programas "manejadores" que controlan la activación de los módulos. Aquí radica gran parte del éxito o fracaso del acoplamiento. Sin embargo, en la vida práctica, muchas veces resulta casi imposible lograr que los módulos sean totalmente independientes.

Tipos de acoplamiento óptimos:

- Acoplamiento por datos. Si la comunicación entre módulos es por medio de datos.

- Acoplamiento por estampilla. Si la comunicación modular es por medio de registros o arreslos.

- Acoplamiento por control. Si la comunicación modular es por una bandera.

2. **Cohesión.** Es la medida del grado de asociación de los elementos dentro de un módulo. Se consideran elementos de un módulo a las instrucciones (individuales o en grupo), activación, invocación o llamadas a otros módulos. Lo que se busca es que el módulo tenga robustez, solidez, coherencia o cohesión, para tener más independencia y menos acoplamiento con otros módulos.

Tipos de cohesión óptimos:

- **Cohesión funcional.** Todos los elementos del módulo participan en una sola tarea.

- **Cohesión secuencial.** Cuando los datos de salida de un módulo sirven como datos de entrada en otro.

- **Cohesión comunicacional.** Cuando los elementos del módulo realizan funciones diferentes, pero tienen en común los parámetros de entrada y salida.

Sumados a estos criterios explicados anteriormente, existen otros de menor importancia como son:

- **Tamaño.** Es el número de líneas de código ejecutable. Sin embargo, no es apropiado usar este criterio como determinante, ya que no es representativo.

- **Funciones generales.** Diseñar módulos con posibles usos a futuro. No son necesarios; concrétese a diseñar lo necesario y olvídense de lo superfluo o de lo que posiblemente se use.

A fin de cuentas lo que se busca es que los módulos presenten el mínimo acoplamiento para poder trabajar, que cada uno de ellos tenga alta cohesión para una función específica y en consecuencia, sean independientes y factibles de ser reubicados en otras aplicaciones.

En realidad no existe un patrón que indique hasta que punto debe subdividirse un sistema. Los únicos parámetros o criterios que consideramos vitales son:

a) **Claridad.** El objetivo y la serie de instrucciones del módulo no deben ser confusos ni ambiguos, sino que deben dejar bien claro su significado.

b) **Funcionalidad.** El módulo debe cumplir con los requerimientos especificados. El módulo debe hacer exactamente lo que se espera de él y estar libre de errores.

c) **Documentación.** La calidad de la documentación debe permitir fácil mantenimiento.

C) DESCRIPCION DE PROGRAMAS.

PROGRAMACION ESTRUCTURADA.

Esta sección sirve para especificar a detalle cada uno de los módulos que intervienen en los sistemas y subsistemas.

En este trabajo usamos la técnica de la Programación Estructurada para detallar los módulos que intervienen en el sistema.

Explicaremos brevemente en qué consiste la Programación Estructurada.

Hace algunos años, se observó que la mayor parte de la gente que se dedica al diseño de sistemas, no tenía un enfoque sistémico del problema, sino que, concretamente en la fase de codificación, cada uno de ellos se valía de su experiencia, sin usar algún método específico. Esto ocasionaba que solo el que hacía los programas le entendía a su novela y nadie más. De esta forma, para localizar errores, actualizar programas, hacer agrasados, etc., era punto menos que imposible. Esto, incrementaba notablemente los costos de mantenimiento.

Sumado a lo anterior, los Programadores no aplicaban alguna metodología para el desarrollo de los sistemas, lo que traía como consecuencia, productos de mala calidad.

A Edsger W. Dijkstra se le dá el crédito como padre de la Programación Estructurada, quien la define como una programación sin so to's, usando el diseño de arriba-abajo.

La Programación Estructurada o Sistémica es una metodología que permite la subdivisión de problemas para establecer una estructura Jerárquica de fácil manejo.

No pierda de vista que la programación estructurada en este trabajo, forma parte de una serie de procesos que deben seguirse para la creación de todo un sistema.

La programación estructurada genera módulos con una sola entrada, una salida; éstos son de fácil traducción a lenguaje de alto nivel; en consecuencia su revisión es fácil.

La Programación Estructurada está basada en el uso de tres estructuras fundamentales o lógicas de control.

- SECUENCIA. Estructura que enuncia una a continuación de otra, un conjunto de instrucciones ejecutantes o imperativas.

Instrucción 1

Instrucción 2

Instrucción 3

.

.

Instrucción N

Significado: Ejecuta una instrucción a continuación de la otra.

- DECISION. Estructura procesal que, por medio de la evaluación de un predicado o condición, determina alternativas para la ejecución de instrucciones. Por su grado de complejidad, existen tres tipos de decisión: simple, binaria o múltiple.

- DECISION SIMPLE. Tiene la siguiente forma:

SI (condición) ENTONCES

Pseudocódigo

FIN (SI)

Significado: Si la condición es verdadera entonces se ejecuta el pseudocódigo, el cual está compuesto por una o varias instrucciones. En caso contrario no ejecuta nada. Por lo tanto toma una sola decisión. Continúa con la secuencia.

- DECISION BINARIA. Tiene la siguiente forma:

SI (condición) ENTONCES

pseudocódigo 1

SINO

pseudocódigo 2

FIN (SI)

Significado: Si la condición es verdadera se ejecuta el pseudocódigo 1; en caso contrario se ejecuta el pseudocódigo 2. Por lo tanto presenta únicamente dos alternativas y solo se ejecuta una de ellas.

- DECISION MULTIPLE. Tiene dos formas. La primera es la siguiente:

SI (condición 1) ENTONCES

pseudocódigo 1

SINO SI (condición 2) ENTONCES

pseudocódigo 2

SINO SI (condición 3) ENTONCES

pseudocódigo 3

·
·

SINO

pseudocódigo N (opcional)

FIN (SI)

Significado: Si la condición 1 es verdadera se ejecuta el pseudocódigo 1, sino evalúa la condición 2 y si es verdadera ejecuta el pseudocódigo 2, y así sucesivamente. Si ninguna de las condiciones es verdadera entonces ejecuta el pseudocódigo N (si es que existe). Es una estructura de decisión múltiple, cuyas condiciones deben ser mutuamente excluyentes.

Segunda forma:

```

EN CASO QUE variable
sea igual a (valor o caso 1) : pseudocódigo 1
    o
sea igual a (valor o caso 2) : pseudocódigo 2
    o
sea igual a (valor o caso 3) : pseudocódigo 3
    o
    .
    .
    .
SINO : pseudocódigo N (opcional)
FIN (CASO)
    
```

Significado: Si el valor o caso de la variable es igual a uno de los especificados entre paréntesis, se ejecuta únicamente el pseudocódigo asociado. Si ninguno de los valores o casos concuerda con el valor de la variable, entonces se ejecuta el pseudocódigo N, si es que existe.

- ITERATIVAS O DE REPETICION. Estructura de control que permite la ejecución iterativa o repetidamente de un pseudocódigo, bajo determinadas condiciones. Existen varios tipos, de acuerdo con la forma de la condición ejecutante.

- REPITE-HASTA. Tiene la forma:

```

REPITE
    pseudocódigo
HASTA (condición)
    
```

Significado: Esta estructura de control ejecuta el pseudocódigo que se encuentra dentro de su dominio o alcance, hasta que la condición sea verdadera. Es decir mientras la condición sea falsa, se ejecuta el pseudocódigo. Tenga la plena seguridad que la ejecución del pseudocódigo se ejecuta al menos una vez, ya que primero ejecuta y después pregunta. Se debe manipular a la condición, en alguna instrucción del dominio, para que cambie de valor.

- MIENTRAS. Tiene la forma:

MIENTRAS (condición)

pseudocódigo

FIN (MIENTRAS)

Significado: Esta estructura ejecuta el pseudocódigo que se encuentra dentro de su alcance, en tanto o mientras la condición sea verdadera. Primero se evalúa la condición, si es verdadera, se ejecuta el pseudocódigo. Si es falsa no se ejecuta más. Es posible que nunca se realice el pseudocódigo, si antes de entrar a esta estructura, la condición tiene el valor de falso. En alguna instrucción del pseudocódigo se debe manipular la condición para que ésta cambie de valor, ya que de lo contrario, si al principio, la condición es verdadera, por siempre se estarán ejecutando las instrucciones. Este punto también se aplica a la estructura REPITE-HASTA. Aquí primero se evalúa la condición para determinar la ejecución del pseudocódigo. Por lo tanto, las instrucciones se ejecutan repetidamente, mientras la condición sea verdadera.

- REPETIR N VECES. Tiene la forma.

HAZ desde que variable = (EXP. Arit. 1 hasta EXP. Arit. 2)

pseudocódigo

FIN (HAZ)

Significado: Esta estructura de control iterativa, permite la ejecución del pseudocódigo N veces, donde N está determinado por el rango entre la Expresión Aritmética 1 y la Expresión Aritmética 2. Así, la variable primero toma el valor inicial de Exp. Arit. 1, y lo incrementa o decrementa, según sea el caso, hasta llegar al valor determinado por la Exp. Arit. 2. Mientras la variable no rebasa el valor de la Exp. Arit. 2, ejecuta el pseudocódigo.

C A P I T U L O I I I

D E S C R I P C I O N D E P R O G R A M A S

III.1 INTRODUCCION A LA DESCRIPCION DE PROGRAMAS.

Esta descripción de programas se llevará a cabo tomando en cuenta, que el Sistema consta de dos Subsistemas, los cuales a continuación serán definidos y descritos, Junto con las actividades que realiza cada uno.

III.1.1 SUBSISTEMA DE ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS:

- SUGERIR PEDIDOS
- LISTAR PEDIDOS
- SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
- CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A CADUCAR
- DAR DE BAJA ALGUN ARTICULO
- DAR DE ALTA ALGUN ARTICULO
- LISTAR LA INFORMACION
- ACTUALIZAR INFORMACION
- CARGA DE UNIDADES PEDIDAS
- LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
- SALIRSE DEL SISTEMA

III.1.2 SUBSISTEMA DE CONTROL DE VENTAS:

- DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO
- LEER EL ARCHIVO DE TRABAJO DEL DIA
- FACTURACION PARA CLIENTES
- CORTE DE CAJA DE LAS VENTAS DEL DIA
- CORRECCIONES AL ARCHIVO DE TRABAJO
- REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
- SALIRSE

Enunciados los dos Subsistemas de los que consta este Sistema, a continuación se presenta la descripción detallada de la documentación de cada Programa, y los manuales correspondientes.

Antes de comenzar esta descripción de Programas, especificaremos primeramente cada uno de los campos que forman los archivos de este Sistema, para que tenga una mejor visión más adelante, acerca de la información que guarda cada uno de los archivos del Sistema.

III.1.3 DESCRIPCION DEL CONTENIDO DE LOS ARCHIVOS DEL SISTEMA.

NOMBRE DEL ARCHIVO : MAESTRO

NOMBRE EN EL PROGRAMA : BF

NOMBRE EN EL DISCO : MAESTRO.FIL

NOMBRE	DESCRIPCION DEL CAMPO
CLAVE	Número de clave del medicamento.
MS1	Número de clave del medicamento similar 1.
MS2	Número de clave del medicamento similar 2.
MS3	Número de clave del medicamento similar 3.
FRECIO	Precio de venta del medicamento.
UNID	Número de unidades en existencia.
ANN	Año en que caducan las unidades en existencia.
MES	Mes en que caducan las unidades en existencia.
DIA	Día en que caducan las unidades en existencia.
CP	Clave del proveedor principal del medicamento.
IMP	Impuesto de venta del medicamento.
PDU	Porcentaje mínimo de unidades en existencia que
UVEN	Número de unidades vendidas.
UPED	Número de unidades pedidas.
UVP1	Número de unidades vendidas de la nueva remesa.
AA	Año en que caducan las unidades pedidas.
MM	Mes en que caducan las unidades pedidas.
DD	Día en que caducan las unidades pedidas.

NOMBRE DEL ARCHIVO : TRABAJO

NOMBRE EN EL PROGRAMA : VA

NOMBRE EN EL DISCO : TRABAJO.FIL

NOMBRE	DESCRIPCION DEL CAMPO
CLAVE	Número de clave del medicamento.
MEDS1	Número de clave del medicamento similar 1.
MEDS2	Número de clave del medicamento similar 2.
MEDS3	Número de clave del medicamento similar 3.
PRECIO	Precio de venta del medicamento.
UNID	Número de unidades en existencia.
IMP	Impuesto de venta del medicamento.
UVEN	Número de unidades vendidas.
UPE1	Número de unidades en existencia del último pedido.
UVPD	Número de unidades vendidas del último pedido.

NOMBRE DEL ARCHIVO : NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

NOMBRE EN EL PROGRAMA : NM

NOMBRE EN EN DISCO : NOMYEST.FIL

NOMBRE	DESCRIPCION DEL CAMPO
CLAVE	Número de clave del medicamento.
NOMBRE	Nombre del medicamento.
R1	Número de unidades vendidas en la temporada 1
R2	Número de unidades vendidas en la temporada 2.
R3	Número de unidades vendidas en la temporada 3.
R4	Número de unidades vendidas en la temporada 4.
PROV1	Número de clave del proveedor principal.
PRE1	Precio de compra que da el proveedor principal.
DP1	Descuento que hace el proveedor principal.
PROV2	Número de clave del segundo proveedor.
PRE2	Precio de compra que da el segundo proveedor.
DP2	Descuento que hace el segundo proveedor.
PROV3	Número de clave del tercer proveedor.
PRE3	Precio de compra que da le tercer proveedor.
DP3	Descuento que hace el tercer proveedor.
DD	Día en que se cargó el campo de unidades en existencia.
MM	Mes en que se cargó el campo de unidades en existencia.
AA	Año en que se cargó el campo de unidades en existencia.

NOMBRE DEL ARCHIVO : PEDIDOS

NOMBRE EN EL PROGRAMA : PD

NOMBRE EN DISCO : PEDIDOS.FIL

NOMBRE	DESCRIPCION DE LOS CAMPOS
CLAVE	Número de clave del medicamento.
UR	Número de unidades solicitadas en el pedido.
CP	Clave del proveedor al que se le hace el pedido.
AP	Apuntador para ordenar la salida de los pedidos.

NOMBRE DEL ARCHIVO : CLAVES DISPONIBLES

NOMBRE EN EL PROGRAMA : NC

NOMBRE EN EL DISCO : CVEDISP.FIL

NOMBRE	DESCRIPCION DE LOS CAMPOS
CDISP	Número de clave que se encuentra disponible.

III.2 D E S C R I P C I O N G E N E R A L
D E L A D O C U M E N T A C I O N
D E L O S P R O G R A M A S

III.2. 1 DESCRIPCION DE LAS SECCIONES QUE INTEGRAN LA DOCUMENTACION DE LOS PROGRAMAS.

La documentación de cada uno de los módulos consta de las siguientes secciones:

- IDENTIFICACION.

Es el nombre del módulo, por medio del cual es reconocido dentro el sistema.

- PROGRAMA DONDE SE LOCALIZA.

Nombre del archivo donde están grabadas todas las instrucciones que forman parte del programa modular que integra al sistema.

1. OBJETIVO.

Especifica de manera clara para qué sirve este programa o que se obtiene por su uso. Indica brevemente las razones o propósitos de este módulo.

2. FUNCIONES PRINCIPALES.

Usando un lenguaje natural, se explica cada una de las funciones que ejecuta el programa, sin entrar en detalle.

3. NARRATIVA DEL MODULO.

Por medio de una sencilla narración, se explican las actividades que realiza el módulo de manera general.

4. ARCHIVOS QUE UTILIZA EL MODULO.

Esta sección enuncia el nombre de los archivos. Un archivo tiene dos nombres: Uno que le sirve de identificación en el sistema y otro que indica su identificación en el disco. Además debe incluir una descripción mínima del archivo. La descripción detallada se hace en la sección correspondiente.

5. VARIABLES.

Se enuncian y especifican todas las variables que aparecen en el programa, las cuales identifican a los elementos de datos. Estas variables aparecen en la sección de declaración de variables del programa principal y en los procedimientos. Incluye nombre y descripción tipificada.

5.1 VARIABLES EXTERNAS.

Sección donde se enuncian y especifican las variables externas, es decir son aquellas que aparecen en el programa manejador y que usa el módulo (overlay) del cual se está haciendo la documentación.

5.2 VARIABLES MODULARES.

Son las variables que se usan únicamente en el módulo overlay y que son globales al mismo.

5.3 VARIABLES LOCALES.

Variables que tiene alcance exclusivamente dentro del procedimiento donde se declaran.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO.

Se enuncian y describen todos los procedimientos y funciones, los cuales son un conjunto de instrucciones que definen maneras específicas de cómo se comporta cada componente y sus relaciones con los otros miembros, que forman parte del módulo.

7. DIAGRAMA JERARQUICO.

Gráfica en la cual se ilustra los procesos del módulo, así como sus interfaces con los componentes del mismo.

8. DIAGRAMA DE FLUJO DE DATOS.

Gráfica para ilustrar el flujo y transformación de los datos a medida que son procesados por el módulo.

9. PSEUDOCODIGO.

Es la narración detallada del método aplicado para la solución, sujeta a ciertas reglas, permitiendo leer de principio a fin todo el proceso. Se usa un lenguaje casi natural.

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO.

Descripción particular y a detalle de cada módulo.

10.1 NOMBRE DEL PROCEDIMIENTO.

Identificación del procedimiento dentro del módulo.

PROGRAMA DONDE SE LOCALIZA.

Nombre del programa grabado en disco, que contiene el procedimiento.

OBJETIVO.

Indica brevemente las razones o propósitos de este procedimiento.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO.

Identificación de los archivos, tanto dentro del programa, como en disco. Incluye breve descripción de los mismos.

VARIABLES.

Es similar a la sección anterior correspondiente, solo que aquí es a nivel de procedimientos dentro de un módulo.

VARIABLES EXTERNAS.

Identificación de variables y breve descripción.

VARIABLES MODULARES.

Identificación de variables y breve descripción.

VARIABLES LOCALES.

Identificación de variables y breve descripción.

PROCEDIMIENTOS QUE INVOCA.

Procedimientos propios del módulo o externos a él. Incluye identificación y objetivo.

PSEUDOCODIGO.

Es la narración detallada del método aplicado para la solución, sujeta a ciertas reglas, permitiendo leer de principio a fin todo el proceso. Se usa un lenguaje casi natural. Está en el nivel de procedimientos como miembros de un programa principal.

III.2.2 D O C U M E N T A C I O N
D E L O S P R O G R A M A S
D E L
S U B S I S T E M A
D E
A C T U A L I Z A C I O N Y C O N S U L T A
D E R E Q U E R I M I E N T O S

III.2.2.1 DOCUMENTACION DEL MODULO: PROGRAMA PRINCIPAL

PROGRAMA DONDE SE LOCALIZA: SABADO

1. OBJETIVO: Controlar el proceso de ejecución del Subsistema de Actualización y Consulta de Requerimientos.

2. FUNCIONES PRINCIPALES:

- Valida la fecha para entrar al Subsistema de Actualización y Consulta de Requerimientos.
- Despliega por pantalla el menú de las funciones del Subsistema de Actualización y Consulta de Requerimientos.
- Solicita la función a ejecutar.

3. NARRATIVA DEL PROGRAMA:

Solicita y valida la fecha de entrada al Subsistema de Actualización y Consulta de Requerimientos, después despliega por pantalla el menú de las funciones que se pueden ejecutar dentro del Subsistema, y solicita la función a ejecutar.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO
NH	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
II	Apuntador para acceder el Archivo de Nombres y Estadísticas del Medicamento, variable de tipo entero.
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.
J	Apuntador para acceder el Archivo de Trabajo, variable de tipo entero.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
BAND	Bandera para detectar el fin del proceso, variable de tipo caracter.
SN	Para respuesta de si ó no, variable de tipo caracter.
BAND1	Para detectar algún error al transmitir la letra de la función a ejecutar, variable de tipo caracter.

5.3 VARIABLES LOCALES: Ninguna

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
FECHA	Valida fecha de entrada al Subsistema.
LISTMENU	Despliega el menú de las funciones que se pueden ejecutar dentro del Subsistema de Actualización y Consulta de Requerimientos.
SUGPED	Sugerir pedidos de los medicamentos, a los cuales sus unidades en existencia están abajo del 20 %.
LISPED	Listar la información del Archivo de Pedidos, de los pedidos que se han solicitado comprar.
PEDXTEM	Sugerir compra de medicamentos por temporada.
CADUCA	Listar los medicamentos que cumplan con la fecha de caducidad transmitida.
BAJA	Dar de baja del Sistema algún medicamento que no sea costeable.
ESCRIBE	Escribir los datos del medicamento que se desea dar de alta dentro del Sistema.
LISTAR	Listar la información más relevante de los medicamentos que se encuentran dentro de los Archivos Maestro y de Nombres y Estadísticas del Medicamento.
ACTUAL	Actualizar la información de los medicamentos.
CARGPED	Cargar al campo de unidades pedidas del Archivo Maestro (BF) la nueva cantidad de unidades pedidas.
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Dar por terminada la sesión.

DIAGRAMA JERARQUICO

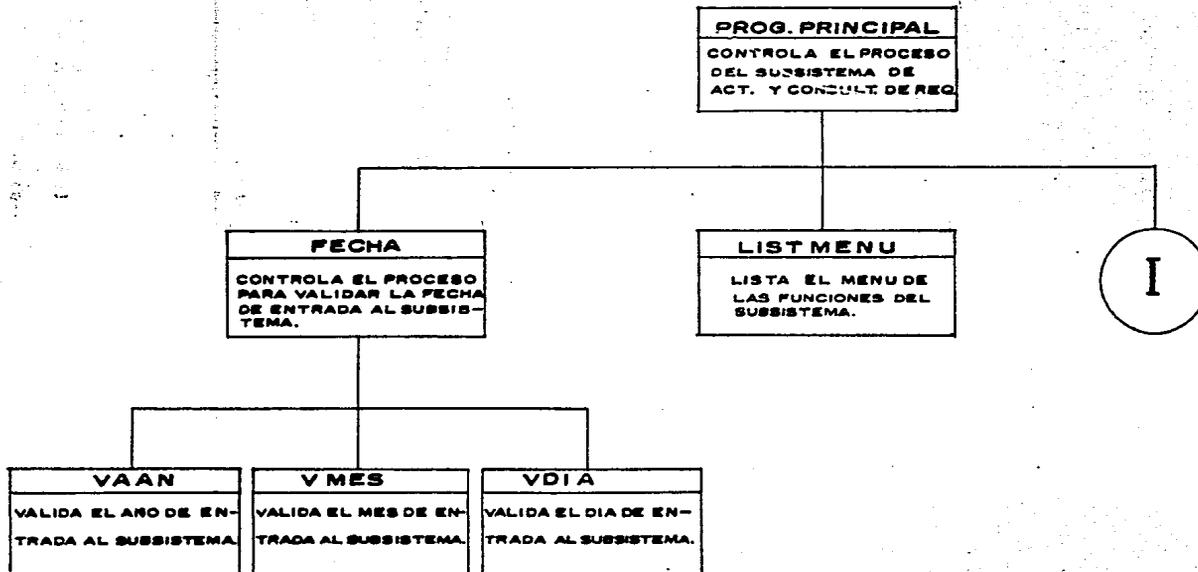
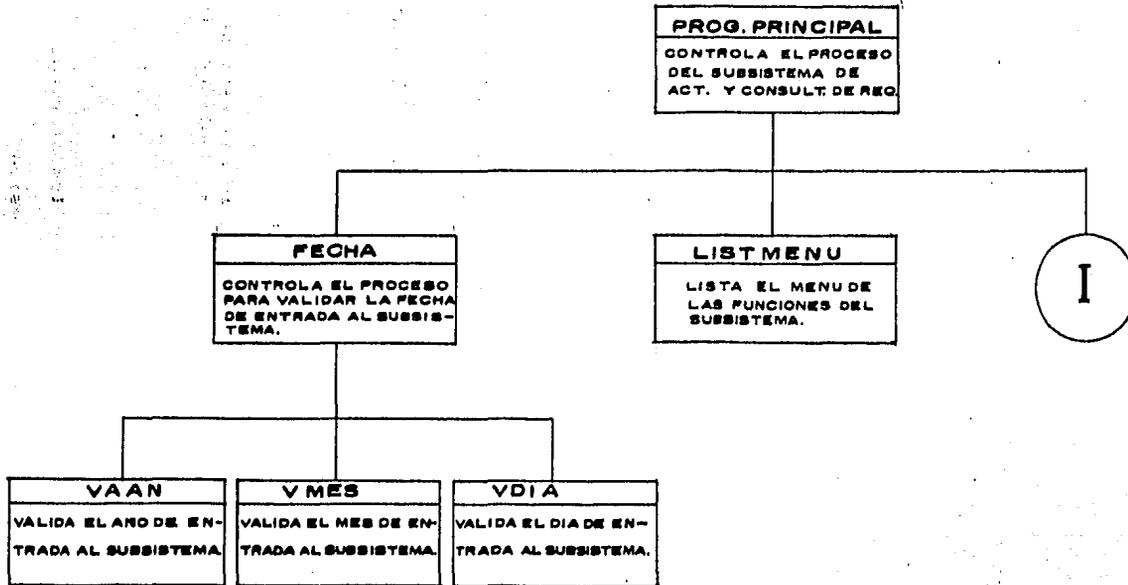


DIAGRAMA JERARQUICO



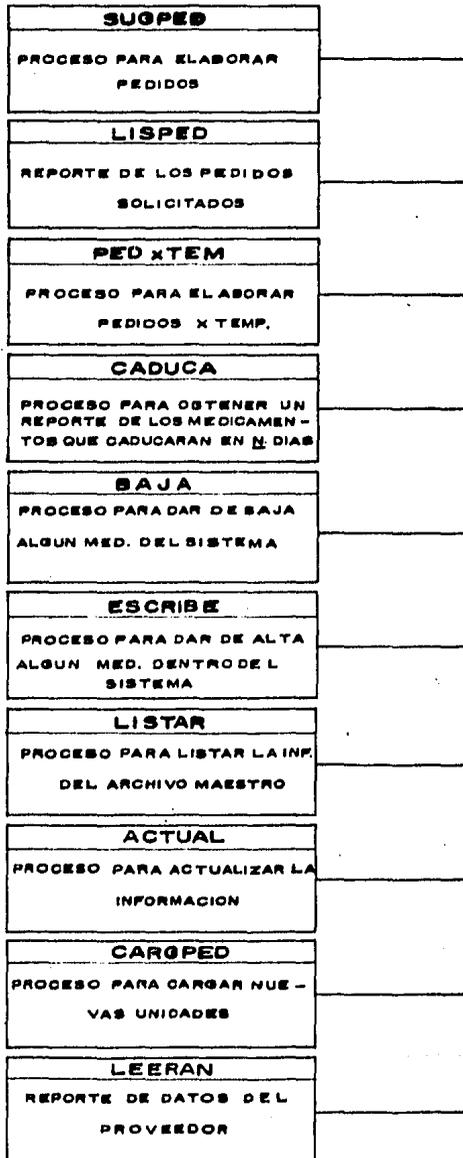
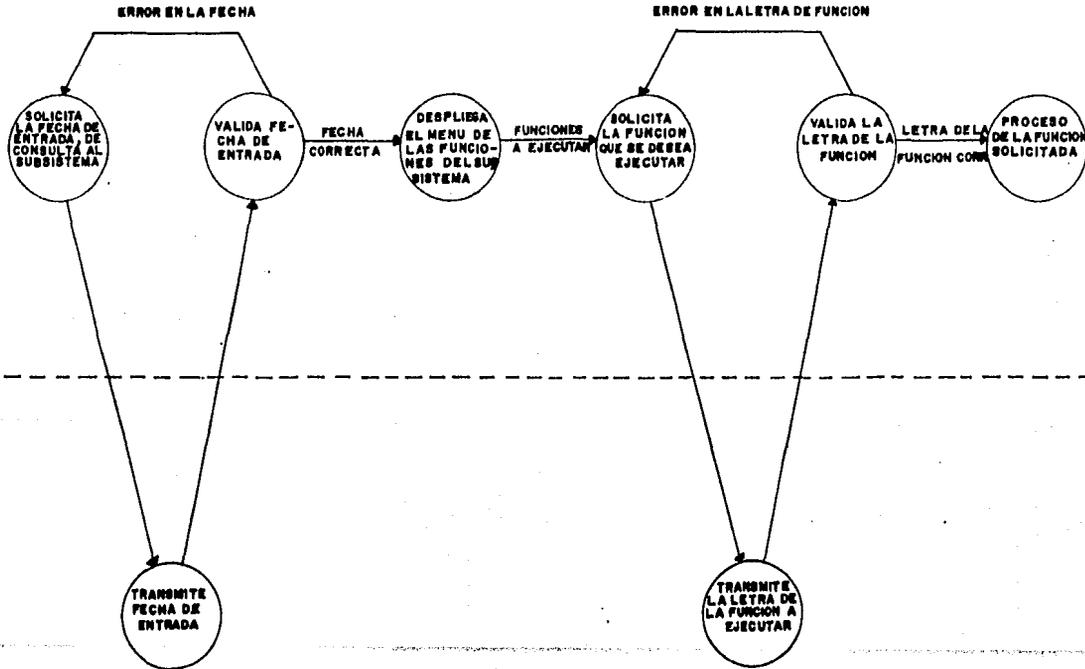


DIAGRAMA JERARQUICO

DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

E 52 J

USUARIO

9. PSEUDOCODIGO

(* Inicia Programa Principal *)

INICIA (PROGRAMA PRINCIPAL)

```

Asigna Archivo Maestro
Asigna Archivo de Trabajo
Asigna Archivo de Nombres y Estadísticas del Medicamento
I <-- 0
Lee Archivo Maestro
Valida lectura
Lee Archivo de Nombres y Estadísticas del Medicamento
Activa FECHA
(* Se asigna a la variable LIM el número de la clave máxima que *)
(* contiene información dentro del Sistema *)
LIM <-- BF.MS1
BAND
Repite
  Despliega pregunta
  'DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => '
Lee dato transmitido
Si SN es igual a 'S' entonces
  inicia (Si)
  Repite
    BAND1 <-- ' '
    Despliega pregunta
    'DEME LA FUNCION QUE DESEA => '
    Lee dato
    En caso que OPC1
      sea igual a 'A'      : Activa SUGPED
      ó
      sea igual a 'B'      : Activa LISPED
      ó
      sea igual a 'C'      : Activa PEDXTEM
      ó
      sea igual a 'D'      : Activa CADUCA
      ó
      sea igual a 'E'      : Activa BAJA
      ó
      sea igual a 'F'      : Activa ESCRIBE

```

```

        ó
sea igual a 'G'      : Activa LISTAR
        ó
sea igual a 'H'      : Activa ACTUAL
        ó
sea igual a 'I'      : Activa CARGPED
        ó
sea igual a 'J'      : Activa LEERAN
        ó
sea igual a 'K'      : BAND <-- '*'
        ó
sea diferente de 'A,B,C,D,E,F,G,
                  H,I,J,K' : Despliega mensaje de error
                               y vuelve a solicitar el dato
fin (caso)

    Hasta que BAND1 sea igual a
fin (Si)
Sino
    BAND <-- '*'
Hasta que BAND sea igual a '*'
Despliega mensaje de terminación de programa
Cierra archivos

FIN ( PROGRAMA PRINCIPAL )

```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: LISTMENU

PROGRAMA DONDE SE LOCALIZA: SABADO

OBJETIVO: Desplegar el menú de las funciones que se pueden ejecutar dentro del Subsistema de Actualización y Consulta de Requerimientos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES. Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento de LISTMENU *)

INICIA (LISTMENU)

Despliega por pantalla el elemento de las funciones que se pueden ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos

FIN (LISTMENU)

10.2 NOMBRE DEL PROCEDIMIENTO: FECHA

PROGRAMA DONDE SE LOCALIZA: MODA

OBJETIVO: Valida fecha de entrada al Subsistema.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
II	Apuntador para acceder el Archivo Maestro, variable de tipo entero.
ANIO	Año del día en que se está consultando al Subsistema, variable de tipo entero.
MES	Mes del día en que se está consultando al Subsistema, variable de tipo entero.
DIA	Día en que se está consultando al Subsistema, variable de tipo entero.
ERROR	Variable por medio de la cual se verifican las lecturas al Archivo Maestro, variable de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
DIFAA	Diferencia de años, entre el año de la última consulta y el de la actual consulta, variable de tipo entero.
DIFMM	Diferencia de meses, entre el mes de la última consulta y el de la actual consulta, variable de tipo entero.
DIFDD	Diferencia de días, entre el día de la última consulta y el de la actual consulta, variable de tipo entero.
DIFTA	Tipo de año (bisiesto o no) en que se está haciendo la consulta al Subsistema, variable de tipo entero.
BANDZ	Para detectar algún error al transmitir la fecha, variable de tipo caracter.

VARIABLES LOCALES: Ninguna

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
VAAN	Valida año de entrada al Subsistema.
VMES	Valida el mes de entrada al Subsistema.
VDIA	Valida el día de entrada al Subsistema.

PSEUDOCODIGO

(* Inicia Procedimiento FECHA *)

INICIA (FECHA)

Activa VAAN
Activa VMES
Activa VDIA

FIN (FECHA)

10.2.1 NOMBRE DEL PROCEDIMIENTO: VAAN

PROGRAMA DONDE SE LOCALIZA: MODA

OBJETIVO: Validar el dato del año de entrada al Subsistema.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ANIO	Año del día en que se está consultando al Subsistema, variable de tipo caracter.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
BANDZ	Para detectar algún error en la transmisión de la fecha, variable de tipo caracter.
DIFAA	Diferencia de años, entre el año de la última consulta y el de la actual consulta, variable de tipo entero.

VARIABLES LOCALES: Ninguna

PROCEDIMEINTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento VAAN *)

INICIA (VAAN)

Despliega encabezado de bienvenida al Subsistema de Actualización y Consulta de Requerimientos

Repite

BANDZ <-- ' '

Despliega mensaje

'DEME AÑO => '

Lee dato

(* Se obtiene diferencia de años *)

DIFAA <-- NM.AA

Si DIFAA es menor que 0 o mayor que 1 entonces

inicia (Si)

Despliega mensaje de error

BNADZ <-- '*'

fin (Si)

Sino

Si DIFAA es igual a 1 entonces

Si NM.MM es menor que 12 entonces

inicia (Si)

Despliega mensaje de error

BANDZ <-- '*'

fin (Si)

Hasta que BANDZ sea igual a ' '

FIN (VAAN)

10.2.2 NOMBRE DEL PROCEDIMIENTO: VNES

PROGRAMA DONDE SE LOCALIZA: MODA

OBJETIVO: Validar el dato del mes de entrada al Subsistema.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
MES	Mes del día en que se está consultando al Subsistema, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
BANDZ	Para detectar algún error en la transmisión de la fecha, variable de tipo caracter.
DIFAA	Diferencia de años, entre el año de la última consulta y el de la actual consulta, variable de tipo entero.
DIFMM	Diferencia de meses, entre el mes de la última consulta y el de la actual consulta, variable de tipo entero.

VARIABLES LOCALES: Ninsuna

PROCEDIMIENTOS QUE INVOCA: Ninsuno

PSEUDOCODIGO

(* Inicia procedimiento VMES *)

INICIA (VMES)

Repite

BANDZ <-- ' '

Despliega mensaje

'DEME MES => ' -

Lee dato

Si MES es mayor que 0 y menor o igual a 12 entonces

inicia (Si)

DIFMM <-- MES - NM.MM

Si DIFAA es igual a 0 entonces

Si DIFMM es menor que 0 ó mayor que 2 entonces

inicia (Si)

Despliega mensaje de error

BANDZ <-- '*'

fin (Si)

Sino

imprime linea de espacios en blanco

Sino

Si DIFMM es mayor ó igual a -9 entonces

inicia (Si)

Despliega mensaje de error en el dato transmitido

BANDZ <-- '*'

fin (Si)

Sino

Despliega mensaje de que el mes transmitido no existe

Hasta que BANDZ sea igual a ' '

FIN (VMES)

10.2.3 NOMBRE DEL PROCEDIMIENTO: VDIA

PROGRAMA DONDE SE LOCALIZA: MODA

OBJETIVO: Valida el dato de entrada, del día en que entro a consultar al Subsistema.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ANIO	Año del día en que se está consultando al Subsistema, variable de tipo entero.
MES	Mes del día en que se está consultando al Subsistema, variable de tipo entero.
DIA	Día en que se está consultando al Subsistema, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
BANDZ	Para detectar algún error en la transmisión de la fecha, variable de tipo caracter.
DIFTA	Tipo de año (Bisiesto o no) en que se está consultando al Subsistema, variable de tipo entero.
DIFAA	Diferencia de años, entre el año de la última consulta y el de la actual consulta, variable de tipo entero.
DIFMM	Diferencia de meses, entre el mes de la última consulta y el de la actual consulta, variable de tipo entero.
DIFDD	Diferencia de días, entre el día de la última consulta y el de la actual consulta, variable de tipo entero.

VARIABLES LOCALES: Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia Procedimiento VDIA *)

INICIA (VDIA)

Repite

Repite

BANDZ <-- ' '

Despliega mensaje

'DEME DIA => '

Lee dato

DIFTA <-- ANIO entre 4

Si DIA es menor que 1 entonces

BANDZ <-- '*'

Sino

inicia (Sino)

En caso que MES

```

sea igual a 1,3,5,7,8,10,12      : Si DIA es mayor que 31 entonces
                                BANDZ <-- '*'

```

ó

```

sea igual a 2                    : Si DIFTA es igual a 0 entonces
                                Si DIA es mayor que 29 entonces
                                BNADZ <-- '*'
                                Sino
                                imprime linea de espacios
                                Sino
                                Si DIA es mayor que 28 entonces
                                BANDZ <-- '*'

```

ó

```

sea igual a 4,6,9,11            : Si DIA es mayor que 30 entonces
                                BANDZ <-- '*'

```

fin (Caso)

fin (Sino)

Si BANDZ es igual a '*' entonces

Despliega mensaje de error de que el día transmitido no existe

Hasta que BANDZ sea igual a ' '

DIFDD <-- DIA - NM.DD

Si DIFAA es igual a 0 entonces

Si DIFMM es igual a 0 entonces

Si DIFDD es menor que 0 entonces

Despliega mensaje de error

Hasta que BANDZ sea igual a ' '

(* Asigna los datos de la fecha a los campos correspondientes *)

(* del registro cero en el archivo de Nom. y Est. de Med. *)

NM.DD <-- DIA

NM.MM <-- MES

NM.AA <-- ANIO

Graba la fecha en el archivo de Nom. y Est. del Med.

Valida escritura

FIN (VDIA)

III.2.2.2 DOCUMENTACION DEL MODULO: SUGPED

NOMBRE EN EL MENU: SUGERIR PEDIDOS

PROGRAMA DONDE SE LOCALIZA: MOD2

1. OBJETIVO. Sugerir pedidos de los medicamentos, a los cuales sus unidades en existencia están abajo del 20 % .

2. FUNCIONES PRINCIPALES:
 - Listar del archivo Maestro (BF) los medicamentos a los cuales sus unidades están abajo del 20 % .
 - Desplegar por pantalla los datos necesarios, para hacer un pedido, del medicamento listado.
 - Proporciona alternativas para dar los datos del medicamento que se desea pedir.
 - Graba los datos del medicamento en el archivo de Pedidos (PD).
 - Despliega por pantalla la Presunta de si desea conocer el monto de lo que lleva pedido.

3. NARRATIVA DEL MODULO:

Si el archivo Maestro (BF) no contiene información, el programa le desplegará un mensaje de error, indicándole que el archivo se encuentra vacío y que necesita activar el módulo ESCRIBE (opcion F), para dar de alta algún medicamento.

En caso que el archivo Maestro contenga información, el programa empezará a listar aquellos medicamentos a los cuales sus unidades en existencia están abajo del 20 %, desplegando por pantalla todos los datos necesarios para que el usuario tenga una mejor opción, a la hora de hacer un pedido del medicamento listado.

Si no desea hacer un pedido del medicamento listado, el programa le listará los datos del siguiente medicamento, al cual sus unidades en existencia están abajo del 20 %, en caso que desee hacer el pedido, el programa le proporcionará la información necesaria para que usted elija como desea proporcionar los datos del pedido. Una vez que se han transmitido todos los datos, se graban en el archivo de Pedidos (PD) e inmediatamente le aparece un mensaje, preguntándole si desea conocer el monto neto total de lo que lleva pedido. Después el proceso se repite hasta el final del archivo Maestro o hasta que ya no existan más medicamentos con unidades abajo del 20 %.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
PD	PEDIDOS.FIL	ARCHIVO DE PEDIDOS
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- DE LOS MEDICAMENTOS

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
II	Apuntador para acceder el archivo de Pedidos, variable de tipo entero.
LIM	Número de la clave máxima que contiene información en el Sistema, variable de tipo entero.
II	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento, variable de tipo entero.
SUMUNI	Suma de unidades en existencia más unidades vendidas, variable de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo Maestro, es de tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
SPP	Suma del pago neto total de lo que lleva pedido, variable de tipo real.
ACVE	Arreslo de claves de proveedores, tipo entero.
ACVEA	Arreslo de claves auxiliares de proveedores, tipo entero.

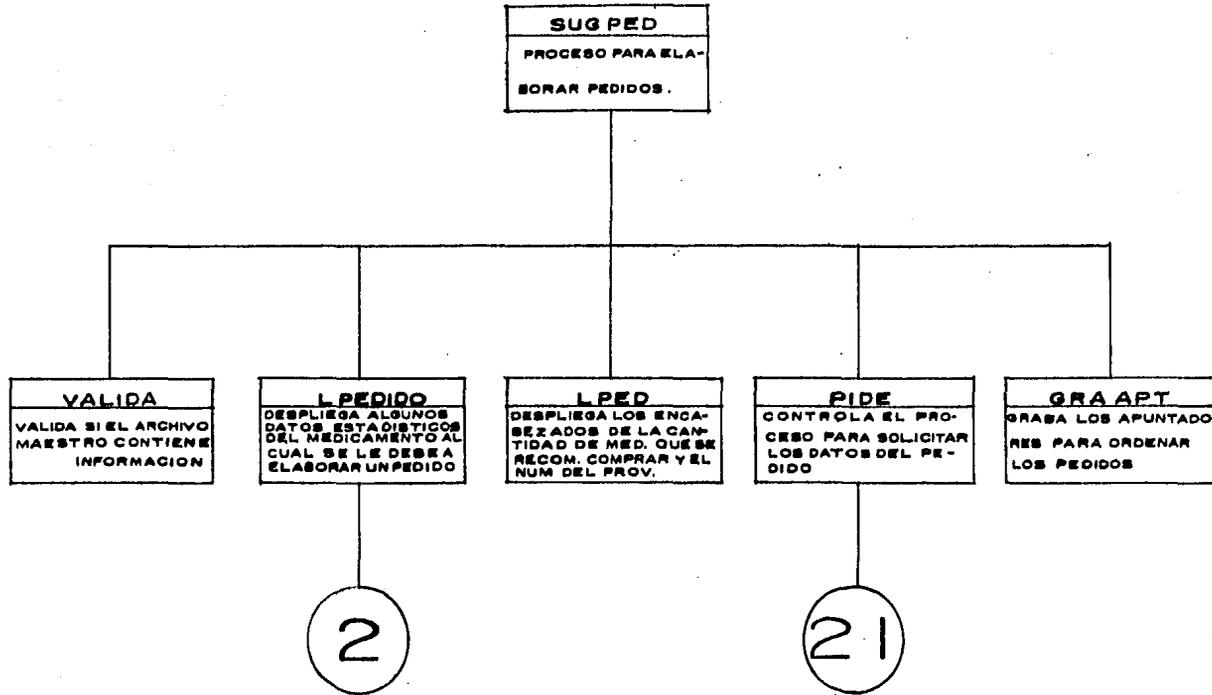
5.3 VARIABLES LOCALES.

NOMBRE	DESCRIPCION
EZ	Indice para limpiar el arreslo de claves de proveedores, variable de tipo entero.
PMUE	Porcentaje mínimo de unidades que debe haber en existencia, variable de tipo real.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO :

NOMBRE	OBJETIVO
VALIDA	Valida la variable que indica, si el archivo Maestro contiene información.
LPEDIDO	Despliega por pantalla información de proveedores.
LPED	Despliega por pantalla información del medicamento.
PIDE	Controla el proceso para elaborar pedidos.
GRAAPT	Graba el apuntador en el archivo de Pedidos, el cual indica el orden en el que deben aparecer las claves de los medicamentos, en el momento de ser listados.

DIAGRAMA JERARQUICO

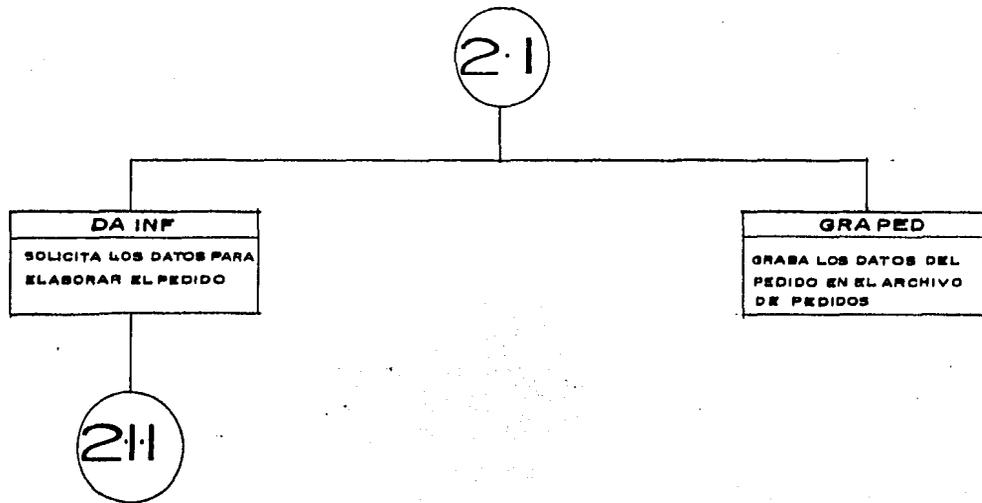


U 88 1

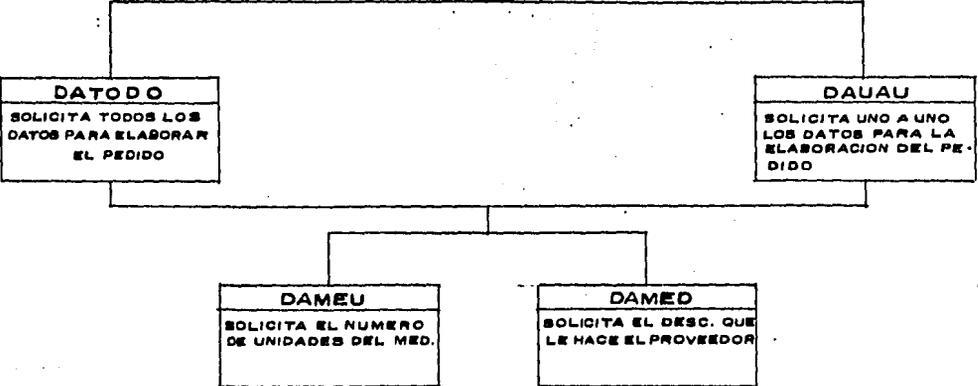
2

NUM DIA
CALCULA EL NUMERO DE DIAS QUE LLEVA DE VENTA UNA REMESA DE MEDICAMENTOS

C 69.1 3

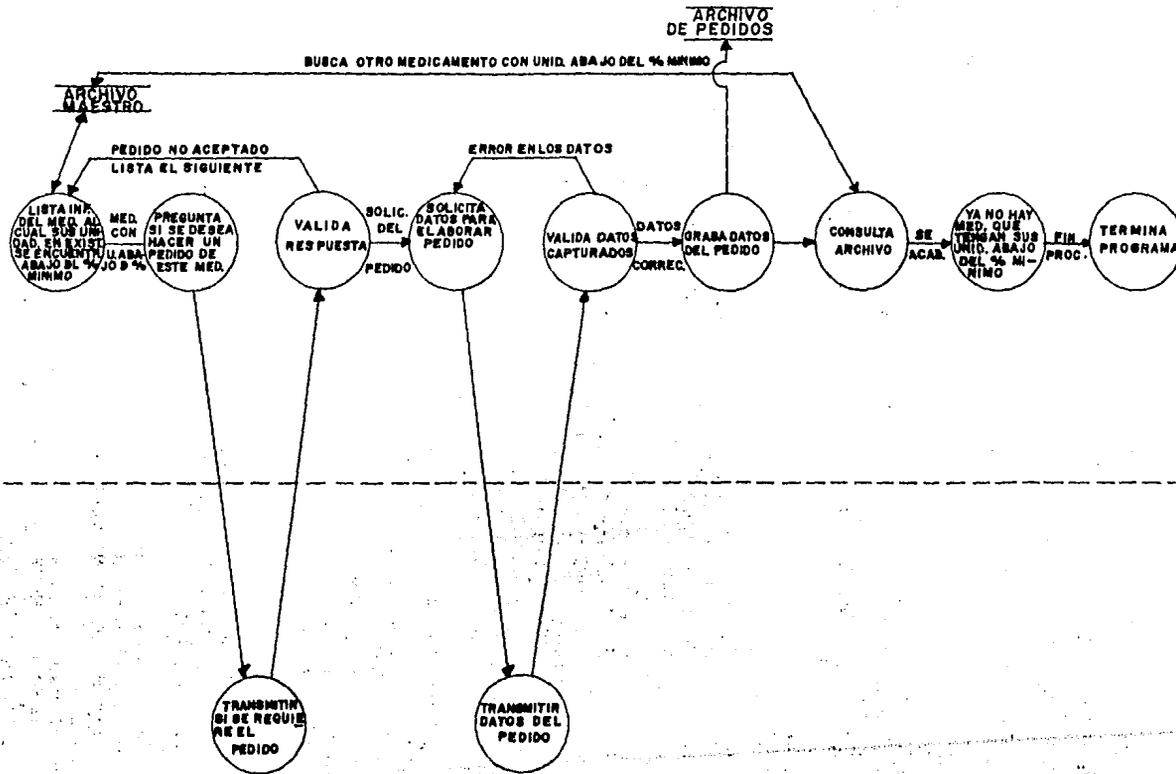


2.1.1



E 69.3 7

DIAGRAMA DE FLUJO DE DATOS



PROGRAMA E 70 3

USUARIO

9. PSEUDOCODIGO.

```
(* Inicia el modulo SUGPED *)
```

```
INICIA (SUGPED)
```

```
Asigna el archivo de pedidos PD.FIL
```

```
(* Inicializa variables *)
```

```
I <-- 1
```

```
I1 <-- 0
```

```
SPP <-- 0
```

```
(* Inicializa los arreglos de las claves de los proveedores *)
```

```
(* con ceros *)
```

```
Haz desde IL=1 hasta 10 lo siguiente
```

```
  inicia(Haz)
```

```
    ACVE(IL) <--- 0
```

```
    ACVEA(IL) <--- 0
```

```
  fin(Haz)
```

```
Activa VALIDA
```

```
(* Inicia proceso para recorrer el archivo Maestro, con el fin *)
```

```
(* de localizar los medicamentos a los cuales sus unidades en *)
```

```
(* existencias se encuentran abajo del 20 % *)
```

```
Mientras I es menor ó igual a LIM haz lo siguiente
```

```
  inicia(Mientras)
```

```
    Lee el archivo Maestro y valida la lectura
```

```
    (* Del archivo Maestro suma unid. en exist. mas unid. vend. *)
```

```
    SUMUNI <-- BF .UNID + BF .UVEN
```

```
    (* Se obtiene el 20 % de las unid. sumadas *)
```

```
    PMUE <--- SUM * Porcentaje de unid. en exist.
```

```
    (* Proceso para seleccionar aquellos medicamentos, que tengan *)
```

```
    (* sus unidades en existencias abajo del 20 % *)
```

```
    Si (PMUE > unid. en exist. ) Y ( unid. ped. = 0 ) entonces
```

```
      inicia(Si)
```

```
        (* Proceso para hacer pedidos *)
```

```
        II <-- I
```

```
        Lee archivo de Nombres y Estadísticas de Medicamentos
```

```
        Valida lectura
```

```
        Activa LPEDIDO
```

```
        Activa LPED
```

```
        Activa PIDE
```

```
      fin(Si)
```

```
    I <-- I + 1
```

```
  fin(Mientras)
```

```
Si LIM es mayor que 0 entonces
```

```
  Si I1 es mayor que 0 entonces
```

```
    Activa GRAAPT
```

```
  Sino
```

```
    inicia(Sino)
```

```
      Despliega mensaje
```

```
      '*** NO GRABA NINGUN PEDIDO ***'
```

```
    fin(Sino)
```

```
(* Despliega mensaje de terminacion *)
```

```
'*** TERMINA PROGRAMA DE PEDIDOS ***'
```

```
Cierra archivo de Pedidos
```

```
FIN(SUGPED)
```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO : VALIDA

PROGRAMA DONDE SE LOCALIZA: SABADO

OBJETIVO: Valida la variable que indica si el archivo Maestro contiene información.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO : Ninsuno

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información en el Sistema, variable de tipo entero.
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.

VARIABLES MODULARES. Ninsuna

VARIABLES LOCALES. Ninsuna

PROCEDIMIENTOS QUE INVOCA: Ninsuno

PSEUDOCODIGO:

(* Inicia procedimiento VALIDA *)

INICIA (VALIDA)

(* Se valida si el archivo Maestro contiene información *)

Si LIM = 0 entonces

 inicia(Si)

 I <-- LIM+1

 Despliega mensaje de error de archivo vacío

 fin(Si)

FIN (VALIDA)

10.2 NOMBRE DEL PROCEDIMIENTO: LPEDIDO

PROGRAMA DONDE SE LOCALIZA: MOD1

OBJETIVO: Despliega información de los medicamentos y de los proveedores

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NONYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
DIA	Día en que se está consultando el subsistema, esta variable es de tipo entero.
MES	Mes del día en que se está consultando el subsistema, esta variable es de tipo entero.
ANIO	Año del día en que se está consultando el subsistema, esta variable es de tipo entero.
SUMUNI	Suma de unidades en existencia más unidades vendidas, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
NDV	Número de días que lleva de venta una remesa de medicamentos, variable de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
PV1	Precio de venta que da el proveedor 1, variable de tipo real.
PV2	Precio de venta que da el proveedor 2, variable de tipo real.
PV3	Precio de venta que da el proveedor 3, variable de tipo real.
DV1	Descuento de venta que da el proveedor 1, variable de tipo real.
DV2	Descuento de venta que da el proveedor 2, variable de tipo real.
DV3	Descuento de venta que da el proveedor 3, variable de tipo real.
E1,E2,E3	Espacios que se dejan para centrar la información de salida, variables de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
NUMDIA	Calcula el número de días que lleva de venta una remesa de medicamentos.

PSEUDOCODIGO

(* Inicia procedimiento LPEDIDO *)

INICIA (LPEDIDO)

Despliega información del medicamento
 Activa NUMDIA
 Asigna datos del proveedor a variables de salida
 Despliega información de proveedores

FIN (LPEDIDO)

10.2.1 NOMBRE DEL PROCEDIMIENTO: NUMDIA

PROGRAMA DONDE SE LOCALIZA : MOD1

OBJETIVO: Calcular el número de días que lleva de venta una remesa de medicamentos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
ANIO	Año del día en que se está consultando el subsistema, esta variable es de tipo entero.
MES	Mes del día en que se está consultando el subsistema, esta variable es de tipo entero.
DIA	Día en que se está consultando el subsistema, esta variable es de tipo entero.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
NDV	Número de días que lleva de venta, una remesa de medicamentos, variable de tipo entero.

VARIABLES LOCALES

NOMBRE	DESCRIPCION
DIFA	Diferencia de años que lleva de venta una remesa de medicamentos, variable de tipo entero.
DIFM	Diferencia de meses que lleva de venta una remesa de medicamentos, variable de tipo entero.
DIFD	Diferencia de días que lleva de venta una remesa de medicamentos, variable de tipo entero.
TIPAR	Tipo de año de la fecha en que se cargo la remesa de un medicamento, variable de tipo entero.
TIPAC	Tipo de año de la fecha en que se está consultando el subsistema, variable de tipo entero.
TIPAV	Tipo de año del número de años que lleva de venta una remesa de algún medicamento, variable de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento NUMDIA *)

INICIA (NUMDIA)

```

DIFA <-- Asigna diferencia de años que lleva de venta una remesa
DIFM <-- Asigna diferencia de meses que lleva de venta una remesa
DIFD <-- Asigna diferencia de días que lleva de venta una remesa
(* Se calcula el tipo de año en que se inicia *)
(* la venta de la remesa *)
TIPAR <-- Año de inicio de venta de la remesa entre 4
(* Se calcula el tipo de año en que se está consultando *)
(* el subsistema *)
TIPAC <-- Año de consulta entre 4
(* Proceso para determinar el número de días que lleva *)
(* de venta de una remesa de medicamentos *)
Si DIFA = 0 entonces
  Si DIFM = 0 entonces
    NDV <-- DIFD
  Sino
    inicia (Sino)
      Calcula el número de días del mes de febrero (MES1)
      Calcula el número de días de un mes diferente a febrero (MES2)
      NDV <-- NDV + DIA
    fin (Sino)
  Sino
    Si DIFA menor que 4 entonces
      Calcula el número de días que lleva de venta una remesa
    Sino
      inicia (Sino)
        Despliega mensaje de error
        ' *** EL ARTICULO ESTA OBSOLETO ***'
      NDV <-- 9998
    fin (Sino)
  NDV <-- NDV + 1
FIN (NUMDIA)

```

10.3 NOMBRE DEL PROCEDIMIENTO: LPED

PROGRAMA DONDE SE LOCALIZA: MOD3

OBJETIVO: Despliega por pantalla información del medicamento.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADÍSTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
CVEPRO	Clave del proveedor principal, variable de tipo entero.
CVEMED	Clave del medicamento, variable de tipo entero.
UNIDP	Unidades a pedir, variable de tipo entero.
PRECV	Precio de venta que nos da el proveedor principal, variable de tipo real.
DESCV	Descuento que nos da el proveedor principal, variable de tipo real.

VARIABLES MODULARES. Ninguna

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento LPED *)

INICIA (LPED)

Despliega información del medicamento
CVEPRO <-- Asigna clave de proveedor
CVEMED <-- Asigna clave de medicamentos
UNIDP <-- Asigna unidades pedidas
PRECV <-- Asigna precio de venta del proveedor
DESCV <-- Asigna descuento de proveedor

FIN (LPED)

10.4 NOMBRE DEL PROCEDIMIENTO: PIDE

PROGRAMA DONDE SE LOCALIZA: SABADO

OBJETIVO: Controlar el proceso para elaborar pedidos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
PRECV	Precio de venta que nos da el proveedor principal, variable de tipo real.
DESCV	Descuento que da el proveedor principal, variable de tipo real.
UNIDP	Unidades a pedir, variable de tipo entero.
LIM	Número de clave máxima que contiene información en el Sistema, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
SN	Respuesta de si ó no, variable de tipo caracter.
BAND	Bandera para detectar error al transmitir algún dato, variable de tipo caracter.
SPP	Suma del pago neto total de lo que lleva pedido, variable de tipo real.

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
DAINF	Controla el proceso de pedir información, para hacer un pedido de un medicamento.
GRAPED	Graba los datos del pedido en el archivo de pedidos.

PSEUDOCODIGO

(* Inicia procedimiento PIDE *)

INICIA (PIDE)

Despliega pregunta

'DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O =>'

Repite

Lee respuesta

Si SN es igual a 'S' o 's' entonces

inicia (Si)

despliega pregunta

'DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD'

'SUGERIDA, TRANSMITA S)I O N)O =>'

Repite

Lee respuesta

En caso que SN

sea igual a 'S' o 's' despliega el siguiente mensaje
'GRABA PEDIDO'

ó

sea igual a 'N' o 'n' Activa DAINF

ó

sea diferente de 'S' o 'N' Despliega mensaje de error y vuelve a pedir el dato.

Hasta que BAND sea igual a ''

Activa GRAPED

SPP <-- Asigna la cantidad que se lleva pedida

Despliega la siguiente pregunta

'DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA'

'AHORA PEDIDO?, TRANSMITA S)I O N)O =>'

Repite

BAND <-- ''

Lee respuesta

En caso que SN

sea igual a 'S' o a 's' Despliega lo siguiente
'HASTA AHORA LLEVA PEDIDO \$ 9999.99'

ó

sea igual a 'N' o a 'n' Despliega una linea en blanco

ó

sea diferente de 'S' o 'N' Despliega mensaje de error y vuelve a solicitar el dato

Hasta que BAND sea igual a ''

fin (Si)

Sino

Despliega mensaje

'DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>'

Repite

BAND <-- ''

Lee respuesta

Valida respuesta

Hasta que BAND sea igual a ''

FIN (PIDE)

10.4.1 NOMBRE DEL PROCEDIMIENTO: DAINF

PROGRAMA DONDE SE LOCALIZA: MOD9

OBJETIVO: Controlar el proceso de pedir información para hacer un pedido de un medicamento.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
CVEPRO	Clave del proveedor principal, variable de tipo entero.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
S	Bandera para detectar si hay error al transmitir algún dato, variable de tipo caracter.
SN	Respuesta de si ó no, variable de tipo caracter.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
DATODO	Pide todos los datos para elaborar un pedido.
DAUAU	Pide uno a uno los datos para elaborar un pedido.

PSEUDOCODIGO

(* Inicia procedimiento DAINF *)

INICIA (DAINF)

Despliega pregunta

'A QUE PROVEEDOR DESEA HACER EL PEDIDO'

'DEME NUMERO DE PROVEEDOR =>'

Repite

Lee respuesta

Valida dato capturado

Hasta que S sea igual a ' '

Despliega pregunta

'DESEA DAR TODOS LOS DATOS DEL PEDIDO, TRANSMITA S)I O N)O =>'

Repite

Lee respuesta

En caso que SN

sea igual a 'S' o 's' Activa DATODO

ó

sea igual a 'N' o 'n' Activa DAUUA

ó

sea diferente de 'S' o 'N' Despliega mensaje de error y vuelve a pedir el dato.

Hasta que S sea igual a ' '

FIN (DAINF)

10.4.1.1 NOMBRE DEL PROCEDIMIENTO: DATODO

PROGRAMA DONDE SE LOCALIZA: MOD9

OBJETIVO: Pide todos los datos para elaborar un pedido.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
PRECV	Precio de venta que nos da el proveedor principal, variable de tipo real.
DESCV	Descuento que nos da el proveedor principal, variable de tipo real.
II	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento, variable de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
DAMEU	Pide y valida el dato de unidades pedidas
DAMED	Pide y valida el dato de descuento de proveedores.

PSEUDOCODIGO

(* Inicia procedimiento DATODO *)

INICIA (DATODO)

```
Activa DAMEU
Despliega mensaje
'DENE PRECIO POR UNIDAD =>'
Lee precio
Activa DAMED
(* Asigna el precio y el descuento en los campos correspondientes*)
(* de el archivo de Nombres y Estadísticas del medicamento *)
NM.PRE1 <-- PRECV
NM.DPI <-- DESCV
Escribe estos datos en el archivo de Nom y Est del med.
Valida escritura
```

FIN (DATODO)

10.4.1.2 NOMBRE DEL PROCEDIMIENTO: DAUAV

PROGRAMA DONDE SE LOCALIZA.: MOD9

OBJETIVO: Pide uno a uno los datos para elaborar el pedido.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
PRECV	Precio de venta que nos da el proveedor, variable de tipo real.
DESCV	Descuento de venta que nos da el proveedor principal, variable de tipo real.
II	Apuntador para acceder el archivo de Nombres y Estadísticas, variable de tipo entero.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
BAND3	Bandera para detectar si hay error al transmitir, el dato, variable de tipo caracter.
RSN	Respuesta de si ó no, variable de tipo caracter.

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
DAMEU	Pide y valida el dato de unidades pedidas
DAMED	Pide y valida el dato de descuento de proveedores.

PSEUDOCODIGO

(* Inicia el Procedimiento de DAUUAU *)

INICIA (DAUUAU)

Despliega pregunta

'DESEA DAR LA CANTIDAD DE ARTICULOS A COMPRAR'
'TRANSMITA S)I O N)O =>'

Repite

BAND3 <-- ' '

Lee dato

En caso que RSN

sea igual a 'S' o 's'
ó

Activa DAMEU

sea igual a 'N' o 'n'

Despliega el siguiente mensaje
'SE TOMA LA CANTIDAD SUGERIDA'

ó

sea diferente a 'S' o 'N'

Despliega mensaje de error y vuelve a pedir el dato.

Hasta que BAND3 sea igual a ' '

Despliega pregunta

'DESEA DAR EL PRECIO X ARTICULO, TRANSMITA S)I O N)O =>'

Repite

BAND3 <-- ' '

Lee dato

En caso que RSN

sea igual a 'S' o 's'
ó

Pide precio, Lee dato y asigna el dato al archivo de Nom y Est del med.

sea igual a 'N' o 'n'

Despliega el siguiente mensaje
'SE TOMA EL PRECIO DEL PROVEEDOR ESCOGIDO'

ó

sea diferente a 'S' o 'N'

Despliega mensaje de error y vuelve a pedir el dato.

Hasta que BAND3 sea igual a ' '

Despliega pregunta

'DESEA DAR EL DESCUENTO, TRANSMITA S)I O N)O =>'

Repite

BAND3 <-- ' '

Lee dato

En caso que RSN

sea igual a 'S' o 's'
ó

Activa DAMED
Asigna el dato del descuento al archivo del Nom. y Est. del medicamento.

sea igual a 'N' o 's'

Despliega el siguiente mensaje
'SE TOMA EL DESCUENTO DEL PROVEEDOR ESCOG.'

sea diferente a 'S' o 'N'

Despliega mensaje de error y vuelve a pedir el dato.

Hasta que BAND3 sea igual a ' '

Se graban los datos en el archivo de Nom y Est de med.

Se valida la escritura

FIN (DAUUAU)

10.4.1.1.1 NOMBRE DEL PROCEDIMIENTO: DAMEU

PROGRAMA DONDE SE LOCALIZA: MOD9

OBJETIVO: Pide y valida el dato de unidades pedidas.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
UNIDP	Unidades a pedir, variable de tipo entero.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
BAND3	Bandera para detectar algún error en la transmisión del dato, variable de tipo caracter.

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO.

(* Inicia Procedimiento DAMEU *)

INICIA (DAMEU)

```
Despliega pregunta
'CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO =>'
Repite
  BAND3 <-- ' '
  Lee dato
  Si UNIDP es mayor que 32000 entonces
    inicia (Si)
      Despliega mensaje de error
      BAND3 <-- '*'
    fin (Si)
  Hasta que BAND3 sea igual a ' '
```

FIN (DAMEU)

10.4.1.1.2 NOMBRE DEL PROCEDIMIENTO: DAMED

PROGRAMA DONDE SE LOCALIZA: MOD9

OBJETIVO: Pide y valida el dato de descuento de proveedores.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
DESCV	Descuento que hace el proveedor principal, variable de tipo real.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
BAND3	Bandera para detectar algún error en la transmisión del dato, variable de tipo entero.

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA. Ninguno

PSEUDOCODIGO.

(* Inicia procedimiento DAMED *)

INICIA (DAMED)

```
Despliega pregunta
'DEME EL DESCUENTO QUE LE HACE EL PROVEEDOR =>'
Repite
BAND3 <-- ' '
Lee dato
Si DESCV es mayor que 1 entonces
  DESCV <-- DESCV entre 100.0
Si DESCV es mayor que 100 entonces
  inicia (Si)
  Despliega mensaje de error
  BAND3 <-- '*'
  fin (Si)
Hasta que S = ' '
```

FIN (DAMED)

10.4.2 NOMBRE DEL PROCEDIMIENTO: GRAPED

PROGRAMA DONDE SE LOCALIZA: SABADO

OBJETIVO: Graba los datos del pedido en el archivo de pedidos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
PD PEDIDOS.FIL	ARCHIVO DE PEDIDOS

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I1	Apuntador para acceder el archivo de pedidos, variable de tipo entero.
CVEMED	Clave del medicamento, variable de tipo entero.
UNIDP	Unidades pedidas, variable de tipo entero.
CVEPRO	Clave del proveedor principal, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
ACVE	Arreslo de claves de proveedores, tipo entero.
ACVEA	Arreslo de claves auxiliares de proveedores, tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
IA	Variable auxiliar, que almacena el apuntador que sirve para acceder el archivo de Pedidos, tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO

(* Inicia procedimiento GRAPED *)

INICIA (GRAPED)

(* Incrementa el apuntador para grabar otro registro de pedidos *)

I1 <-- I1 + 1

(* Asigna los datos del pedido a los campos correspondientes *)

(* en el archivo de Pedidos *)

PD.CLAVE <-- CVEMED

PD.UR <-- UNIDP

PD.CP <-- CVEPRO

Si ACVE (CVEPRO) es igual a 0 entonces
inicia (Si)

(* Inicializa arreglos *)

ACVE (CVEPRO) <-- I1

ACVEA (CVEPRO) <-- I1

IA <-- I1

fin (Si)

Sino

inicia (Sino)

Escribe (ESC) los datos en el archivo de Pedidos

IA <-- I1

I1 <-- ACVE (CVEPRO)

Lee (LEE) el archivo de Pedidos

Asigna (ASIGNA) los datos al archivo de Pedidos

(* Guarda apuntador *)

PD.AP <-- IA

ACVE (CVEPRO) <-- IA

fin (Sino)

Escribe (ESC) los datos en el archivo de Pedidos

I1 <-- IA

FIN (GRAPED)

10.5 NOMBRE DEL PROCEDIMIENTO: GRAAPT

PROGRAMA DONDE SE LOCALIZA: SABADO

OBJETIVO: Graba el apuntador en el archivo de Pedidos, el cual indica el orden en el que deberán aparecer en el momento de ser listados.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
PD	PEDIDOS.FIL ARCHIVO DE PEDIDOS

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I1	Apuntador para acceder el archivo de Pedidos, variable de tipo entero.

VARIABLES MODULARES. Ninguna

NOMBRE	DESCRIPCION
ACVE	Arreglo de claves de proveedores, tipo entero.
ACVEA	Arreglo de claves auxiliares de proveedores, tipo entero.

VARIABLES LOCALES.

NOMBRE	DESCRIPCION
I9	Indice para recorrer los arreglos de las claves de proveedores, variable de tipo entero.
IB	Variable para detectar el primer elemento del arreglo de claves de proveedores diferente de 0, variable de tipo entero.
VACP	Variable auxiliar para almacenar la clave de proveedores, de tipo entero.

PROCEDIMIENTOS QUE INVOCA. Ninguno

PSEUDOCODIGO.

(* Inicia procedimiento GRAAPT *)

INICIA (GRAAPT)

IB <-- 0

Haz desde I9=1 hasta 10 lo siguiente

 inicia (Haciendo)

 Si ACVE (I9) es diferente de 0 entonces

 inicia (Si)

 IB <-- IB + 1

 (* Detecta el primer elemento del arreslo ACVE diferente de 0 *)

 Si IB es igual a 1 entonces

 VACP <-- ACVE (I9)

 (* Graba los apuntadores en el archivo de Pedidos *)

 Si IB es mayor que 1 entonces

 inicia (Si)

 I1 <-- VACP

 Lee (LEE) el archivo de Pedidos

 Asigna (ASIGNA) los datos al archivo de Pedidos

 (* Asigna apuntadores *)

 PD.AP <-- ACVEA (I9)

 Escribe (ESC) los datos en el archivo de Pedidos

 VACP <-- ACVE (I9)

 fin (Si)

 fin (Si)

 fin (Haz)

 I1 <-- 0

 Escribe (ESC) los datos en el archivo de Pedidos

 Asigna (ASIGNA) los datos al archivo de Pedidos

 PD.AP <-- 0

 Escribe (ESC) los datos en el archivo de Pedidos

FIN (GRAAPT)

III.2.2.3 DOCUMENTACION DEL MODULO: LISPED

NOMBRE EN EL MENU: LISTAR PEDIDOS

PROGRAMA DONDE SE LOCALIZA : MOD2

1. OBJETIVO. Listar la información del archivo de Pedidos, de los pedidos que se han suserido comprar.

2. FUNCIONES PRINCIPALES:

- Listar la información del archivo de Pedidos(PD).

3. NARRATIVA DEL MODULO:

Si el archivo de Pedidos(PD), no contiene información, el programa le desplegará un mensaje de error y terminará. Si el archivo contiene información, entonces el programa le empezará a listar los medicamentos que usted susirió deben comprarse. Este listado de medicamentos se hará por proveedor y al final le aparecerá el monto neto total del pedido por cada proveedor.

4. ARCHIVOS QUE UTILIZA EL MODULO.

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
PD	PEDIDOS.FIL	ARCHIVO DE PEDIDOS
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
I1	Apuntador para acceder el archivo de pedidos, variable de tipo entero.
II	Apuntador para acceder el archivo de Nom. y Est. del medicamento, variable de tipo entero.
LIM	Número de la clave máxima que contiene información en el Sistema, variable de tipo entero.
PRECV	Precio de venta que da el proveedor principal, variable de tipo real.
DESCV	Descuento que da el proveedor principal, variable de tipo real.
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo Maestro, es de tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
LIE	Variable de control para listar el encabezado del proveedor, tipo entero.
E1,E2,E3	Espacios que se deben dejar para centrar la información de salida, estas variables son de tipo entero.
NOMMED	Nombre del medicamento, variable de tipo string.
PNXMED	Paso neto por medicamento, variable de tipo real.
NOMP	Nombre del proveedor principal, variable de tipo string.
MESP	Nombre del mes en que se desea listar los pedidos, variable de tipo string.
AAP	Año del día en se listan los pedidos, variable de tipo entero.
MMP	Mes del día en que se listan los pedidos, variable de tipo entero.
DDP	Día en que se listan los pedidos, variable de tipo entero.

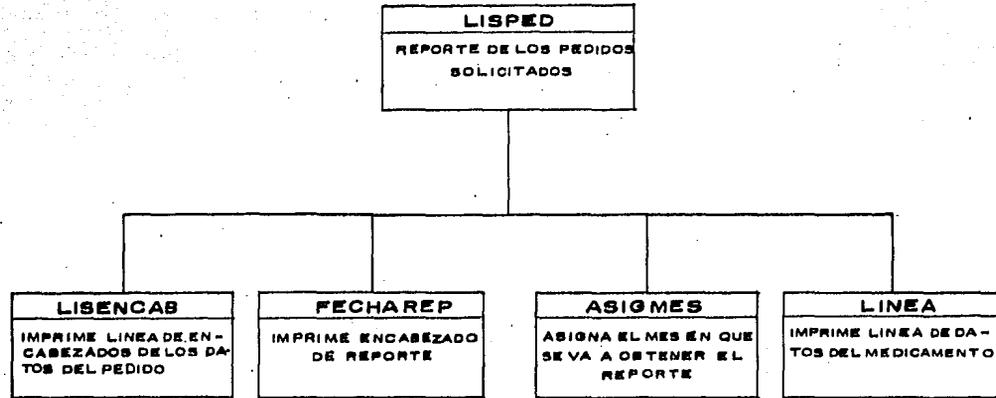
5.3 VARIABLES LOCALES.

NOMBRE	DESCRIPCION
CP	Clave del proveedor principal, variable de tipo entero.
LONGNM	Longitud del nombre del medicamento, variable de tipo entero.
NENM	Número de espacios que quedan en la cadena de caracteres que conforman el nombre del medicamento, variable de tipo entero.
LISP	Controla el proceso de listar pedidos, variable de tipo caracter.
SUMPNTF	Suma del paso neto total por proveedor, variable de tipo real.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

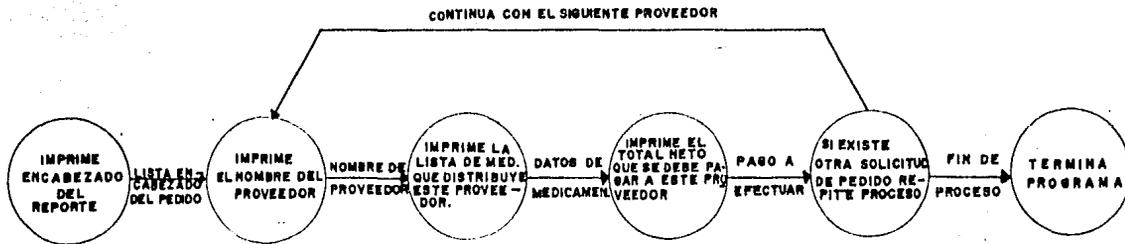
NOMBRE	OBJETIVO
LISENCAB	Listar el encabezado, con el nombre del proveedor.
ASIGMES	Asigna el mes en el que se desea listar los pedidos.
ERRCHK	Valida las lecturas y las escrituras que se hacen a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Finaliza la sesión.

DIAGRAMA JERARQUICO



□ 97 3

DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO.

(* Inicia Programa de LISPED *)

INICIA (LISPED)

Asigna Archivo de Pedidos (PD)

Si LIM es mayor que 0 entonces

inicia Si(1)

Si I1 es mayor que 0 entonces

inicia Si(2)

LIE <-- 0

SUMPNTP <-- 0

I1 <-- 0

Lee Archivo de Pedidos

Valida lectura

Si ERROR entonces Activa EXIT

(* Inicializa variables *)

I1 <-- PD.AP

AAP <-- PD.CLAVE

MMP <-- PD.UR

DDP <-- PD.CP

Activa ASIGNES

fin Si(2)

Sino

Despliega mensaje de error

'*** EL ARCHIVO AUN NO CONTIENE INFORMACION, FAVOR DE ***'

' *** EJECUTAR LA OPCION <A> PARA CARGAR PEDIDOS ***'

Si I1 es diferente de 0 entonces

Inicia Si(3)

Despliega mensaje del encabezado del reporte

Lee Archivo de Pedidos

Valida lectura

I1 <-- asigna clave de medicamento (PD.CLAVE)

CP <-- asigna clave de proveedor (PD.CP)

Repite

LISP <-- ' '

Lee archivo de Nom. y Est. del med.

Asigna los datos del pedido a las variables de salida

PNXMED <-- paso neto por medicamento

SUMPNTP <-- asigna la suma del paso neto total por proveedor

Calcula espacios para centrar el descuento en la línea de impresión

I1 <-- asigna la clave del siguiente medic. que debe ser listado

Si I1 es diferente de 0 entonces

inicia Si(4)

Lee archivo de Pedidos

Valida lectura

fin Si(4)

Sino

inicia Sino(4)

CP <-- 0

LISP <-- '*'

fin Sino(4)

Si CP es igual a PD.CP entonces

inicia Si(5)

Si LIE es igual a 0 entonces

activa LISENCAR

Imprime línea de datos (Activa LINEA)

fin Si(5)

```
Sino
  inicia Sino(5)
  Si LIE es igual a 0 entonces.
    activa LISENCAB
    Imprime linea de datos (Activa LINEA)
    Despliega mensaje de lo que se lleva pedido
    'TOTAL DEL PEDIDO A ESTE PROVEEDOR .- $ 99999.99'
    SUMPNTP <-- 0
    LIE <-- 0
    CP <-- PD.CP
  fin Sino(5)
  I1 <-- PD.CLAVE
  Hasta que LISP sea igual a '*'
  fin Si(3)
fin Si(1)
Sino
  Despliega mensaje de error
  Despliega mensaje de finalizacion de programa.
FIN (LISPED)
```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: LISENCAB

NOMBRE DEL PROGRAMA DONDE SE LOCALIZA : MOD2

OBJETIVO: Listar el encabezado del proveedor.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO. Ninguno

VARIABLES.

VARIABLES EXTERNAS. Ninguna

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
NOMP	Nombre del proveedor principal, variable de tipo string.
LIE	Variable de control para listar el encabezado del proveedor, tipo entero.

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento de LISENCAB *)

INICIA (LISENCAB)

Despliega el nombre del proveedor
Despliega el encabezado de la información que se
requiere para el pedido

FIN (LISENCAB)

10.2 NOMBRE DEL PROCEDIMIENTO : ASIGMES

NOMBRE DEL PROGRAMA DONDE SE LOCALIZA : MOD2

OBJETIVO: Asignar el mes en el que se desea listar los pedidos.

ARCHIVOS QUE USA EL PROCEDIMIENTO. Ninguno

VARIABLES.

VARIABLES EXTERNAS

NOMBRE	DESCRIPCION
MMP	Mes del día en que se desea listar los pedidos, variable de tipo entero.
MESP	Nombre del mes en que se desea listar los pedidos, variable de tipo string.

VARIABLES LOCALES. Ninguna

PROCEDIMIENTOS QUE INVOCA. Ninguno

PSEUDOCODIGO

(* Inicia el procedimiento de ASIGMES *)

INICIA (ASIGMES)

Asigna el mes del día en que se va a listar los pedidos

FIN (ASIGMES)

III.2.2.4 DOCUMENTACION DEL MODULO PEDXTEM

NOMBRE EN EL MENU: SUGERIR COMPRA DE ARTICULOS POR TEMPORADA

PROGRAMA DONDE SE LOCALIZA: SABADO

1. OBJETIVO: Sugerir compra de medicamentos por temporada.

2. FUNCIONES PRINCIPALES:

- Desplegar por pantalla las temporadas a consultar.
- Desplegar por pantalla los datos necesarios, para elaborar un pedido de los medicamentos que cumplan, con el número de unidades vendidas, que se han fijado como una buena venta, en dicha temporada.
- Proporcionar alternativas para dar los datos del medicamento que se desee pedir.
- Grabar los datos del medicamento en el archivo de Pedidos (PD).
- Desplegar por pantalla, la pregunta de si desea conocer el monto de lo que lleva pedido.

3. NARRATIVA DEL MODULO:

El programa primeramente le desplegará por pantalla, las temporadas que usted puede consultar, para elaborar algunos pedidos de medicamentos, e inmediatamente después de haber desplegado esta información, el programa le pedirá transmita la temporada que desea consultar. Después, el programa le pedirá transmita el número de unidades, que usted considere representan una buena venta en la temporada eleuida.

Si el archivo Maestro (BF), no contiene información, el programa le desplegará un mensaje de error indicándole que el archivo se encuentra vacío, y que necesita activar el módulo ESCRIBE (opción F) para dar de alta algún medicamento.

En caso que el archivo Maestro contenga información, el programa empezará a listar aquellos medicamentos que cumplan, con la condición de que el número de unidades vendidas en esa temporada, sea igual o mayor al número que se considera representa una buena venta, desplegando por pantalla todos los datos necesarios para que el usuario tenga una mejor opción, a la hora de hacer un pedido del medicamento listado.

Si no desea hacer un pedido del medicamento listado, el programa le listará los datos del siguiente medicamento que cumpla con la condición del número de unidades vendidas, en caso contrario el programa le proporcionará la información necesaria para que usted elija, como desea proporcionar los datos del pedido.

Una vez que se han transmitido todos los datos del pedido, se graban en el archivo de Pedidos (PD) e inmediatamente le aparece en pantalla un mensaje presuntándole, si desea conocer cuál es el monto neto total de lo que lleva pedido.

Después el proceso se repite hasta el final del archivo Maestro.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
PD	PEDIDOS.FIL	ARCHIVO DE PEDIDOS
NM	NOMYEST.FIL	ARCHIVO DE NOMBRE Y ESTADIS- TICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
I1	Apuntador para acceder el archivo de Pedidos, variable de tipo entero.
LIM	Número de la clave máxima que contiene información en el Sistema, variable de tipo entero.
II	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento, variable de tipo entero.
OFC2	Opción para elegir temporada, variable de tipo caracter.
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras al archivo Maestro, tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
SPP	Suma del pago neto total de lo que lleva pedido, variable de tipo real.
ACVE	Arreslo de claves de proveedores, tipo entero.
ACVEA	Arreslo de claves auxiliares de proveedores, tipo entero.
SN	Respuesta de si o no, variable de tipo caracter.

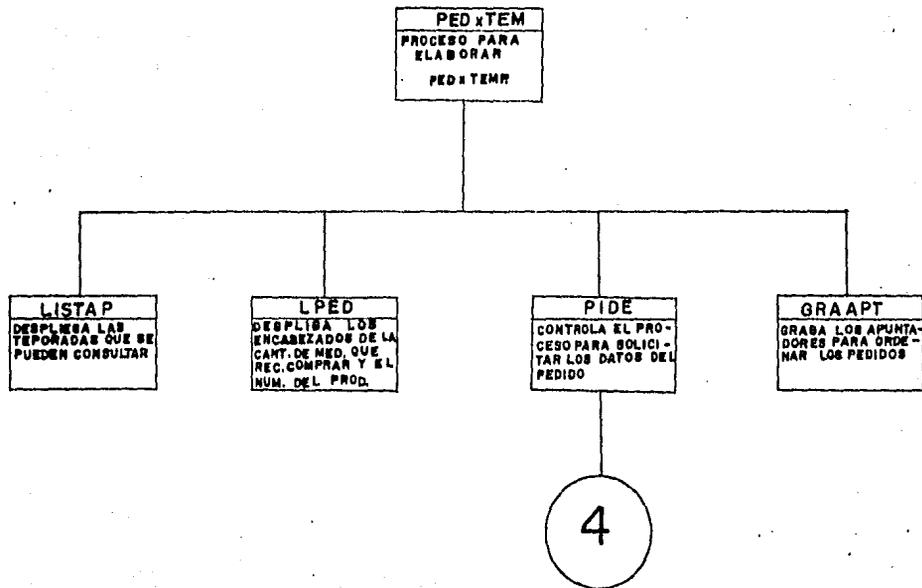
5.3 VARIABLES LOCALES:

NOMBRE	DESCRIPCION
IL	Indice Para limpiar el arreslo de claves de proveedores, variable de tipo entero.
CUVXT	Cantidad de unidades vendidas por temporada, variable de tipo entero.
CUVXTE	Cantidad de unidades vendidas en la temporada elesida.
NOMMED	Nombre del medicamento, variable de tipo string.
LONGNM	Longitud de la cadena de caracteres que conforman el nombre del medicamento, variable de tipo entero.
NECNM	Número de espacios en blanco que quedan en la cadena del nombre del medicamento, variable de tipo entero.
E1,E2	Espacios que se dejan para centrar el nombre del medicamento en el formato de salida, variable de tipo entero.
SUMU	Suma de unidades en existencia más unidades pedidas, variable de tipo entero.

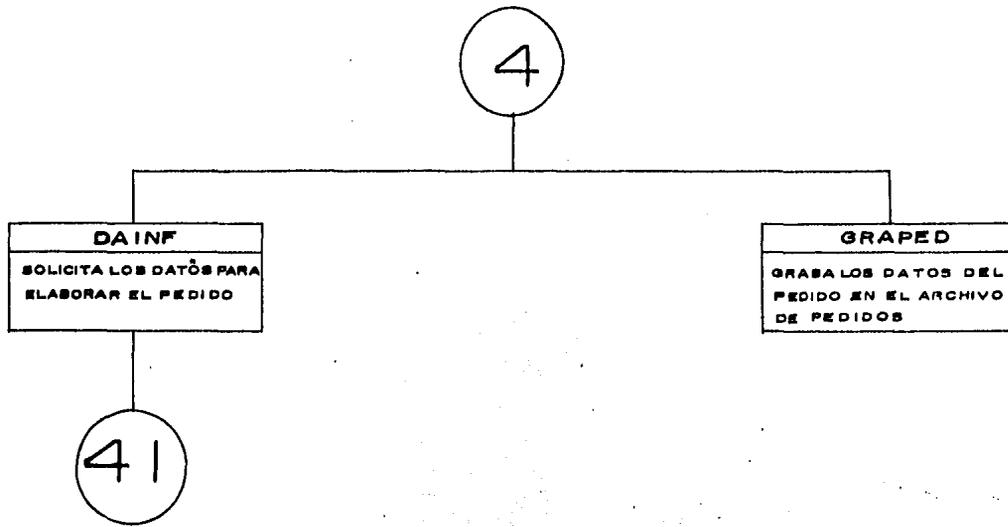
6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
LISTAP	Listar temporadas a consultar.
VALIDA	Valida la variable que indica si el archivo Maestro contiene informacion.
LPED	Despliega por pantalla información del medicamento.
PIDE	Controla el proceso de pedidos.
GRAAPT	Graba el apuntador en el archivo de Pedidos, el cual indica el orden en el que deben aparecer las claves de los medicamentos, en el momento de ser listados.
ERRCHK	Valida las lecturas y las escrituras que se hacen a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Finaliza la sesión.

DIAGRAMA JERARQUICO

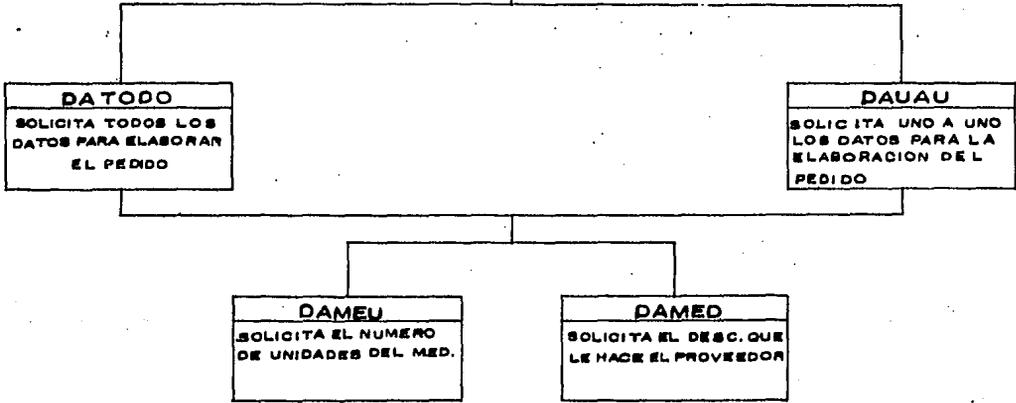


C 107 3



C 1071 3

41



C 1072 3

DIAGRAMA DE FLUJO DE DATOS

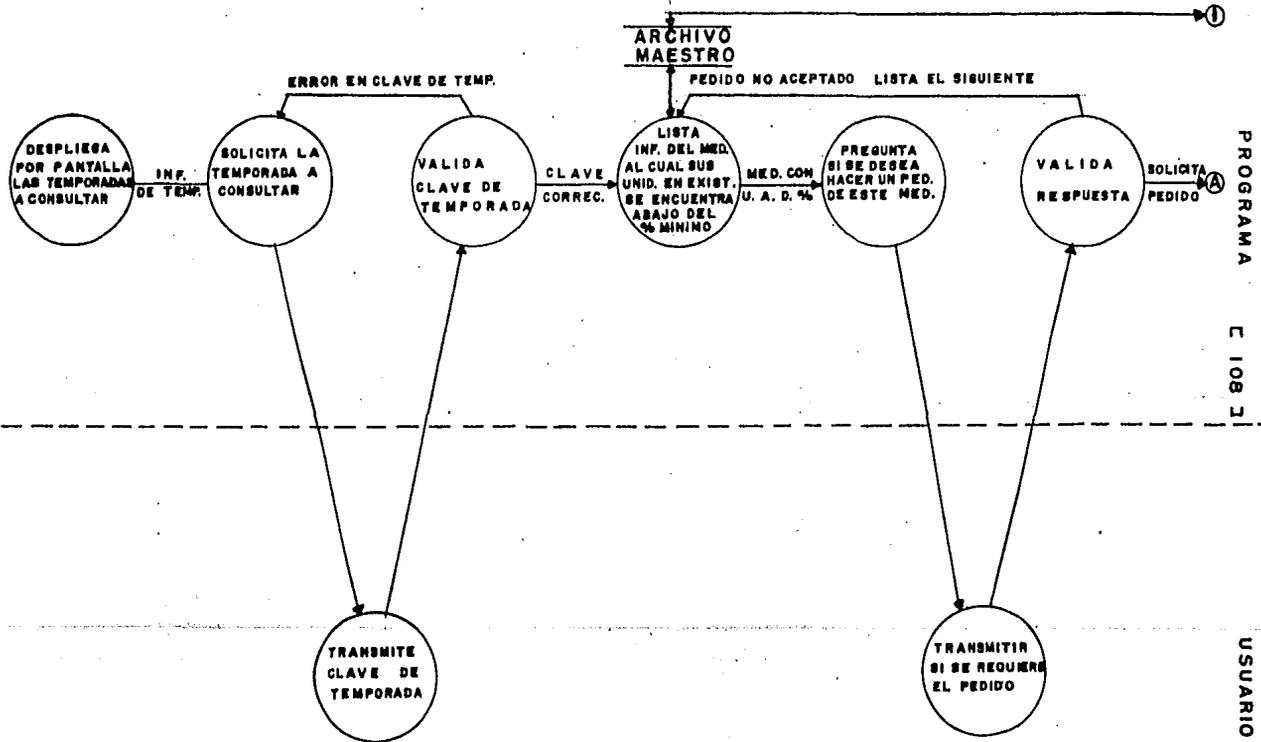
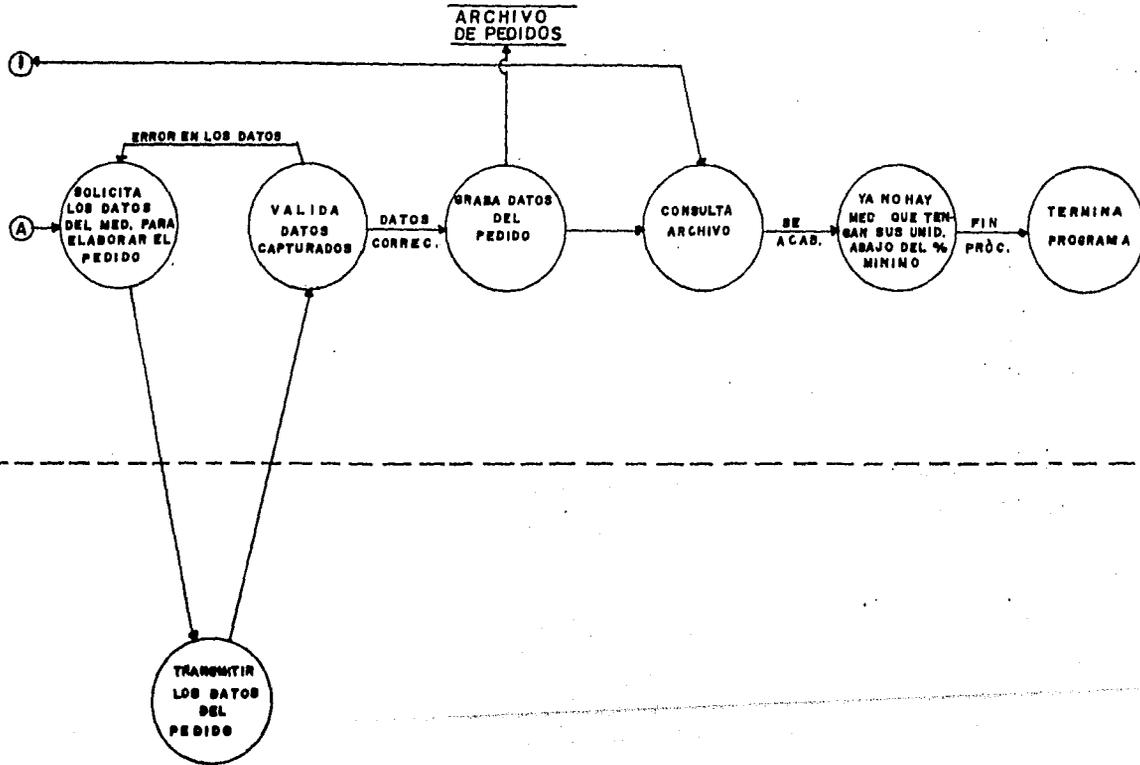


DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

E 1081 7

USUARIO

9. PSEUDOCODIGO

(* Inicia el programa de PEDXTEM *)

INICIA (PEDXTEM)

Asigna el archivo de Pedidos

Si LIM es mayor que 0 entonces

inicia Si

Activa LISTAP

Lee el número de unidades, que se considera como una buena venta

I <-- 1

II <-- 0

SPP <-- 0

Haz desde IL=1 hasta 10 lo siguiente

inicia Haz

(* Inicializa con ceros los arreslos de las claves de proveedores *)

ACVE (IL) <-- 0

ACVEA (IL) <-- 0

fin Haz

Activa VALIDA

(* Proceso para localizar los medicamentos que cumplen con la *)

(* condición del número de unidades vendidas en la temporada *)

Mientras I sea menor o igual a LIM Haz lo siguiente

inicia (Mientras)

II <-- I

Lee el archivo de Nom. y Est. del med.

Valida lectura

Lee el archivo Maestro

En caso que OPC2

sea igual a 'A' asigna a CUVXTE <-- Número de unidades vendidas

ó

sea igual a 'B' asigna a CUVXTE <-- Número de unidades vendidas
en la temporada dos.

ó

sea igual a 'C' asigna a CUVXTE <-- Número de unidades vendidas
en la temporada tres.

ó

sea igual a 'D' asigna a CUVXTE <-- Número de unidades vendidas
en la temporada cuatro.

ó

sea diferente a 'A', 'B', 'C', 'D' Despliega el siguiente mensaje
'*** E R R O R F A T A L ***'

fin (caso)

Si CUVXTE es mayor o igual a CUVXT entonces

inicia (Si)

Despliega información del medicamento

SUMU <-- Unid. en exist. mas unid. vend.

```
Despliega pregunta
'DESEA HACER UN PEDIDO DE ESTE ARTICULO'
'TRANSMITA S)I O N)O =>'
Lee respuesta
Si S es igual a 'S' o a 's' entonces
  inicia (Si)
  Activa LPED
  Activa PIDE
  fin (Si)
fin (Si)
I <-- I + 1
fin (Mientras)
Si LIM es mayor que 0 entonces
Si I1 es mayor que 0 entonces
  Activa GRAAPT
Sino
  Despliega mensaje
  '*** NO GRABO NINGUN PEDIDO ***'
Cierra archivo de Pedidos
Despliega mensaje de terminación
' **** TERMINA PROGRAMA DE PEDIDOS POR TEMPORADA ****'
FIN (PEDXTEM)
```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: LISTAP

PROGRAMA DONDE SE LOCALIZA: MODB

OBJETIVO: Listar temporadas a consultar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninsuno

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
OPC2	Opción para elegir la temporada, variable de tipo caracter.

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
BAND	Bandera para detectar algún error al transmitir el dato, variable de tipo caracter.

PROCEDIMIENTOS QUE INVOCA. Ninguno

PSEUDOCODIGO:

(* Inicia procedimiento LISTAP *)

INICIA (LISTAP)

 Repite

 Despliega las temporadas a consultar
 Pide el dato de la temporada que desea consultar
 Valida la respuesta transmitida
 Hasta que BAND sea igual a ' '

FIN (LISTAP)

10.2 NOMBRE DEL PROCEDIMIENTO: VALIDA

(Esta descripción esta hecha en la documentación del módulo de SUGPED, por lo que se le sugiere consulte dicha sección)

10.3 NOMBRE DEL PROCEDIMIENTO: LPED

(Esta descripción esta hecha en la documentación del módulo de SUGPED, por lo que se le sugiere consulte dicha sección)

10.4 NOMBRE DEL PROCEDIMIENTO: PIDE

(Esta descripción esta hecha en la documentación del módulo de SUGPED, por lo que se le sugiere consulte dicha sección)

10.5 NOMBRE DEL PROCEDIMIENTO: GRAAPT

(Esta descripción esta hecha en la documentación del módulo de SUGPED, por lo que se le sugiere consulte dicha sección)

III.2.2.5 DOCUMENTACION DEL MODULO: CADUCAR

PROGRAMA DONDE SE LOCALIZA: MOD4

1. **OBJETIVO:** Obtener un reporte de los medicamentos que están próximos a caducar, tomando como referencia la fecha del reporte.

2. **FUNCIONES PRINCIPALES:**
 - Asignar el archivo Maestro al proceso.
 - Validar la fecha del reporte.
 - Listar encabezados.
 - Buscar en todo el archivo Maestro, aquellos medicamentos próximos a caducar de acuerdo con la fecha del reporte y listarlos.

3. **NARRATIVA DEL MODULO.**

Solicita y valida la fecha del reporte. Pide el número de días, contados a partir de la fecha anterior, para buscar en el archivo BF, a todos los medicamentos que caducarán en ese periodo.

Tomando como base la fecha de caducidad de cada medicamento y el número de días, despliega en el reporte, aquellos medicamentos que se encuentren en esta condición.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del sistema. Es de tipo entero.
ERRORR	Variable por medio de la cual se verifican las lecturas y escrituras al archivo. Es de tipo booleano.
W1	Espacios en blanco, para centrar letreros. Es de tipo entero.
NNO	Contiene el nombre de la empresa. Es una cadena de 30 caracteres.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
NUMDIAS	Cantidad de días dentro de los cuales caducarán los medicamentos. Es de tipo entero.
I	Apuntador para acceder a los registros del archivo. Es de tipo entero.
CANTMED	Cantidad de medicinas que caducan en NUMDIAS. Es de tipo entero.
TOTANNEX	Resultado de restar el contenido del campo ANN menos ANO. Es de tipo entero.

TOTMESEX	Resultado de restar el contenido del campo MES menos MES. Es de tipo entero.
TOTDIAEX	Resultado de restar el contenido del campo DIA menos DIA. Es de tipo entero.
MESESEX	Resultado de multiplicar 12 por TOTANEX y sumarle TOTMESEX. Es de tipo entero.
TOTALDEX	Total de días que faltan para que caduquen las existencias actuales del medicamento. Resultado de multiplicar MESESEX por 30 y sumarle TOTDIAEX. Es de tipo entero.
TOTANOPE	Resultado de restar el contenido del campo AA menos ANO. Es de tipo entero.
TOTMESPE	Resultado de restar el contenido del campo MM menos MES. Es de tipo entero.
TOTDIAPE	Resultado de restar el contenido del campo DD menos DIA. Es de tipo entero.
MESESPE	Resultado de multiplicar 12 por TOTANOPE y sumarle TOTMESPE. Es de tipo entero.
TOTALDPE	Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TOTDIAPE. Es de tipo entero.
ANO	Contiene el valor del año de la fecha a ser validada. Es de tipo entero.
MES	Contiene el valor del mes de la fecha a ser validada. Es de tipo entero.
DIA	Contiene el valor del día de la fecha a ser validada. Es de tipo entero.
VER	Si la fecha es válida toma el valor 1; de lo contrario vale 0. Es de tipo entero.
SIG	Variable usada para controlar el proceso de validación. Es de tipo caracter.

5.3 VARIABLES LOCALES: Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
LETRA	Escribir el encabezado del reporte.
SEEKREAD	Realiza lecturas de acceso directo.
ERRCHK	Valida lecturas y escrituras a los archivos.
IMPRESION	Escribir en el reporte los datos de aquellas medicinas que caducará.
CALCULOS	Hacer los cálculos necesarios para convertir la fecha de caducidad en días, para poder compararlas con el valor de NUMDIAS.
IMPRESO1	Escribir los datos de las medicinas existentes.
IMPRESO2	Escribir los datos de las medicinas existentes del pedido.
FECHITA	Lee los datos de la fecha y la valida.
ASIGNAR	Asignar los archivos BF y NM al proceso y leer los datos contenidos en el registro cero, de máxima seguridad.

DIAGRAMA JERARQUICO

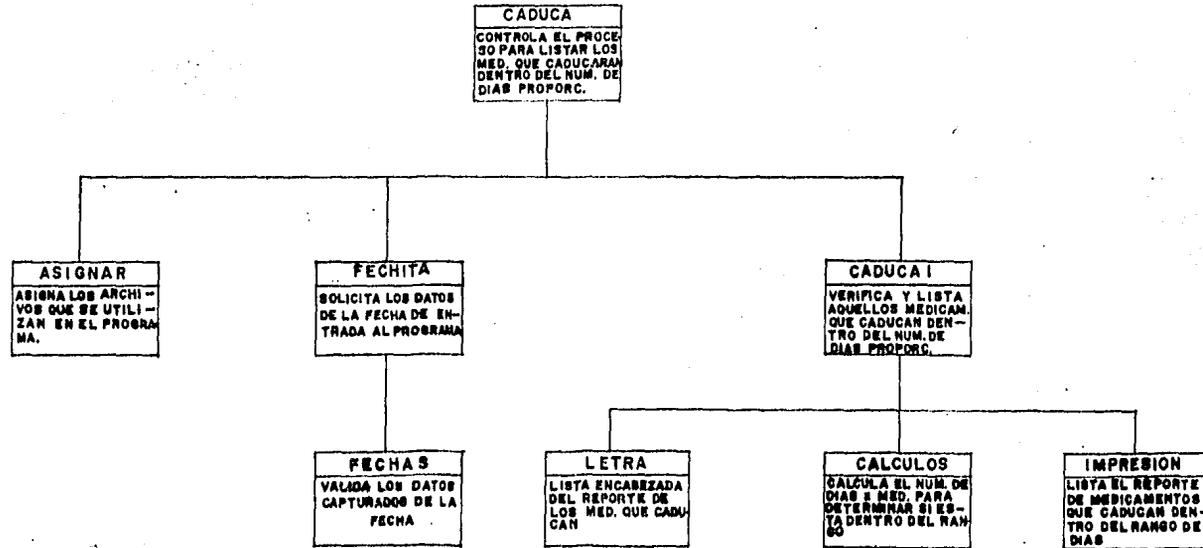
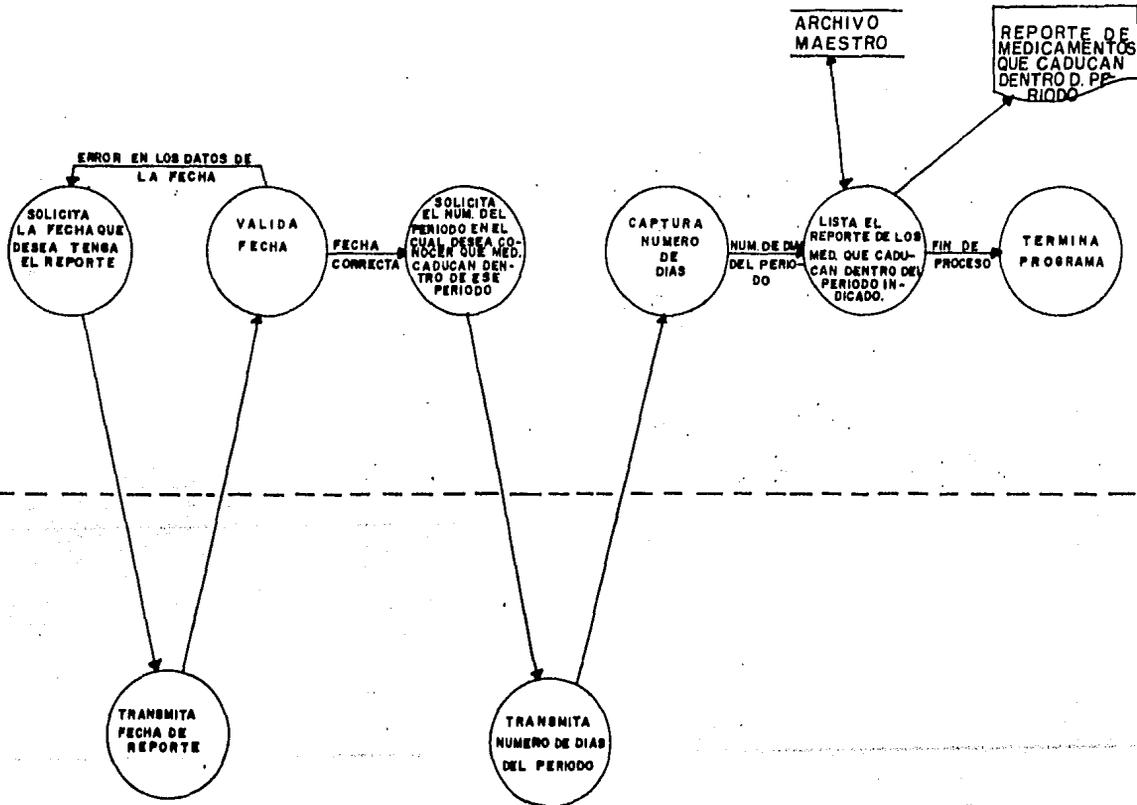


DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

E 118 3

USUARIO

9. PSEUDOCODIGO.

(* Inicia el módulo CADUCAR *)

INICIA (CADUCAR)

```

SI LIM es mayor que 0 entonces
  Inicia (Si)
  Activa ASIGNAR
  Solicita la fecha
  Activa FECHITA
  (* si la fecha es válida VER es igual a 1 *)
  Si VER es igual a 1 entonces
    Activa CADUCA1
sino
  Despliega mensaje de error

```

FIN (CADUCAR)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: CADUCA1

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Buscar los medicamentos próximos a caducar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTA DISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del sistema. Es de tipo entero.

ERROR Variable por medio de la cual se verifican las lecturas y escrituras al archivo. Es de tipo booleano.

W1 Espacios en blanco, para centrar letreros. Es de tipo entero.

NNO Contiene el nombre de la empresa. Es una cadena de 30 caracteres.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
NUMDIAS	Cantidad de días dentro de los cuales caducarán los medicamentos. Es de tipo entero.
I	Apuntador para acceder a los registros del archivo. Es de tipo entero.
CANTMED	Cantidad de medicinas que caducan en NUMDIAS. Es de tipo entero.
TOTALDEX	Total de días que faltan para que caduquen las existencias actuales del medicamento. Resultado de multiplicar MESESEX por 30 y sumarle TOTDIAEX. Es de tipo entero.
TOTALDPE	Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TOTDIAPE. Es de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
II	Variable de control para un proceso iterativo; toma valor desde 1 hasta LIM. Es de tipo entero.

PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
LETRA	Escribir el encabezado del reporte.
SEEKREAD	Realiza lecturas de acceso directo.
ERRCHK	Valida lecturas y escrituras a los archivos.
IMPRESION	Escribir en el reporte los datos de aquellas medicinas que caducarán.
CALCULOS	Hacer los cálculos necesarios para convertir la fecha de caducidad en días, para poder compararlas con el valor de NUMDIAS.

PSEUDOCODIGO.

(* Inicia el procedimiento CADUCA1 *)

INICIA (CADUCA1)

```

Despliega encabezado.
Solicita el número de días que se toma como base para
buscar los medicamentos que caducarán en ese periodo
Lee NUMDIAS
Activa LETRA
  Haz desde que II sea igual a 1 hasta LIM
  Inicia (Haz)
  I <--- II
  Lee el registro I del archivo BF
  Activa ERRCHK
  Si ERROR es verdadero entonces
    Activa EXIT
  fin (Si)
  (* se verifican las existencias *)
  Si BF.UNID es mayor que cero o
  BF.UPED es mayor que cero entonces
    Inicia (Si)
    (* solo busca en aquellos que tienen existencias *)
    Activa CALCULOS
    SI TOTALDEX es menor o igual a NUMDIAS o
    TOTALDPE es menor o igual a NUMDIAS entonces
      Activa IMPRESION
    fin (Si)
  fin (si)
  fin (Haz)
  Si CANTMED es mayor que cero entonces
    Anota el total de medicinas que caducarán
  fin (si)

```

FIN (CADUCA1)

10.2 NOMBRE DEL PROCEDIMIENTO: CALCULOS

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Hacer los cálculos necesarios para convertir la fecha de caducidad en días, para poder compararlas con el valor de NUMDIAS.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
TOTANEX	Resultado de restar el contenido del campo ANN menos ANO. Es de tipo entero.
TOTMESEX	Resultado de restar el contenido del campo MES menos MES. Es de tipo entero.
TOTDIAEX	Resultado de restar el contenido del campo DIA menos DIA. Es de tipo entero.
MESESEX	Resultado de multiplicar 12 por TOTANEX y sumarle TOTMESEX. Es de tipo entero.
TOTALDEX	Resultado de multiplicar MESESEX por 30 y sumarle TOTDIAEX. Es de tipo entero.
TOTANOPE	Resultado de restar el contenido del campo AA menos ANO. Es de tipo entero.
TOTMESPE	Resultado de restar el contenido del campo MM menos MES. Es de tipo entero.
TOTDIAPE	Resultado de restar el contenido del campo DD menos DIA. Es de tipo entero.
MESESPE	Resultado de multiplicar 12 por TOTANOPE y sumarle TOTMESPE. Es de tipo entero.

TOTALDPE Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TOTDIAPE. Es de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
AUXIL1	Multiplicación de los campos de la fecha de la unidades existentes. Es de tipo entero.
AUXIL2	Multiplicación de los campos de la fecha de unidades pedidas existentes. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
LETRA	Escribir el encabezado del reporte.
SEEKREAD	Realiza lecturas de acceso directo.
IMPRESION	Escribir en el reporte los datos de aquellas medicinas que caducarán.

PSEUDOCODIGO.

(* Inicia el Procedimiento CALCULOS *)

INICIA (CALCULOS)

TOTALDEX <--- 0

TOTALDPE <--- 0

(* multiplica los campos de la fecha de las existencias
actuales *)

AUXIL1 <--- BF.ANN * BF.MES * BF.DIA

SI AUXIL1 es mayor que cero entonces

Inicia (Si)

(* la fecha no tiene ceros *)

TOTANNEX <--- BF.ANN - ANO

TOTMESEX <--- BF.MES - MES

TOTDIAEX <--- BF.DIA - DIA

MESESEX <--- 12 * TOTANNEX + TOTMESEX

TOTALDEX <--- MESESEX * 30 + TOTDIAEX

fin (Si)

AUXIL2 <--- BF.AA * BF.MM * BF.DD

SI AUXIL2 es mayor que cero entonces

Inicia (Si)

(* la fecha de las unidades pedidas no tiene ceros *)

TOTANOPE <--- BF.AA - ANO

TOTMESPE <--- BF.MM - MES

TOTDIAPE <--- BF.DD - DIA

MESESPE <--- 12 * TOTANOPE + TOTMESPE

TOTALDPE <--- MESESPE * 30 + TOTDIAPE

fin (Si)

FIN (CALCULOS)

10.3 NOMBRE DEL PROCEDIMIENTO: IMPRESION

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Escribir en el reporte los datos de aquellas medicinas que caducarán.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
NOMMED	Nombre del medicamento. Es una cadena de 30 caracteres.
LONGNOM	Longitud del nombre del medicamento. Es de tipo entero.
SANGRIA	Resultado de 30 menos LONGNOM. Es de tipo entero.
TOTALDEX	Total de días que faltan para que caduquen las existencias actuales del medicamento. Resultado de multiplicar MESESEX por 30 y sumarle TOTDIAEX. Es de tipo entero.
TOTALDPE	Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TOTDIAPE. Es de tipo entero.
NUMDIAS	Cantidad de días dentro de los cuales caducarán los medicamentos. Es de tipo entero.

10.3 NOMBRE DEL PROCEDIMIENTO: IMPRESION

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Escribir en el reporte los datos de aquellas medicinas que caducarán.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS: Ninsuna.

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
NOMMED	Nombre del medicamento. Es una cadena de 30 caracteres.
LONGNOM	Longitud del nombre del medicamento. Es de tipo entero.
SANGRIA	Resultado de 30 menos LONGNOM. Es de tipo entero.
TOTALDEX	Total de días que faltan para que caduquen las existencias actuales del medicamento. Resultado de multiplicar MESESEX por 30 y sumarle TOTDIAEX. Es de tipo entero.
TOTALDPE	Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TOTDIAPE. Es de tipo entero.
NUMDIAS	Cantidad de días dentro de los cuales caducarán los medicamentos. Es de tipo entero.

CANTMED Cantidad de medicinas que caducan en NUMDIAS.
Es de tiro entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
IMPRESO1	Escribir los datos de las medicinas existentes.
IMPRESO2	Escribir los datos de las medicinas existentes del pedido.

PSEUDOCODIGO.

(* Inicia el procedimiento IMPRESION *)

INICIA (IMPRESION)

```
Lee el registro I del archivo BF
Activa ERRCHK
  Si ERROR es verdadera entonces
    Activa EXIT
  fin (Si)
(* hace las asignaciones *)"
NOMMED <--- BF.NOMBRE
LONGNOM <--- longitud de NOMMED
SANGRIA <--- 30 - LONGNOM
Si BF.UNID es mayor que cero y
  TOTALDEX es menor que NUMDIAS y
  BF.UPED es mayor que cero y
  TOTALDPE es menor o igual que NUMDIAS entonces
  Inicia (si)
  Escribe el nombre del medicamento
  (* escribe datos de identificación del medicamento *)
  Activa a IMPRESO1 e IMPRESO2
  fin (si)
sino
```

```
Inicia (sino)
Si BF.UNID es mayor que cero y
  TOTALDEX es menor o igual que NUMDIAS entonces
  Inicia (si)
  Escribe el nombre del medicamento (si no ha sido escrito)
  Activa IMPRESO1
  fin (si)
sino
  Inicia (sino)
  Si BF.UNID es mayor que cero y
    TOTALDPE es menor o igual que NUMDIAS entonces
    Inicia (si)
    Escribe el nombre del medicamento (si no ha sido escrito)
    Activa IMPRESO2
    fin (si)
  fin (sino)
fin (sino)
Si se escribió el nombre del medicamento entonces
  Incrementa CANTMED en uno
fin (Si)

FIN (IMPRESION).
```

10.4 NOMBRE DEL PROCEDIMIENTO: IMPRESO1

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Escribir los datos de las medicinas existentes que caducarán en NUMDIAS.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
TOTALDEX	TOTALDEX Total de días que faltan para que caduquen las existencias actuales del medicamento. Resultado de multiplicar MESESEX por 30 y sumarle TOTDIAEX. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento IMPRESO1 *)

INICIA (IMPRESO1)

Escribe en un renglón, los siguientes datos:
CLAVE UNIDADES EN EXISTENCIA FECHA DE CADUCIDAD
y TOTALDEX

FIN (IMPRESO1)

10.5 NOMBRE DEL PROCEDIMIENTO: IMPRESO2

PROGRAMA DONDE SE LOCALIZA: MODA

OBJETIVO: Escribir los datos de las medicinas existentes del pedido que caducarán en NUMDIAS.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
--------	-------------

TOTALDPE	TOTALDPE Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TOTDIAPE. Es de tipo entero.
----------	--

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento IMPRESO2 *)

INICIA (IMPRESO2)

Escribe en un renglón, los siguientes datos:
CLAVE UNIDADES DEL PEDIDO EN EXISTENCIA FECHA DE CADUCIDAD
CUANTOS DIAS FALTAN

FIN (IMPRESO2)

10.6 NOMBRE DEL PROCEDIMIENTO: ASIGNAR

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Asignar los archivos BF y NM al proceso y leer los datos contenidos en el registro cero, de máxima seguridad.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
NNO	
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
TOTALDPE	Total de días que faltan para que caduquen las unidades pedidas del medicamento. Resultado de multiplicar MESESPE por 30 y sumarle TODIAPE. Es de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
LONGN	Número de caracteres en el nombre de la empresa. Es de tipo caracter.
L	Apuntador para acceder al registro cero del archivo NM. Es de tipo entero.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.

PSEUDOCODIGO.

(* Inicia el procedimiento ASIGNAR *)

INICIA (ASIGNAR)

```
Asignar el Archivo BF
Asignar el Archivo NM
L <--- 0
Lee el registro L del archivo NM
Activa ERRCHK
  Si ERROR es verdadero entonces
    Activa EXIT
  fin (Si)
(* asigna nombre del medicamento *)
NNO <--- NM.NOMBRE
LONGN <--- longitud de NNO
M1 <--- 30 - LONGN
CANTHED <--- 0
```

FIN (ASIGNAR)

10.7 NOMBRE DEL PROCEDIMIENTO. LETRA.

PROGRAMA DONDE SE LOCALIZA: MOD4.

OBJETIVO: Desplegar el encabezado del reporte.

ARCHIVOS QUE USA EL PROCEDIMIENTO: Ninguno.

VARIABLES. Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento LETRA *)

INICIA (LETRA)

Escribe el encabezado del reporte para los medicamentos que cadurán dentro del periodo marcado.

FIN (LETRA)

10.8 NOMBRE DEL PROCEDIMIENTO: FECHITA

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Lee los datos de la fecha y la valida.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
ANO	Contiene el valor del año de la fecha a ser validada. Es de tipo entero.
MES	Contiene el valor del mes de la fecha a ser validada. Es de tipo entero.
DIA	Contiene el valor del día de la fecha a ser validada. Es de tipo entero.
VER	Si la fecha es válida toma el valor 1; de lo contrario vale 0. Es de tipo entero.
SIG	Variable usada para controlar el proceso de validación. Es de tipo caracter.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
AN1	Sirve para ver si un año es bisiesto. Es de tipo entero.
DI2	Número de días que tiene el mes. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento FECHITA *)

INICIA (FECHITA)

Mensaje: "ANOTE LA FECHA DE HOY : "

SIG <--- /

Repite

ANO <--- 0

MES <--- 0

DIA <--- 0

Solicita los datos del año, mes y día por medio de las variables ANO, MES y DIA

VER <--- 0

Si ANO es mayor o igual que 86 y

ANO es menor que 100 y

MES es mayor que 0 y

DIA es mayor que 0 entonces

Inicia (Si)

VER <--- 1

ANI <--- ANO MOD 4

En caso que MES

sea igual a 1,3,5,7,8,10,12: DI2 <---31

o

sea igual a 4,6,9,11 : DI2 <--- 30

o

sea igual a 2 : Inicia (MES =2)
 Si AN1 es igual a 0 entonces
 DI2 <--- 29
 sino
 DI2 <--- 28
 fin (Si)
 fin (MES =2)

sino : VER <--- 0

fin (caso)

Si VER es igual a 1 entonces

Inicia (si)

SI DIA es mayor que DI2 entonces

VER <--- 0

fin (Si)

fin (Si)

```
Si VER es igual a 1 entonces
  Inicia (Si)
  SIG <--- Q
  Mensaje: "ANOTO LA FECHA CORRECTAMENTE "
  Escribe la fecha
  fin (si)
sino
  Inicia (sino)
  Marca error y la solicita nuevamente
  SIG <--- ' '
  fin (sino)
Hasta que SIG sea igual a 'Q'
```

FIN (FECHITA)

III.2.2.6 DOCUMENTACION DEL MODULO: BAJA

NOMBRE EN EL MENU: DAR DE BAJA ALGUN ARTICULO

PROGRAMA DONDE SE LOCALIZA: MOD5

1. OBJETIVO: Dar de baja del Sistema algún medicamento que no sea costeable.

2. FUNCIONES PRINCIPALES:

- Dar de baja algún medicamento que ya no se desee tener dentro del Sistema.
- Listar claves disponibles.

3. NARRATIVA DEL MODULO:

Si el archivo Maestro (BF) no contiene información, el programa le desplegará un mensaje de error, indicándole que el archivo se encuentra vacío y que necesita activar el módulo ESCRIBE (opción F) para dar de alta algún medicamento. En caso que el archivo Maestro contenga información, el programa le pedirá transmita la clave del medicamento que desea dar de baja del Sistema, el programa valida la clave transmitida y si es correcta la pone disponible, sino le despliega un mensaje de error y le vuelve a pedir de nuevo, transmita la clave del medicamento. Este proceso se repite hasta que usted lo desee.

Al final el programa le pregunta si desea listar las claves que se encuentran disponibles, en caso que si lo desee se listan las claves.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO
NC	CVEDISP.FIL	ARCHIVO DE CLAVES DISPONIBLES

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
X1	Apuntador para acceder el archivo de claves disponibles, variable de tipo entero.
LIM	Número de la clave máxima que contiene información en el Sistema, variable de tipo entero.
II	Apuntador para acceder el archivo de Nom. y Est. del medicamento, variable de tipo entero.
J	Indice para recorrer el archivo de claves disponibles, variable de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras al archivo Maestro, tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CMB	Clave del medicamento que se desea dar de baja del Sistema.

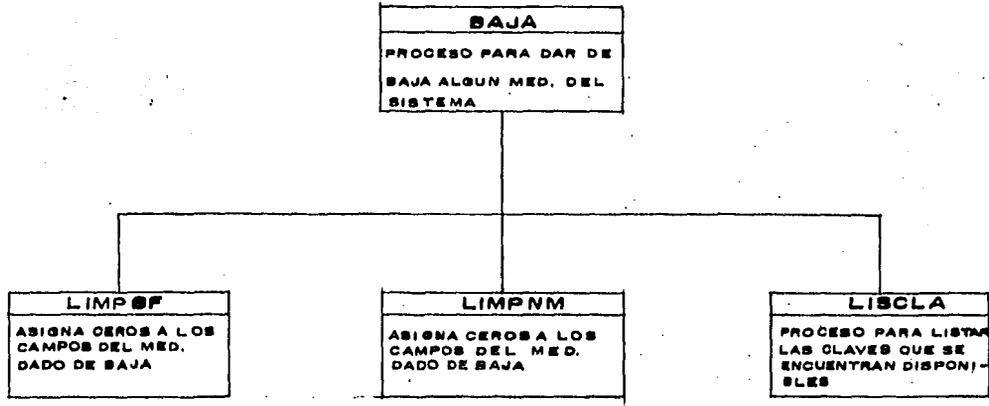
5.3 VARIABLES LOCALES:

NOMBRE	DESCRIPCION
NCD	Número de claves disponibles, variable de tipo entero.
TER	Bandera para terminar el programa, variable de tipo caracter.
REP	Bandera para detectar si la clave transmitida ya está repetida, variable de tipo caracter.
SNN	Respuesta de si o no, variable de tipo caracter.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

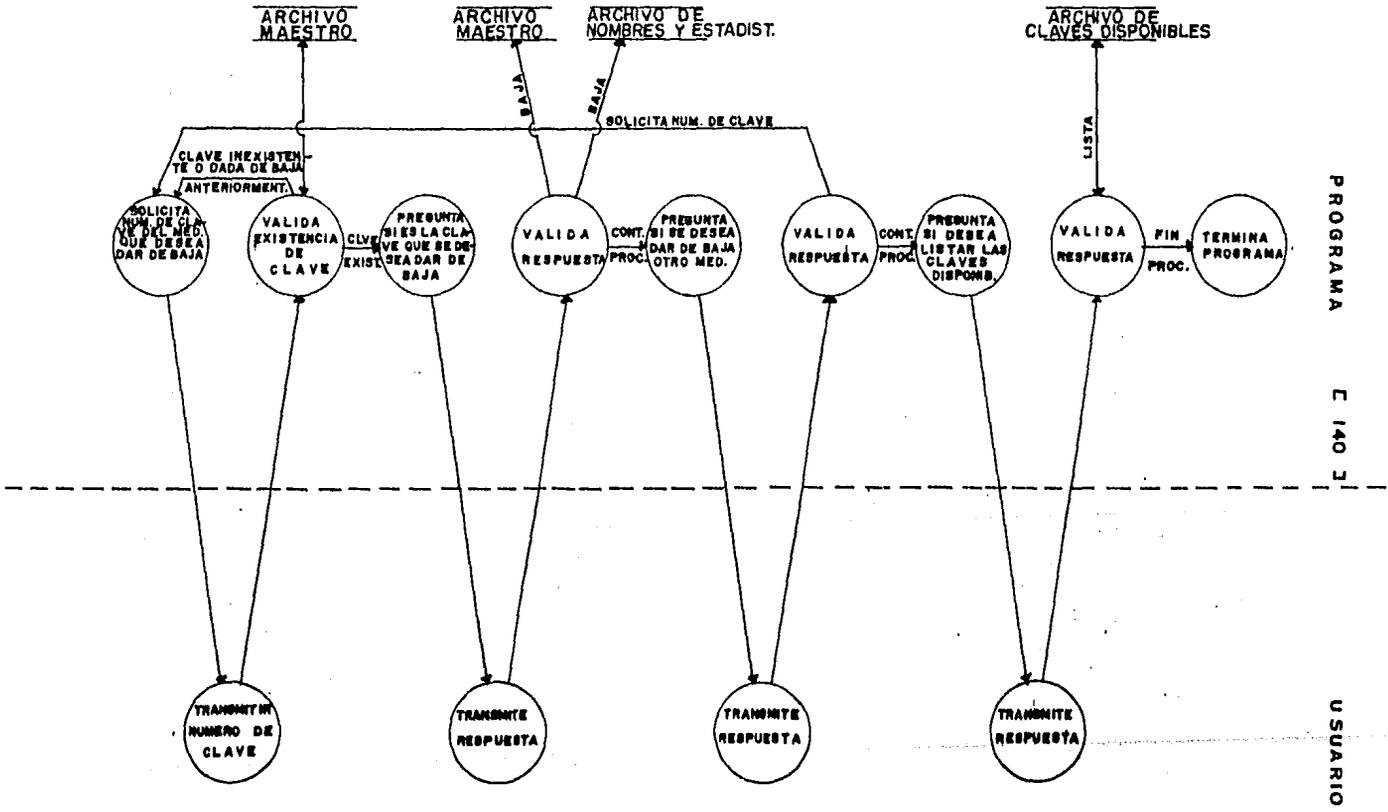
NOMBRE	OBJETIVO
LIMPF	Asignar ceros a todos los campos del registro del medicamento que se desea dar de baja, en el archivo Maestro.
LIMPNM	Asignar con ceros y asteriscos los campos del registro del medicamento que se desea dar de baja, en el archivo de Nom. y Est. del medicamento.
LISCLA	Listar claves disponibles.
ERRCHK	Valida las lecturas y las escrituras hechas a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Finaliza la sesión.

DIAGRAMA JERARQUICO



E 139 7

DIAGRAMA DE FLUJO DE DATOS



PSEUDOCODIGO

(* Inicia programa de BAJA *)

INICIA (BAJA)

Asigna el archivo de Claves Disponibles (NC)

X1 <-- 0

Lee el archivo de Claves Disponibles

Valida lectura

(* Asigna el número de claves que se encuentran en el archivo de NC *)

Si LIM es diferente de 0 entonces

inicia (Si)

Repite

Repite

TER <-- ' '

Despliega pregunta

'DEME LA CLAVE DEL ARTICULO QUE DESEA DAR DE BAJA => '

Lee el dato de la clave del medicamento (CMB)

Si CMB es mayor que 0 entonces

inicia (Si)

Si CMB es menor o igual a LIM entonces

inicia (Si)

Asigna el valor de CMB a I, II y J

REP <-- ' '

Haz desde J=1 hasta NCD lo siguiente

inicia (Haz)

X1 <-- J

Lee archivo de claves Disponibles

Si NC.CDISP es igual a NCD entonces

REP <-- '*'

fin (Haz)

Si REP es igual a ' ' entonces

inicia (Si)

Despliega pregunta

'ES CORRECTO EL MEDICAMENTO'

Lee respuesta (SNN)

Si SNN es igual a 'S' o a 's' entonces

inicia (Si)

NCD <-- NCD + 1

Activa LIMPBF

Activa LIMPNN

X1 <-- NCD

NC.CDISP <-- CMB

Escribe el dato en el archivo de Claves Disponibles

fin (Si)

fin (Si)

Sino

inicia (Sino)

TER <-- '*'

Despliega mensaje de error

'CLAVE PUESTA DISPONIBLE YA ANTERIORMENTE'

fin (Sino)

fin (Si)

```

Sino
  Despliega mensaje de error por la clave transmitida
fin (Si)
Sino
  inicia (Sino)
    Despliega mensaje de error
    'ERROR LAS CLAVES INICIAN DESDE LA 00001'
    TER <-- '*'
  fin (Sino)
Hasta que TER sea igual a un espacio
Despliega pregunta
'DESEAR DAR DE BAJA A OTRO ARTICULO, TRANSMITA S)I O N)O =>'
Lee dato transmitido
Valida dato
Hasta que TER sea igual a ' '
fin (Si)
Sino
  Despliega mensaje de que el archivo Maestro se encuentra vacío
  XI <-- 0
  NC.CDISP <-- NCD
  Escribe en el registro cero del archivo de Claves Disponibles,
  el numero
  claves disponibles
  Valida escritura
  Cierra archivo de Claves Disponibles
  Si LIM es diferente de 0 entonces
    Activa LISCLA
  Despliega mensaje de terminación de programa
FIN (BAJA)

```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: LIMPBF

PROGRAMA DONDE SE LOCALIZA: MOD5

OBJETIVO: Asignar ceros a todos los campos del registro del medicamento que se desea dar de baja del archivo Maestro.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CMB	Clave del medicamento que se desea dar de baja del Sistema, variable de tipo entero.

VARIABLES LOCALES: Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento de LIMPBF *)

INICIA (LIMPBF)

```
I <-- CMB
Lee archivo Maestro
Valida lectura
Asigna ceros a los campos del registro que se desea dar de baja
Reescribe los datos en el registro
Valida reescritura
```

FIN (LIMPBF)

10.2 NOMBRE DEL PROCEDIMIENTO: LIMPNM.

PROGRAMA DONDE SE LOCALIZA: MOD5

OBJETIVO: Asignar ceros y asteriscos a los campos del registro del medicamento que se desea dar de baja dentro del archivo de Nom. y Est. del medic.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
II	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento, variable de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CMB	Clave del medicamento que se desea dar de baja, variable de tipo entero.

VARIABLES LOCALES: Ninsuna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento de LIMPNN *)

INICIA (LIMPNN)

II <-- CMB

Lee archivo de Nom. y Est. del ned.

Asigna ceros y asteriscos a los campos del registro
que se desea dar de baja

Reescribe el registro con ceros

Valida reescritura

FIN (LIMPNN)

10.3 NOMBRE DEL PROCEDIMIENTO: LISCLA

PROGRAMA DONDE SE LOCALIZA: MOD5

OBJETIVO: Listar las claves disponibles

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NC	CVEDISP.FIL ARCHIVO DE CLAVES DISPONIBLES

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
X1	Apuntador para acceder el archivo de claves disponibles, variable de tipo entero.

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
NCA	Número de claves disponibles almacenadas dentro del archivo de Claves Disponibles, variable de tipo entero.
ICD	Indice para recorrer el archivo de claves disponibles, variable de tipo entero.
RSN	Respuesta de si o no, variable de tipo caracter.

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento de LISCLA *)

INICIA (LISCLA)

```
Despliega mensaje
'DESEA LISTAR LAS CLAVES QUE SE ENCUENTRAN'
'DISPONIBLES, TRANSMITA S)I O N)O =>'
Lee respuesta
Asigna el archivo de Claves Disponibles
Si RSN es igual a 'S' o a 's' entonces
  inicia (Si)
  XI <-- 0
  Lee archivo de Claves Disponibles
  NCA <-- número de claves disponibles
  Si NCA es mayor que 0 entonces
    inicia (Si)
    Lista claves disponibles
  fin (Si)
Sino
  Despliega mensaje
  'NO HAY CLAVES DISPONIBLES'
fin (Si)
```

FIN (LISCLA)

III.2.2.7 DOCUMENTACION DEL MODULO: ESCRIBE.

PROGRAMA DONDE SE LOCALIZA: MOD6

1. OBJETIVO: Dar de alta algún artículo en los archivos BF y NM.

2. FUNCIONES PRINCIPALES:

- Asignar los archivos Maestro BF y el de Nombres y Estadísticas NM, al módulo.
- Guardar en el registro correspondiente, la información proporcionada de cada medicamento, la cual se usa en el sistema.
- Validar la información.

3. NARRATIVA DEL MODULO:

Asigna los archivos BF y NM al módulo, para que estén listos a recibir información. Solicita la fecha del día de trabajo para validarla y la guarda en el registro cero. A continuación empieza a solicitar la información de cada medicamento, para colocarla en el registro correspondiente de acuerdo con la clave de cada uno de ellos. Valida la clave para llevar un orden estricto en los registros.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	MOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADÍSTICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Clave máxima contenida en el sistema. Es de tipo entero.
AÑO	Año de la fecha correspondiente al día de trabajo. Es de tipo entero.
MES	Mes de la fecha correspondiente al día de trabajo. Es de tipo entero.
DÍA	Día de la fecha correspondiente al día de trabajo. Es de tipo entero.
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
C	Clave del medicamento. Es de tipo entero.
NOMB	Nombre del medicamento. Es una cadena de 30 caracteres.
PR01	Clave del primer proveedor o proveedor principal. Es de tipo entero.
PR02	Clave del segundo proveedor. Es de tipo entero.

PRO3 Clave del tercer proveedor. Es de tipo entero.

PR1 Precio de compra del medicamento, del primer proveedor. Es de tipo real.

PR2 Precio de compra del medicamento, del segundo proveedor. Es de tipo real.

PR3 Precio de compra del medicamento, del tercer proveedor. Es de tipo real.

D1 Descuento del medicamento del primer proveedor. Es de tipo real.

D2 Descuento del medicamento del segundo proveedor. Es de tipo real.

D3 Descuento del medicamento del tercer proveedor. Es de tipo real.

MES6 Mes de la fecha de trabajo; fecha de caducidad. Es de tipo entero.

DIA6 Día de la fecha de trabajo; fecha de caducidad. Es de tipo entero.

ANO Año de la fecha en que se da de alta el medicamento; fecha de caducidad. Es de tipo entero.

II Apuntador para acceder al archivo NM. Es de tipo entero.

I Apuntador para acceder al archivo BF.

M1 Clave de la medicina similar 1. Es de tipo entero.

M2 Clave de la medicina similar 2. Es de tipo entero.

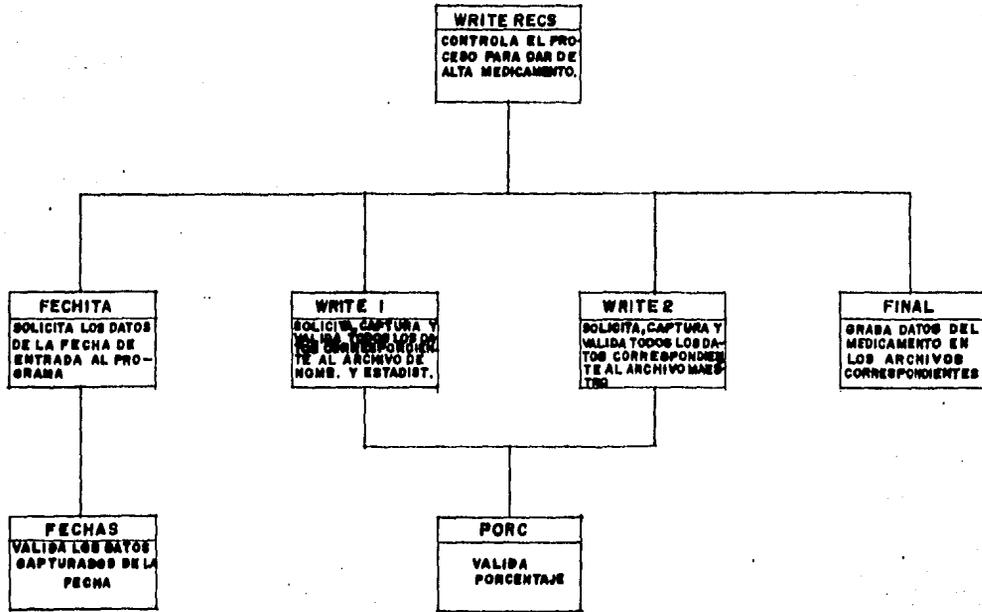
M3	Clave de la medicina similar 3. Es de tipo entero.
P	Precio de venta del medicamento. Es de tipo real.
U	Número de unidades en existencia. Es de tipo entero.
IM	Impuesto del medicamento. Es de tipo real.
Y1	Porcentaje de unidades mínimas que deben existir. Es de tipo real.
SIG	Variable de control; puede asumir valores con S, N, O. Es de tipo caracter.

5.3 VARIABLES LOCALES: Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
WRITE1	Guardar la información del medicamento en el archivo NM.
WRITE2	Guardar la información del medicamento en el archivo BF.
FINAL	Grabar información trascendental del sistema en el registro de alta seguridad.
PORC	Obtener el porcentaje en forma fraccionaria o decimal.
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Da por terminada la sesión.

DIAGRAMA JERARQUICO

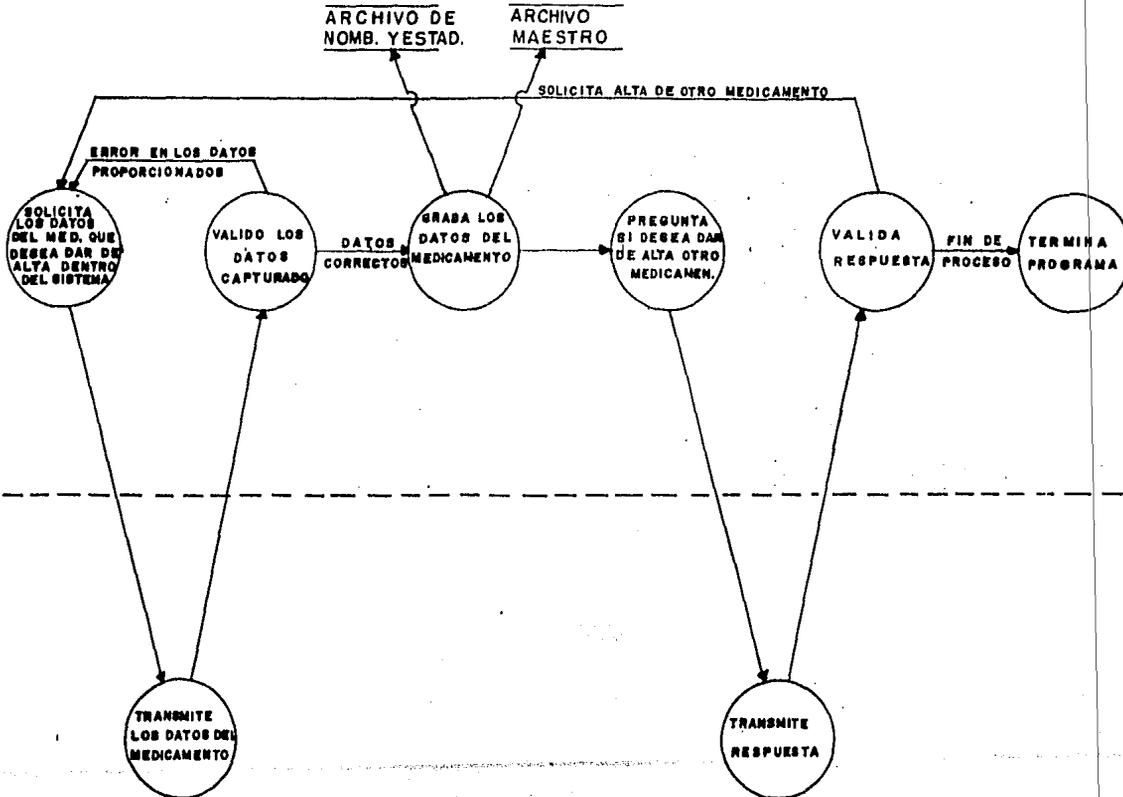


C 181 7

DIAGRAMA DE FLUJO DE DATOS

ARCHIVO DE
NOMB. Y ESTAD.

ARCHIVO
MAESTRO



PROGRAMA

C 152 J

USUARIO

9. PSEUDOCODIGO.

(* Inicia el módulo WRITEREC *)

INICIA (WRITERECS)

Asigna el Archivo BF
Asigna el Archivo NM

VFE <--- 1

Activa FECHITA

VFE <--- 0

SIG <--- ' '

Repite

N2 <--- N2 + 1

Despliega el mensaje:

"ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION !"

(* se inicia la solicitud de los datos del medicamento,
uno a uno, para llenar los campos correspondientes,
desplegando el mensaje del dato y leyendo la varia-
ble correspondiente *)

Mensaje: "CLAVE DEL MEDICAMENTO =>"

Lee C

Si C es igual a N2

Inicia (Si)

(* la clave es válida *)

(* Recibe la primera parte de datos *)

Activa WRITE1

(* Recibe la segunda parte de datos *)

Activa WRITE2

fin (Si)

sino

Inicia (sino)

(* la clave está equivocada *)

Despliega mensaje de error

N2 <--- N2 - 1

fin (sino)

Despliega mensaje:

"SI DESEA SEGUIR METIENDO INFORMACION TRANSMITA <RETURN>

EN CASO CONTRARIO ANOTE Q =>"

Hasta que SIG sea igual a 'Q'

Activa FINAL

Cierra archivos

FIN (WRITERECS)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: WRITE1

PROGRAMA DONDE SE LOCALIZA: MOD6

OBJETIVO: Guardar la información del medicamento en el archivo NM.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
C	Clave del medicamento. Es de tipo entero.
NOMB	Nombre del medicamento. Es una cadena de 30 caracteres.
PRO1	Clave del primer proveedor o proveedor principal. Es de tipo entero.

PRO2 Clave del segundo proveedor. Es de tipo entero.

PRO3 Clave del tercer proveedor. Es de tipo entero.

PR1 Precio de compra del medicamento, del primer proveedor. Es de tipo real.

PR2 Precio de compra del medicamento, del segundo proveedor. Es de tipo real.

PR3 Precio de compra del medicamento, del tercer proveedor. Es de tipo real.

D1 Descuento del medicamento del primer proveedor. Es de tipo real.

D2 Descuento del medicamento del segundo proveedor. Es de tipo real.

D3 Descuento del medicamento del tercer proveedor. Es de tipo real.

ANO Año de la fecha de trabajo. Es de tipo entero.

MES Mes de la fecha de trabajo. Es de tipo entero.

DIA6 Día de la fecha de trabajo. Es de tipo entero.

II Apuntador para acceder al archivo NM. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO:
FORC	Obtener el porcentaje en forma fraccionaria o decimal.
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Da por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento WRITE1 *)

INICIA (WRITE1)

```

NM.CLAVE <--- C
Solicita el NOMBRE DEL MEDICAMENTO
Lee NOMB
NM.NOMBRE <--- NOMB
(* inicializa ventas por temporada en ese registro *)
NM.R1 <--- 0
NM.R2 <--- 0
NM.R3 <--- 0
Solicita CLAVE DEL PROVEEDOR PRINCIPAL
Lee PR01
NM.PROV1 <--- PR01
Solicita PRECIO DE COMPRA DEL PROVEEDOR PRINCIPAL
Lee PR1
NM.PRE1 <--- PR1
Solicita DESCUENTO DE COMPRA DEL PROVEEDOR PRINCIPAL
Lee D1
Activa PORC
NM.DP1 <--- D1
Solicita CLAVE DEL SEGUNDO PROVEEDOR
Lee PR02
NM.PROV1 <--- PR02
Solicita PRECIO DE COMPRA DEL SEGUNDO PROVEEDOR
Lee PR2
NM.PRE2 <--- PR2
Solicita DESCUENTO DE COMPRA DEL SEGUNDO PROVEEDOR
Lee D2
Activa PORC
NM.DP2 <--- D2
    
```

Solicita CLAVE DEL TERCER PROVEEDOR
Lee PRO3
NM.PROV3 <---PRO3
Solicita PRECIO DE COMPRA DEL TERCER PROVEEDOR
Lee PR3
NM.PRE3 <--- PR3
Solicita DESCUENTO DE COMPRA DEL TERCER PROVEEDOR .
Lee D3
Activa PORC
NM.DP3 <--- D3
(* asigna la fecha de esta sesión de trabajo *)
NM.DD <--- DIA
NM.MM <--- MES
NM.AA <--- ANIO
II <--- C
Escribe en el registro II del archivo NM
Activa ERRCHK
Si ERROR es verdadero entonces
Activa EXIT
fin (Si)

FIN (WRITE1)

10.2 NOMBRE DEL PROCEDIMIENTO: WRITE2

PROGRAMA DONDE SE LOCALIZA: MOD6

OBJETIVO: Guardar la información del medicamento en el archivo BF.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras al archivo. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
C	Clave del medicamento. Es de tipo entero.
I	Apuntador para acceder al archivo BF.
M1	Clave de la medicina similar 1. Es de tipo entero.
M2	Clave de la medicina similar 2. Es de tipo entero.
M3	Clave de la medicina similar 3. Es de tipo entero.
P	Precio de venta del medicamento. Es de tipo real.
U	Número de unidades en existencia. Es de tipo entero.
ANO	Año de la fecha de caducidad. Es de tipo entero.
MES6	Mes de la fecha de caducidad. Es de tipo entero.
DIA6	Día de la fecha de caducidad. Es de tipo entero.
PRO1	Clave del primer proveedor o proveedor principal. Es de tipo entero.
IM	Impuesto del medicamento. Es de tipo real.
Y1	Porcentaje de unidades mínimas que deben existir. Es de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO:
PORC	Obtener el porcentaje en forma fraccionaria o decimal.
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Da por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento WRITE2 *)

INICIA (WRITE2)

Solicita CLAVE DE MEDICINA SIMILAR 1
Lee M1
BF.MS1 <--- M1
Solicita CLAVE DE MEDICINA SIMILAR 2
Lee M2
BF.MS2 <--- M2
Solicita CLAVE DE MEDICINA SIMILAR 3
Lee M3
BF.MS3 <--- M3
Solicita EL PRECIO DE VENTA DEL MEDICAMENTO
Lee P
BF.PRECIO <--- P
Solicita EL NUMERO DE UNIDADES EN EXISTENCIA
Lee U
BF.UNID <--- U
Solicita FECHA DE CADUCIDAD DEL MEDICAMENTO
Activa FECHITA
BF.ANN <--- ANO
BF.MES <--- MES6
BF.DIA <--- DIA6
(* asigna la clave del primer proveedor a proveedor principal *)
BF.CP <--- PRO1

```
Solicita EL IMPUESTO
Lee IM
Activa PORC
BF.IMP <--- IM
Solicita el PORCENTAJE DE UNIDADES MINIMAS QUE DEBEN EXISTIR
Lee Y1
Activa PORC
BF.PDU <--- Y1
(* inicializa campos *)
BF.UVEN <--- 0
BF.UPED <--- 0
BF.UVP1 <--- 0
(* inicializa la fecha de caducidad de pedidos posteriores *)
BF.AA <--- 0
BF.MM <--- 0
BF.DD <--- 0
BF.CLAVE <--- C
Comunica que ya recibió todos los datos del medicamento
I <--- C
Escribe todo en el registro I del archivo BF
Activa ERRCHK
  Si ERROR es verdadero entonces
    Activa EXIT
```

FIN (WRITE2)

10.3 NOMBRE DEL PROCEDIMIENTO: PORC

PROGRAMA DONDE SE LOCALIZA: MOD6

OBJETIVO: Obtener el porcentaje en forma fraccionaria o decimal.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras al archivo. Es de tipo booleano.

VARIABLES MODULARES: Ninguna.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
TTT	Parámetro para redresar el valor del porcentaje decimal. Es de tipo real.
S	Variable de control para validar el impuesto; si vale 1 significa que el dato está correcto. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el Procedimiento PORC *)

INICIA (PORC)

```
Repite
  S <--- 1
  Si TTT es mayor e igual que 1 entonces
    Inicia (Si)
    Si TTT es mayor e igual que 1 y
      además es menor que 100 entonces
        TTT <--- TTT/100
      sino
        Inicia (sino)
        S <--- 0
        Despliega mensaje de error
        TTT <--- 0
        Solicita el dato nuevamente
        Lee TTT
      fin (sino)
    fin (si)
Hasta que S sea igual a 1
```

FIN (PORC)

10.4 NOMBRE DEL PROCEDIMIENTO: FINAL

PROGRAMA DONDE SE LOCALIZA: MOD6

OBJETIVO: Guardar información confidencial en los archivos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras al archivo. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
D1	Descuento del medicamento del primer proveedor. Es de tipo real.
D2	Descuento del medicamento del segundo proveedor. Es de tipo real.
ANIO	Año de la fecha de trabajo. Es de tipo entero.
MES	Mes de la fecha de trabajo. Es de tipo entero.

DIA Día de la fecha de trabajo. Es de tipo entero.

II Apuntador para acceder al archivo NM. Es de tipo entero.

I Apuntador para acceder al archivo BF. Es de tipo entero.

N2 Clave máxima contenida en los archivos. Es de tipo entero.

NOMB Nombre del medicamento. Es una cadena de 30 caracteres.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento FINAL *)

INICIA (FINAL)

```

I <--- 0
BF.MS1 <--- N2
BF.IMP <--- D1
BF.PDU <--- D2
Escribe en el registro I del archivo BF
Activa ERRCHK
  Si ERROR es verdadero entonces
    activa EXIT
  fin (Si)
II <--- 0
NM.CLAVE <--- N2
NM.NOMBRE <--- NOMB
NM.DD <--- DIA
NM.MM <--- MES
NM.AA <--- ANIO
Escribe en el registro II del archivo NM
Activa ERRCHK
  Si ERROR es verdadero entonces
    activa EXIT
  fin (Si)

```

FIN (FINAL)

10.8 NOMBRE DEL PROCEDIMIENTO: FECHITA

PROGRAMA DONDE SE LOCALIZA: MOD4

OBJETIVO: Lee los datos de la fecha y la valida.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
ANO	Contiene el valor del año de la fecha a ser validada. Es de tipo entero.
MES6	Contiene el valor del mes de la fecha a ser validada. Es de tipo entero.
DIA6	Contiene el valor del día de la fecha a ser validada. Es de tipo entero.
VER	Si la fecha es válida toma el valor 1; de lo contrario vale 0. Es de tipo entero.
SIG	Variable usada para controlar el proceso de validación. Es de tipo caracter.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
AN1	Sirve para ver si un año es bisiesto. Es de tipo entero.
DI2	Número de días que tiene el mes. Es de tipo entero.

Si VER es igual a 1 entonces

Inicia (Si)

SIG <--- Q

Mensaje: 'ANOTO LA FECHA CORRECTAMENTE '

Escribe la fecha

fin (si)

sino

Inicia (sino)

Marca error y la solicita nuevamente

SIG <--- ' '

fin (sino)

Hasta que SIG sea igual a 'Q'

FIN (FECHITA)

III.2.2.8 DOCUMENTACION DEL MODULO LISTAR

NOMBRE EN EL MENU: LISTAR LA INFORMACION

PROGRAMA DONDE SE LOCALIZA: MOD7

1. OBJETIVO: Listar la información más relevante de los medicamentos que se encuentran dentro de los archivos Maestro y Nombre y Estadísticas del Medicamento.

2. FUNCIONES PRINCIPALES:
 - Listar medicamento por medicamento.
 - Listar los medicamentos de una clave inicial a otra clave final.
 - Listar todos los medicamentos contenidos dentro del Sistema.

3. NARRATIVA DEL PROGRAMA:

Si el archivo Maestro (BF) no contiene información, el programa le desplegará un mensaje de error, indicándole que el archivo se encuentra vacío y que necesita activar el módulo ESCRIBE (opción F) para dar de alta algún medicamento. En caso que el archivo Maestro contenga información, el programa le preguntará de que manera desea, se le liste la información de los medicamentos que requiere consultar. El programa le proporcionará tres alternativas, para que usted liste la información deseada; una será, la de listar uno a uno los medicamentos que usted desee consultar, otra será la de listar medicamentos desde una clave inicial hasta una clave final, y por último usted puede listar todos los medicamentos que se encuentran dentro del Sistema. Este proceso se repite hasta que usted lo desee.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras al archivo Maestro, tipo booleano.

5.2 VARIABLES MODULARES: Ninguna

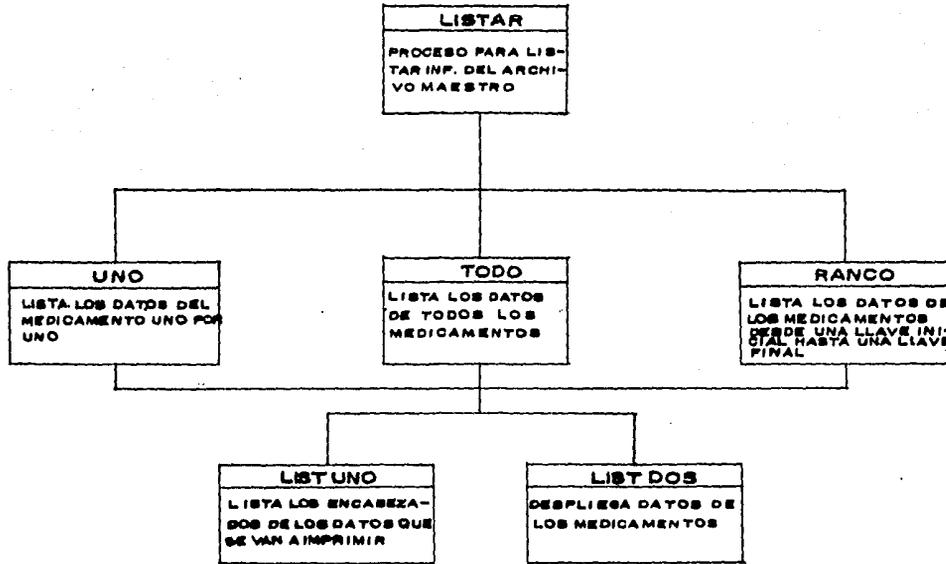
5.3 VARIABLES LOCALES:

NOMBRE	DESCRIPCION
TIPO	Tipo de opción para listar información, variable de tipo entero.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
UNO	Listar uno a uno la información de los medicamentos que se desean consultar.
RANGO	Listar medicamentos desde una clave inicial hasta una clave final.
TODO	Listar todos los medicamentos del Sistema.
ERRCHK	Valida las lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Finaliza la sesión.

DIAGRAMA JERARQUICO



9. PSEUDOCODIGO

(* Inicia Programa LISTAR *)

INICIA (LISTAR)

Si LIM es diferente de 0 entonces
inicia (Si)

Repite

Repite

Despliega pregunta

'COMO DESEA LISTAR SU INFORMACION? '

'DE U)NO A UNO, T)ODA, O DE UN R)ANGO A OTRO => '

Lee respuesta

En caso que TIPO

Sea igual a 'U' o a 'u' Activa UNO

ó

Sea igual a 'T' o a 't' Activa TODO

ó

Sea igual a 'R' o a 'r' Activa RANGO

ó

Sea diferente de 'U', 'T' o 'R' Despliega mensaje de error
fin (caso)

Hasta que TIPO sea diferente de '*'

TIPO <-- ' '

Despliega pregunta

'DESEA CONTINUAR LISTANDO INFORMACION?'

'TRANSMITA S)I O N)O => '

Lee dato

Valida dato

Hasta que TIPO sea igual a '*'

fin (Si)

Sino

Despliega mensaje de archivo vacío

Despliega mensaje de terminación de programa

FIN (LISTAR)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: UNO

PROGRAMA DONDE SE LOCALIZA: MOD7

OBJETIVO: Listar uno a uno la información del medicamento que se desea consultar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
SIG	Bandera para continuar el proceso de listar, variable de tipo caracter.
BAND	Bandera para detectar algún error en la transmisión del dato variable de tipo caracter.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
LISTUNO	Listar el encabezado de la información que se desea listar.
LISTDOS	Listar datos del medicamento.

PSEUDOCODIGO

(* Inicia Procedimiento UNO *)

INICIA (UNO)

```

SIG <-- ' '
Repite
  Despliega mensaje
  'TRANSMITA LA CLAVE DEL MEDICAMENTO, QUE DESEA CONSULTAR =>'
  BAND <-- ' '
  Repite
    Lee clave del registro que se desea listar
    Si I es mayor o igual a 0 e I es menor o igual a LIM entonces
      inicia (Si)
        Lee archivo Maestro
        Valida lectura
        BAND <-- '*'
        fin (Si)
    Sino
      Despliega mensaje de error en el dato transmitido
      y vuelve a solicitar de nuevo el dato
  Hasta que BAND sea igual a '*'
  Imprime línea de encabezado ( LISTUNO )
  Imprime línea de datos del medicamento ( LISTDOS )
  Despliega mensaje
  'DESEA CONSULTAR LA INFORMACION DE OTRO MEDICAMENTO?,'
  'TRANSMITA S)I O N)O => '
  Lee respuesta
  Hasta que SIG sea igual a 'N'

```

FIN (UNO)

10.2 NOMBRE DEL PROCEDIMIENTO: RANG0

PROGRAMA DONDE SE LOCALIZA: MOD7

OBJETIVO: Listar medicamentos desde una clave inicial hasta una clave final.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
BAND1	Bandera para validar las claves de inicio y final, variable de tipo caracter.
CVEI	Clave de inicio para listar medicamentos, variable de tipo entero.
CVEF	Clave final para listar medicamentos, variable de tipo entero.
BAND2	Bandera para terminar proceso, variable de tipo entero.
J	Número de líneas por página para ser listadas, variable de tipo entero.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
LISTUNO	Listar el encabezado de la información que se desea listar.
LISTDOS	Listar la información que se requiere.

PSEUDOCODIGO

(* Inicia procedimiento RANGO *)

INICIA (RANGO)

```

BAND1 <-- ' '
Mientras BAND1 sea igual a ' ' Haz
  inicia (Mientras)
    Despliega pregunta
    'DEME CLAVE DE INICIO => '
    Lee clave de inicio
    Si CVEI es menor o igual a 0 entonces
      despliega mensaje
      'ERROR: LAS CLAVES INICIAN DESDE LA 00001 '
    Sino
      Si CVEI es mayor que LIM entonces
        despliega mensaje de error
      Sino
        BAND1 <-- '*'
  fin (Mientras)
BAND1 <-- ' '
Mientras BAND1 sea igual a ' ' Haz
  inicia (Mientras)
    Despliega mensaje
    'DEME CLAVE FINAL => '
    Lee clave final
    (* Valida clave final *)
    Si CVEF es mayor o igual a CVEI entonces
      Si CVEF es menor o igual a LIM entonces
        BAND1 <-- '*'
      Sino
        Despliega mensaje de error
    Sino
      Despliega mensaje
      'ERROR: TRANSMITA UNA CLAVE MAYOR O IGUAL A LA INICIAL '
  fin (Mientras)
I <-- CVEI - 1
Repite
  Imprime linea de encabezado ( LISTUNO )
  J <-- 0
  BAND2 <-- ' '
  Repite
    I <-- I + 1
    J <-- J + 1
    Si I es menor o igual a CVEF entonces
      inicia (Si)
        Lee archivo Maestro
        Valida lectura
        Imprime linea de datos del medicamento ( LISTDOS )
      fin (Si)
    Sino
      BAND2 <-- '*'
  Hasta que J = 20 o que BAND2 = '*'
  Hasta que BAND2 sea igual a '*'

```

FIN (RANGO)

10.3 NOMBRE DEL PROCEDIMIENTO: TODO

PROGRAMA DONDE SE LOCALIZA: MOD7

OBJETIVO: Listar todos los medicamentos del Sistema.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
J	Número de líneas que debe listar por página, variable de tipo entero.
BAND	Bandera para continuar el proceso de listar, variable de tipo caracter.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
LISTUNO	Listar el encabezado de la información que se desea.
LISTDOS	Listar la información que se requiere.

PSEUDOCODIGO

(* Inicia procedimiento TODO *)

INICIA (TODO)

```

I <-- 0
Repite
  Imprime linea de encabezado ( LISTUNO )
  J <-- 0
  BAND <-- ' '
  Repite
    I <-- I + 1
    J <-- J + 1
    (* Lista todos los medicamentos *)
    Si I es menor o igual a LIM entonces
      inicia (Si)
        Lee el archivo Maestro (BF)
        Valida lectura
        Imprime linea de datos del medicamento ( LISTDOS )
      fin (Si)
    Sino
      BAND <-- '*'
  Hasta que J=20 o BAND='*'
Hasta que BAND = '*'

```

FIN (TODO)

10.1.1 NOMBRE DEL PROCEDIMIENTO: LISTUNO
10.2.1 NOMBRE DEL PROCEDIMIENTO: LISTUNO
10.3.1 NOMBRE DEL PROCEDIMIENTO: LISTUNO

PROGRAMA DONDE SE LOCALIZA: MOD7

OBJETIVO: Listar el encabezado de la información que se desea listar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES.

VARIABLES EXTERNAS: Ninguna

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES: Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento LISTUNO *)

INICIA (LISTUNO)

Despliega las líneas del encabezado,
de la información que se desea listar.

FIN (LISTUNO)

10.1.2 NOMBRE DEL PROCEDIMIENTO: LISTDOS
10.2.2 NOMBRE DEL PROCEDIMIENTO: LISTDOS
10.3.2 NOMBRE DEL PROCEDIMIENTO: LISTDOS

PROGRAMA DONDE SE LOCALIZA: MOD7

OBJETIVO: Listar la información que se requiere.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
E,E1	Espacios que se dejan para centrar la información de salida, estas variables son de tipo entero.
IMP	Impuesto de venta del medicamento, variable de tipo real.
PMUE	Porcentaje mínimo de unidades en existencia, variable de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento LISTDOS *)

INICIA (LISTDOS)

(* Calcula espacios para centrar la información de salida, tanto del *)

(* impuesto, como del porcentaje mínimo de unidades en existencia *)

Si IMP es menor a 10.0 entonces

E <-- 7

Sino

E <-- 6

Si PMUE es menor que 10.0 entonces

E1 <-- 6

Sino

E1 <-- 5

Despliega línea con la información requerida del medicamento.

FIN (LISTDOS)

III.2.2.9 DOCUMENTACION DEL MODULO: ACTUAL

NOMBRE EN EL MENU: ACTUALIZAR INFORMACION

PROGRAMA DONDE SE LOCALIZA: MOD8

1. OBJETIVO: Actualizar campos del registro de Medicamentos.

2. FUNCIONES PRINCIPALES:

- Despliega el nombre de los campos cuya información puede ser modificada.
- Solicita el número de la clave del medicamento, al cual se le desea modificar información.
- Solicita el número del campo del registro, del medicamento al que se le desea modificar información.
- Realiza la actualización del registro.
- Dar de baja claves, del archivo de Claves Disponibles.

3. NARRATIVA DEL PROGRAMA:

Al inicio del programa verifica si el archivo Maestro contiene información, si el archivo se encuentra vacío, despliega un mensaje de error, indicándole que debe listar el menú de las funciones del Subsistema y activar el módulo ESCRIBE opción <F>.

Si el archivo Maestro contiene información, despliega el nombre de los campos cuya información puede ser modificada, después el programa le solicita el número de la clave del medicamento, al cual se le desea modificar información, luego le solicita el número del campo del registro del medicamento y por último realiza la actualización del registro.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS
NC	CVEDISF.FIL	ARCHIVO DE CLAVES DISPONIBLES

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero
ERROR	Valida lecturas y escrituras de los archivos, variable de tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CL1	Primera clave de contraseña, variable de tipo real.
CL2	Segunda clave de contraseña, variable de tipo real.
I2	Apuntador para acceder el archivo Maestro, variable de tipo entero.
SIGZ	Bandera para detectar el fin del proceso, variable de tipo caracter.
COVE	Contador que sirve para determinar si se da de alta alguna clave que anteriormente ha sido puesta disponible.
SIGA	Bandera para realizar la actualización, variable de tipo caracter.
X	Número del campo que se desea actualizar, variable de tipo entero.

5.3 VARIABLES LOCALES: Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
ACTUA1	Despliega el menú de los campos que pueden ser modificados.
ACTUA3	Solicita y actualiza el registro con la nueva información.
CLAVBA	Da de alta claves del archivo de Claves Disponibles.
ERRCHK	Valida las lecturas y escrituras en los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Finaliza la sesión.

DIAGRAMA JERARQUICO

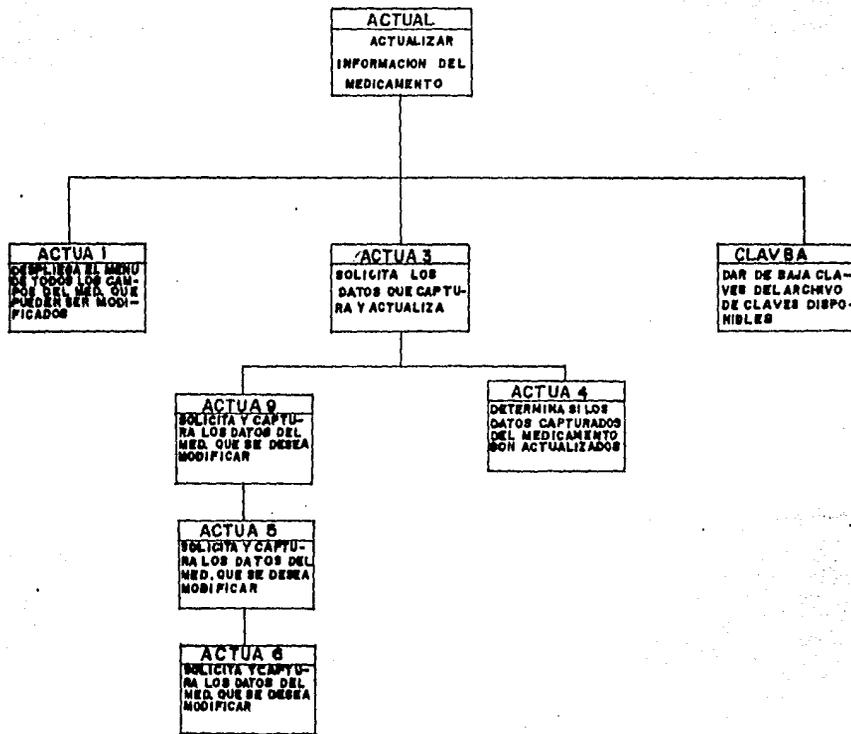


DIAGRAMA DE FLUJO DE DATOS

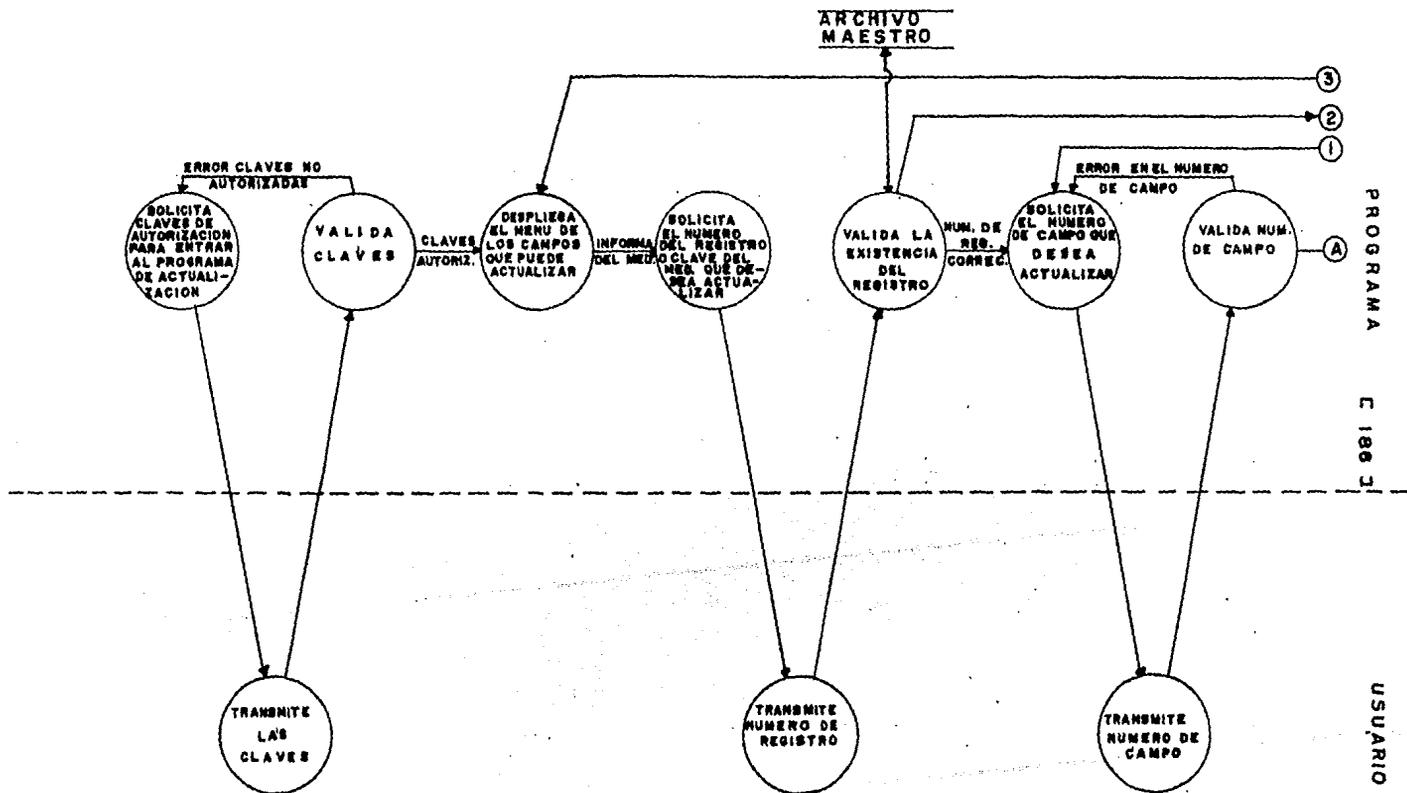
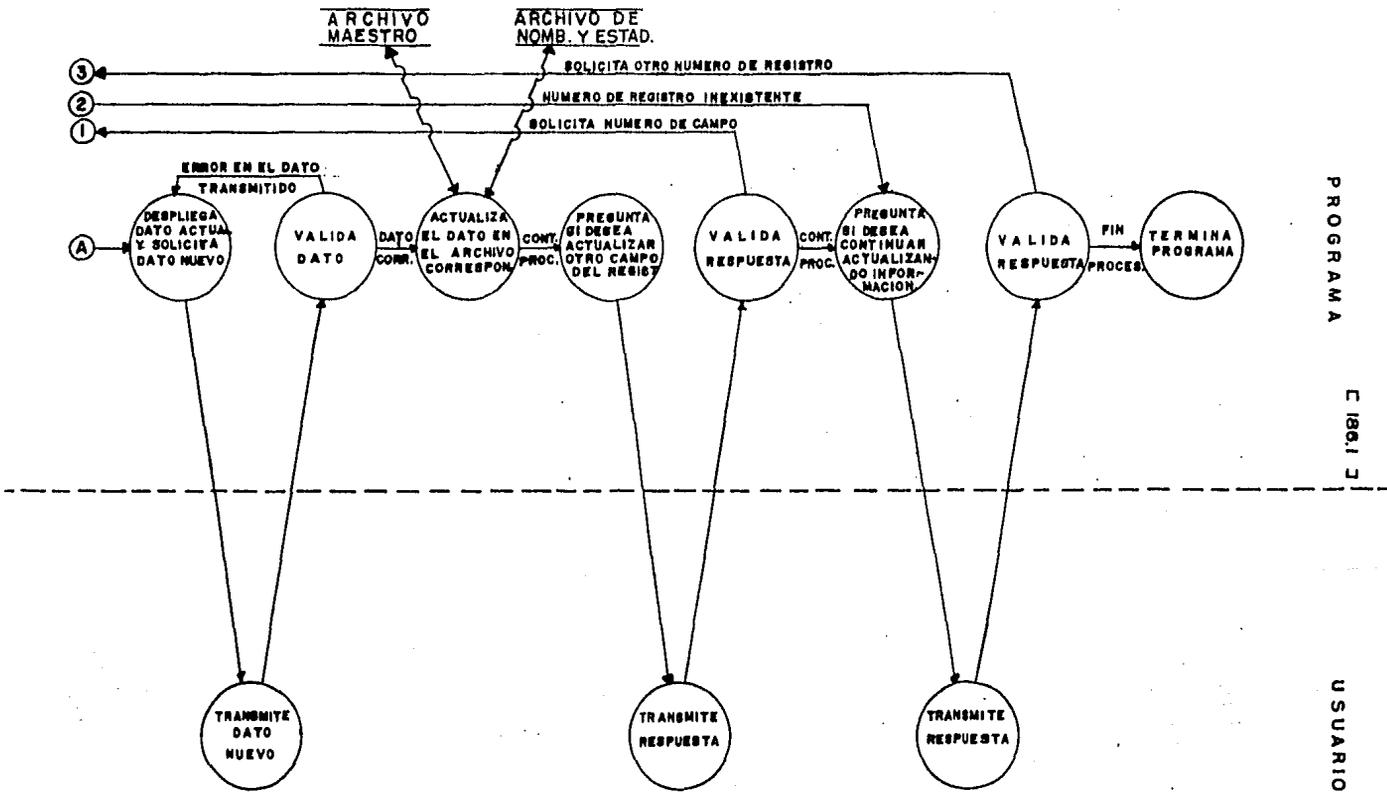


DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO.

(* Inicia procedimiento ACTUA1 *)

INICIA (ACTUAL)

```

Si LIM es mayor que 0 entonces
  inicia (Si)
  Asigna Archivo Maestro
  Asigna Archivo de Claves Disponibles
  Asigna Archivo de Nombres y Estadísticas
  Despliega mensaje
  'ANOTE LA PRIMERA CONTRASEÑA'
  Lee dato
  Despliega mensaje
  'ANOTE LA SEGUNDA CONTRASEÑA'
  Lee dato
  I2 <-- 0
  Lee archivo Maestro
  Valida lectura
  Si CL1 es igual a BF .IMP y CL2 es igual a BF .UVP1 entonces
    inicia (Si)
    SIGZ <-- ' '
    Repite
      Activa ACTUA1
      Despliega mensaje
      'CUAL REGISTRO QUIERE MODIFICAR? ANOTE EL NUMERO : '
      Lee dato
      Si I2 es mayor que cero e I2 es menor o igual a LIM entonces
        inicia (Si)
        COVE <-- 0
        Lee archivo Maestro
        Lee archivo de Nombres y Estadísticas
        Valida lecturas
        SIGA <-- 'N'
        Repite
          Despliega mensaje
          'CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : '
          Lee dato
          Si X es mayor que 0 ó menor o igual a 28 entonces
            activa ACTUA3
          Sino
            Despliega mensaje
            'NO EXISTE EL CAMPO, ESTA FUERA DE RANGO'
          Despliega mensaje
          'DESEA CAMBIAR OTRO CAMPO DEL REGISTRO? ANOTE S)I O N)O : '
          Lee dato
          Hasta que siga sea igual a 'N'
          Si COVE es igual a 2 entonces
            Activa CLAVBA
          fin (Si)
        Despliega mensaje
        'REGISTRO FUERA DE RANGO'
      Hasta que SIGZ sea igual a 'Q'
  
```

```
I2 <-- LIM
Cierra el archivo Maestro
Cierra el archivo de Nombres y Estadísticas
I2 <-- CONT
Cierra el Archivo de Claves Disponibles
I2 <-- 0
fin (Si)
Sino
  Despliega mensaje
  'USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES'
fin (Si)
Sino
  Despliega mensaje
  'Archivo maestro no contiene informacion'
FIN (ACTUAL)
```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: ACTUA1

PROGRAMA DONDE SE LOCALIZA: MOD8

OBJETIVO: Desplegar por pantalla el menú de todos los campos del medicamento, que pueden ser modificados.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno

VARIABLES. Ninguna

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO

(* Inicia procedimiento ACTUA1 *)

INICIA (ACTUA1)

Despliega por pantalla el menú de todos los campos del medicamento que pueden ser modificados

FIN (ACTUA1)

10.2 NOMBRE DEL PROCEDIMIENTO: ACTUA3

PROGRAMA DONDE SE LOCALIZA: MOD8

OBJETIVO: Solicitar y capturar los datos del medicamento, que se desea modificar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVOS MAESTRO

VARIABLES.

VARIABLES EXTERNAS. Ninguna.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
X	Número del campo que se desea actualizar, variable de tipo entero.
M11,M22,M33	Para lecturas de datos, variables de tipo entero.
PP	Para leer precios y descuentos, variable de tipo real.
COVE	Contador que es utilizado para determinar si se da de alta, alguna clave que anteriormente ha sido puesta disponible, variable de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ACTUA9	Solicitar y capturar los datos del medicamento que se desea modificar.
ACTUA4	Determinar si los datos capturados del medicamento son actualizados.

PSEUDOCODIGO.

(* Inicia procedimiento ACTUA3 *)

INICIA (ACTUA3)

```

Despliega mensaje
' DATO VIEJO '
Si X es menor que 7 entonces
  inicia (Si)
  En caso que X
    sea igual a 1: Desplegar el dato actual del Medicamento similar 1
                  Solicita el nuevo dato
                  Capturar y actualizar el dato
    ó
    sea igual a 2: Despliega el dato actual del Medicamento similar 2
                  Solicita el nuevo dato
                  Capturar y actualizar el dato
    ó
    sea igual a 3: Despliega el dato actual del Medicamento similar 3
                  Solicita el nuevo dato
                  Captura y actualiza el dato
    ó
    sea igual a 4: Si BF .PRECIO es igual a 0 entonces
                  COVE <-- COVE + 1
                  Despliega el dato actual del precio del Medicamento
                  Solicita el nuevo precio
                  Captura y actualiza el dato
    ó
    sea igual a 5: Despliega el dato actual de las unidades en exist.
                  Solicita el nuevo dato
                  Captura y actualiza el dato
    ó
    sea igual a 6: Despliega el dato actual de la fecha de caducidad
                  Solicita la nueva fecha
                  Captura y actualiza el dato

  fin (caso)
  fin (Si)
Sino
  Activa ACTUA9
  Despliega mensaje
  'ESTA CORRECTO ? ANOTE S)I O N)O :
FIN (ACTUA3)

```

10.2.1 NOMBRE DEL PROCEDIMIENTO: ACTUA9

PROGRAMA DONDE SE LOCALIZA: MOD8

OBJETIVO: Solicitar y capturar los datos del medicamento, que se desea modificar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
X	Número del campo que se desea actualizar, variable de tipo entero.
M11,M22,M33	Para lecturas de datos, variable de tipo entero.
PP	Para leer precios y descuentos, variable de tipo real.
COVE	Contador que se utiliza, para determinar si se da de alta, alguna de las claves que anteriormente ha sido puesta disponible, variable de tipo entero.

VARIABLES LOCALES:

PROCEDIMIENTOS QUE INVOCA:

NOMBRE OBJETIVO

ACTUA5 Solicitar y capturar los datos del medicamento;
 que se desea modificar.

PSEUDOCODIGO.

(* Inicia procedimiento ACTUA9 *)

INICIA (ACTUA9)

Si X es menor que 14 entonces
 inicia (Si)

 En caso que X

 sea igual a 7: Despliega el dato actual de la clave del Proveedor
 Solicita la nueva clave
 Captura y actualiza el dato

 ó
 sea igual a 8: Despliega el dato actual del impuesto que causa
 el medicamento
 Solicita el nuevo impuesto
 Captura y actualiza el dato

 ó
 sea igual a 9: Despliega el dato actual del porcentaje mínimo de
 unidades que debe haber en existencias
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó
 sea igual a 10: Despliega el dato actual del número de unidades
 vendidas
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó
 sea igual a 11: Despliega el dato actual del número de unidades
 pedidas
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó
 sea igual a 12: Despliega el dato actual del número de unidades
 vendidas de la nueva remesa
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó
 sea igual a 13: Despliega el dato actual de la fecha de caducidad
 de la nueva remesa
 Solicita el nuevo dato
 Captura y actualiza el dato

 fin (caso)

 fin(Si)

 Sino

 Activa ACTUA5

FIN (ACTUA5)

10.2.1.1 NOMBRE DEL MEDICAMENTO: ACTUAS

PROGRAMA DONDE SE LOCALIZA: MOD8

OBJETIVO: Solicitar y capturar los datos del medicamento que se desea modificar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
X	Número del campo que se desea actualizar, variable de tipo entero.
M11	Para lecturas de datos, variable de tipo entero.
NOAL	Nombre del medicamento, variable de tipo string.
COVE	Contador que se utiliza, para determinar si se da de alta, alguna clave que anteriormente ha sido puesta disponible, variable de tipo entero.
PP	Para lectura de precios y descuentos, variable de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ACTUA6	Solicitar y capturar los datos del medicamento, que se desea modificar.

PSEUDOCODIGO.

(* Inicia procedimiento ACTUA5 *)

INICIA (ACTUA5)

Si X es menor que 21 entonces

 inicia (Si)

 En caso que X

 sea igual a 14: Despliega el dato actual del nombre del medicamento
 solicita el nuevo dato
 captura y actualiza el dato

 ó

 sea igual a 15: Despliega el dato actual del número de unidades
 vendidas en la temporada 1
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó

 sea igual a 16: Despliega el dato actual del número de unidades
 vendidas en la temporada 2
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó

 sea igual a 17: Despliega el dato actual del número de unidades
 vendidas en la temporada 3
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó

 sea igual a 18: Despliega el dato actual del número de unidades
 vendidas en la temporada 4
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó

 sea igual a 19: Si NM .PROV1 es igual a 0 entonces
 COVE <-- COVE + 1
 Despliega el dato actual del número de la clave
 del proveedor principal
 Solicita el nuevo dato
 Captura y actualiza el dato

 ó

 sea igual a 20: Despliega el dato actual del precio que da el
 primer proveedor
 Solicita el nuevo dato
 Captura y actualiza el dato

 fin (caso)

 fin (Si)

Sino

 Activa ACTUA6

FIN (ACTUA5)

10.2.1.1.1 NOMBRE DEL PROCEDIMIENTO: ACTUA6

PROGRAMA DONDE SE LOCALIZA: MOD8

OBJETIVO: Solicita y captura los datos del medicamento que se desea modificar.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
X	Número del campo que se desea actualizar, variable de tipo entero.
M11,M22,M33	Para lecturas de datos, variables de tipo entero.
PP	Para leer precios y descuentos, variable de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia Procedimiento ACTUA6 *)

INICIA (ACTUA6)

En caso que X

sea igual a 21: Despliega el dato actual del descuento que hace el
primer proveedor
Solicita el nuevo dato
Captura y actualiza el dato

ó

sea igual a 22: Despliega el dato actual de la clave del segundo
proveedor
Solicita el nuevo dato
Captura y actualiza el dato

ó

sea igual a 23: Despliega el dato actual del precio que da el segundo
proveedor
Solicita el nuevo dato
Captura y actualiza el dato

ó

sea igual a 24: Despliega el dato actual del descuento que hace el segundo
proveedor
Solicita el nuevo dato
Captura y actualiza el nuevo dato

ó

sea igual a 25: Despliega el dato actual de la clave del tercer proveedor
Solicita el nuevo dato
Captura y actualiza el dato

ó

sea igual a 26: Despliega el dato actual del precio que da el tercer
proveedor
Solicita el nuevo dato
Captura y actualiza el dato

ó

sea igual a 27: Despliega el dato actual del descuento que hace el tercer
proveedor
Solicita el nuevo dato
Captura y actualiza el dato

ó

sea igual a 28: Despliega el dato actual de la fecha en que se cargo
el campo de unidades en existencia
Solicita el nuevo dato
Captura y actualiza el dato

fin (caso)

FIN (ACTUA6)

10.2.2 NOMBRE DEL PROCEDIMIENTO: ACTUA4

PROGRAMA DONDE SE LOCALIZA: MODB

OBJETIVO: Determinar si los datos capturados del medicamento son actualizados.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
BF MAESTRO.FIL	ARCHIVO MAESTRO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
SIGA	Bandera para realizar la actualización, variable de tipo caracter.
X	Número del campo que se desea actualizar, variable de tipo entero.
I2	Apuntador para acceder tanto el archivo Maestro, como el archivo de Nombres y estadísticas, variable de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIHIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia procedimiento ACTUA4 *)

INICIA (ACTUA4)

Lee dato de continuación

Si SIGA es igual a 'S' entonces

 inicia (Si)

 Si X es menor que 14 entonces

 inicia (Si)

 Actualiza los datos capturados del Medicamento
 en el archivo Maestro

 Valida escritura

 fin (Si)

 Sino

 inicia (Sino)

 Actualiza los datos capturados del medicamento
 en el archivo de Nombres y Estadísticas

 Valida escritura

 fin (Sino)

 fin (Si)

FIN (ACTUA4)

10.3 NOMBRE DEL PROCEDIMIENTO: CLAVBA

PROGRAMA DONDE SE LOCALIZA: MODB

OBJETIVO: Dar de baja claves, del archivo de Claves Disponibles.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DESCRIPCION DISCO
NC	CVEDISP.FIL ARCHIVO DE CLAVES DISPONIBLES

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
XZ	Apuntador para acceder el archivo de Claves Disponibles, variable de tipo entero.
SS	Bandera para detectar si se decrementa el número de claves disponibles, variable de tipo caracter.
Z1	Variable auxiliar para almacenar el apuntador XZ, variable de tipo entero.
CVE	Número de claves disponibles, variable de tipo entero.
CONT	Contador de claves disponibles, variable de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
JJ	Indice para recorrer el archivo de Clave Disponibles, variable de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia procedimiento CLAUBA *)

INICIA (CLAUBA)

XZ <-- 0

Lee archivo de Claves Disponibles

Valida lectura

(* Obtener el número de claves disponibles *)

CONT <-- NC .CDISP

SS <-- ' '

Z1 <-- I2

(* Proceso para recorrer el archivo de Claves Disponibles *)

Haz desde JJ=1 hasta CONT lo siguiente

inicia (Haz)

XZ <-- JJ

Lee archivo de Claves Disponibles

Valida lectura

CVE <-- NC .CDISP

(* Proceso para dar de baja alguna clave, del archivo de Claves Disp.*)

Si SS es igual a '*' entonces

inicia (Si)

XZ <-- JJ - 1

Lee el archivo de Claves Disponibles

Valida lectura

NC .CDISP <-- CVE

Actualiza la clave en el archivo de Claves Disponibles

Valida escritura

fin (Si)

Si NC .CDISP es igual a Z1 entonces

SS <-- '*'

fin (Haz)

Si SS es igual a '*' entonces

CONT <-- CONT - 1

XZ <-- 0

Lee el archivo de Claves Disponible

Valida lectura

NC .CDISP <-- CONT

Escribe en el archivo el nuevo número de Claves Disponibles

Valida escritura

FIN (CLAUBA)

III.2.2.10 DOCUMENTACION DEL MODULO: CARGPED

NOMBRE EN EL MENU: CARGA DE UNIDADES PEDIDAS

PROGRAMA DONDE SE LOCALIZA: MODC

1. **OBJETIVO:** Cargar al campo de unidades pedidas del archivo Maestro (BF) la nueva cantidad de unidades pedidas.

2. **FUNCIONES PRINCIPALES:**
 - Solicitar los datos del medicamento al cual se le requieren pedir más unidades.
 - Validar los datos solicitados.
 - Resrabar los datos en el archivo Maestro.

3. **NARRATIVA DEL MODULO.**

Si el archivo Maestro (BF) no contiene información, el programa le desplegará un mensaje de error, indicándole que el archivo se encuentra vacío y que necesita activar el módulo ESCRIBE (opción F), para dar de alta algún medicamento. En caso que el archivo Maestro contenga información, el programa le solicitará los datos del medicamento al cual le desea cargar nuevas unidades, validará los datos proporcionados y resrabará esta información en el archivo Maestro. Este proceso de cargar unidades se repite hasta que usted lo desee.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL PROGRAMA	ARCHIVO DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información en el Sistema, variable de tipo entero.
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
II	Apuntador para acceder el archivo de Nom. y Est. del Med., variable de tipo entero.

5.2 VARIABLES MODULARES: Ninguna

5.3 VARIABLES LOCALES:

NOMBRE	DESCRIPCION
CVEM	Clave del medicamento, variable de tipo entero.
BAND	Bandera para detectar algún error en la transmisión de los datos, variable de tipo caracter.
SN	Respuesta de si o no, variable de tipo caracter.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
PIEDAT	Solicitar los datos del medicamento al cual se le desean cargar nuevas unidades.
ERRCHK	Valida las lecturas y las escrituras que se hacen a los archivos.
SEEKREAD	Realiza lecturas de acceso directo.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Finaliza la sesión.

DIAGRAMA JERARQUICO

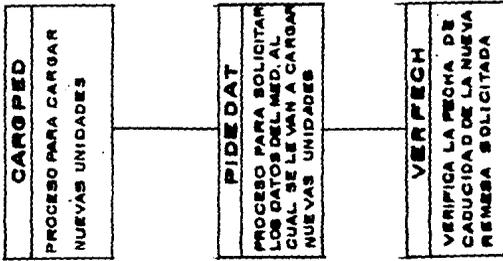


DIAGRAMA DE FLUJO DE DATOS

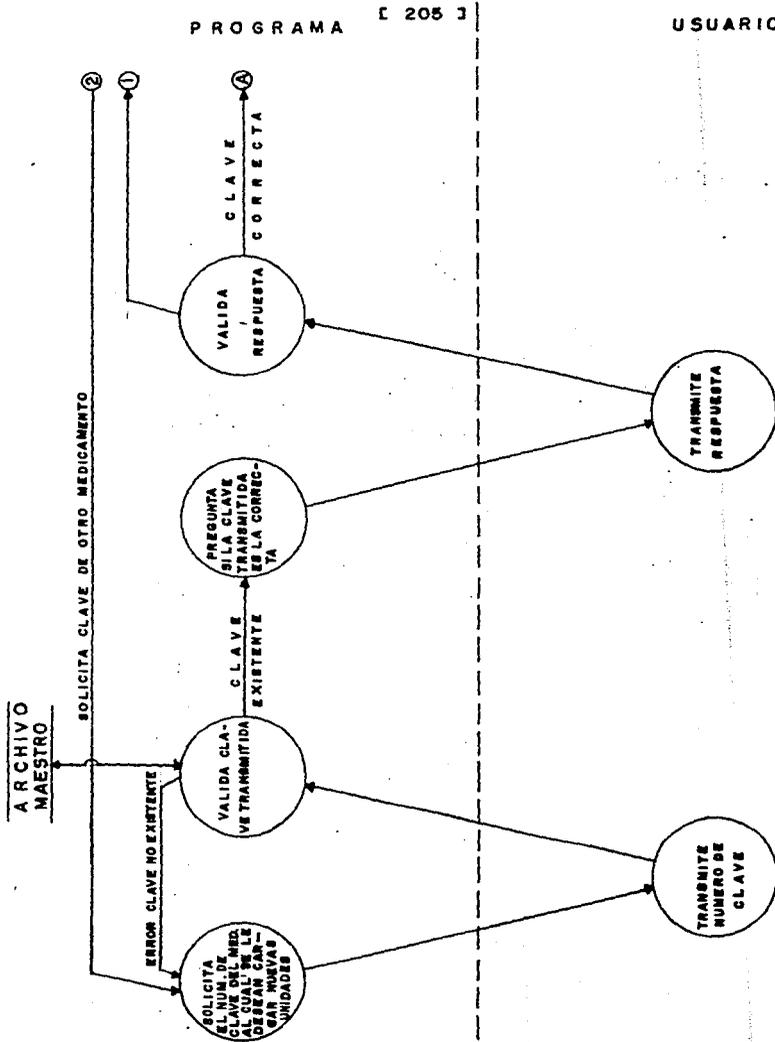


DIAGRAMA JERARQUICO

CARGPED
PROCESO PARA CARGAR
NUEVAS UNIDADES

PIDE DAT
PROCESO PARA SOLICITAR
LOS DATOS DEL MED. AL
CUAL SE LE VAN A CARGAR
NUEVAS UNIDADES

VERFECH
VERIFICA LA FECHA DE
CADUCIDAD DE LA NUEVA
REMSA SOLICITADA

E 204 1

DIAGRAMA DE FLUJO DE DATOS

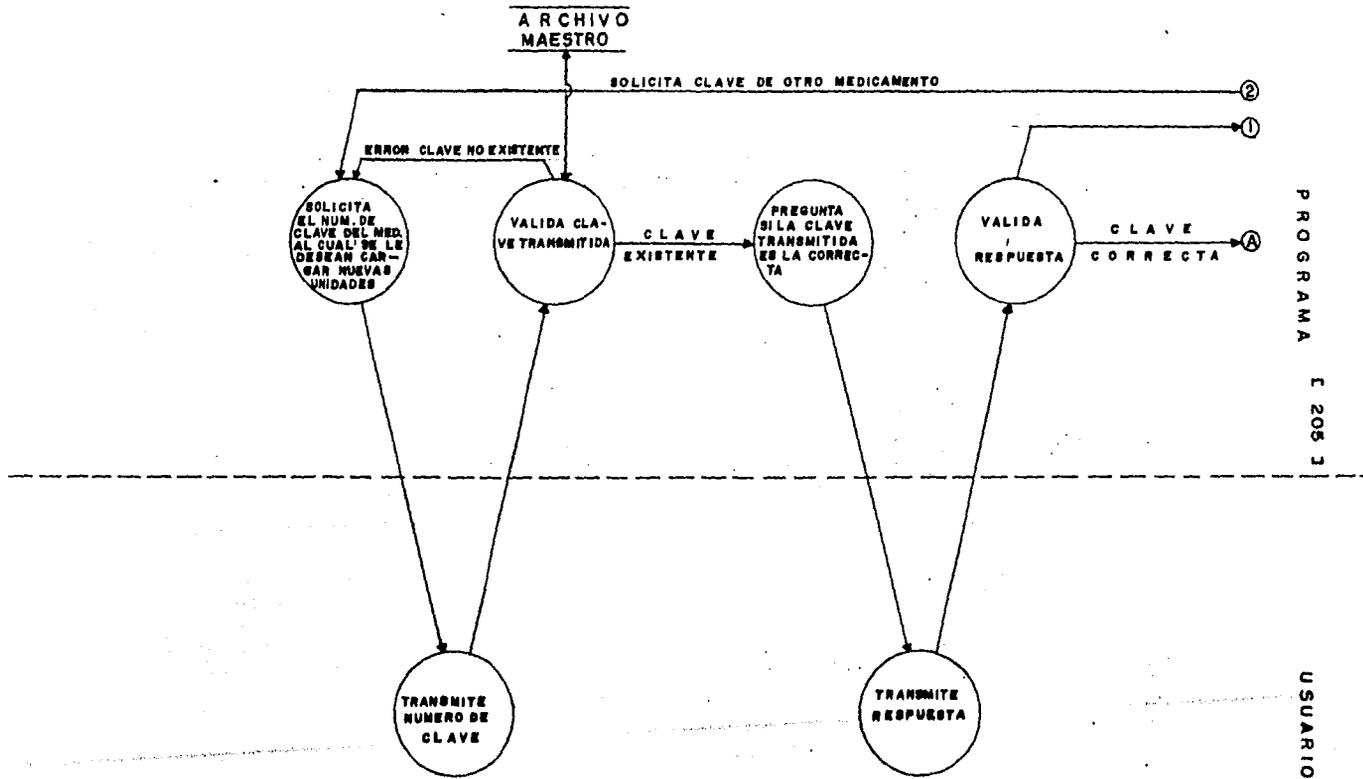
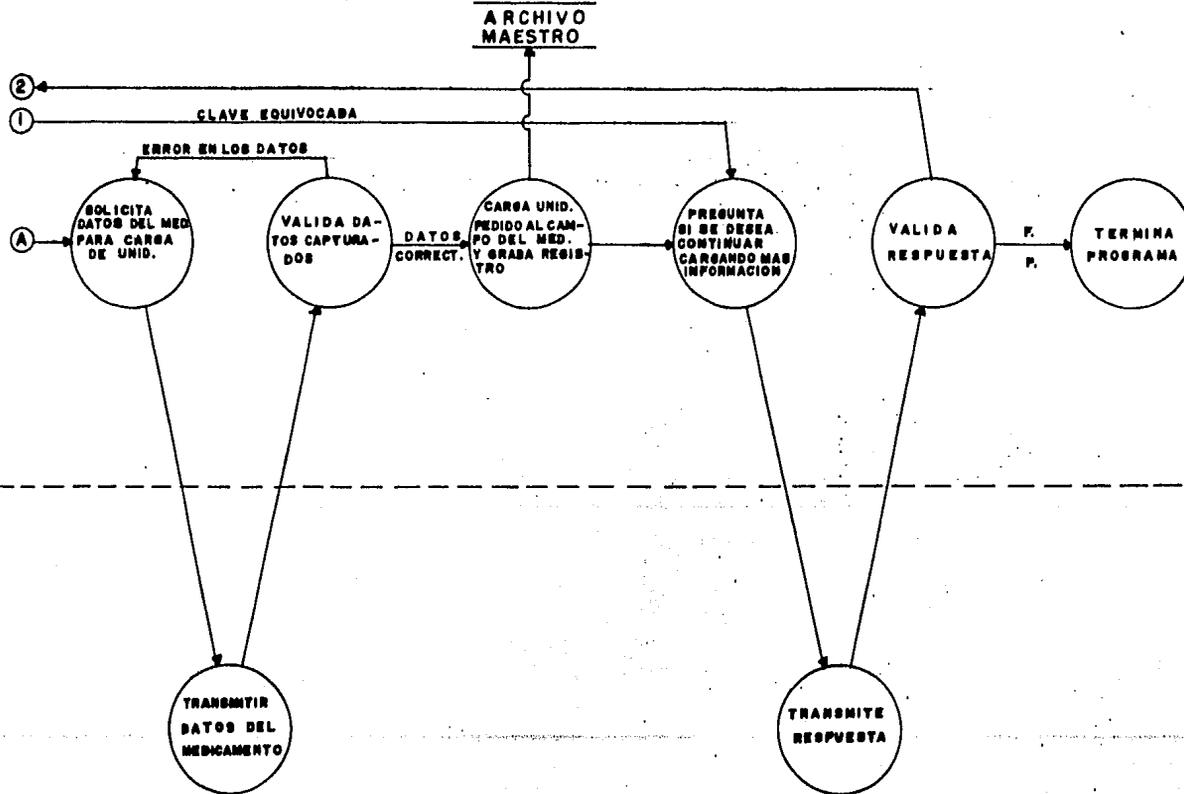


DIAGRAMA DE FLUJO DE DATOS

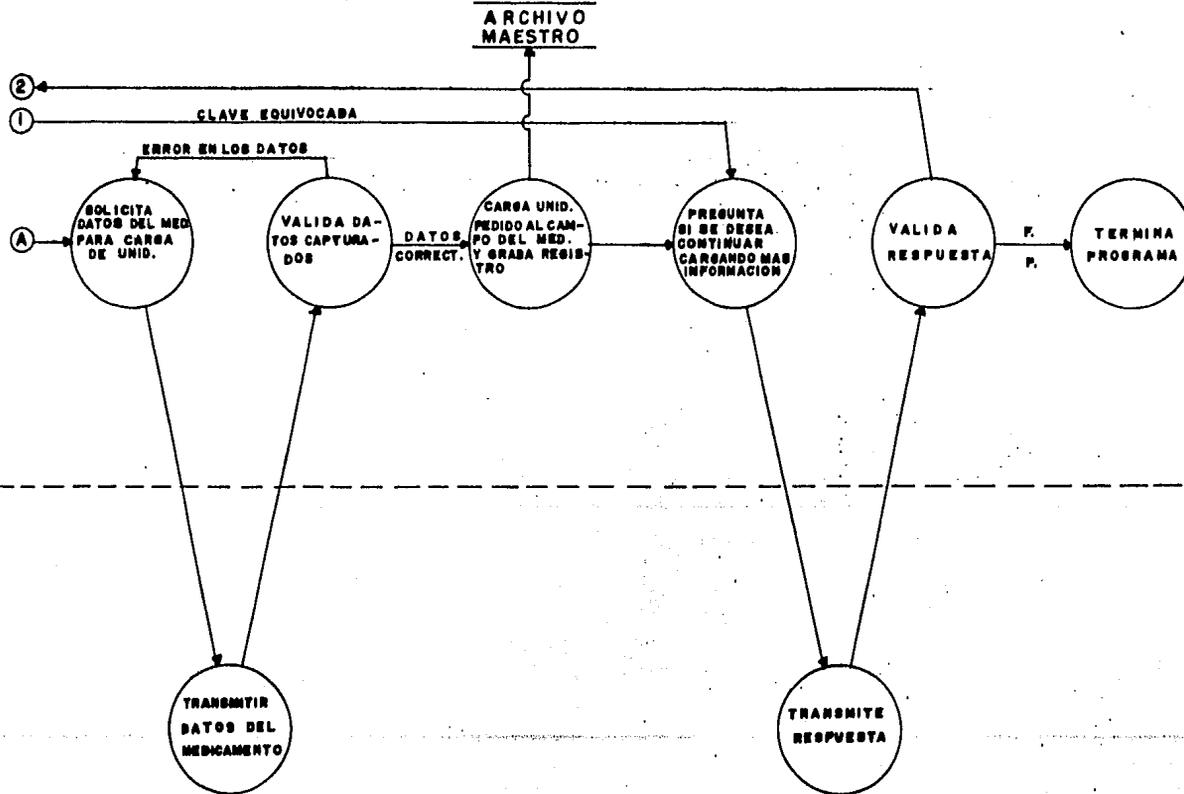


PROGRAMA

C 205,1 J

USUARIO

DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

C 205,1 J

USUARIO

9. PSEUDOCODIGO

(* Inicia programa CARGPED *)

INICIA (CARGPED)

```

BAND <-- ' '
SN <-- ' '
Si LIM es diferente de 0 entonces
  inicia (Si)
  Despliega mensaje
  'A CONTINUACION SE LE PIDEN LOS DATOS DE LOS MEDICAMENTOS, A'
  'LOS CUALES SE LES DESEA CARGAR UNIDADES PEDIDAS'
  Repite
  Despliega mensaje
  'DEME CLAVE DEL MEDICAMENTO => '
  Lee clave del medicamento
  Si CVEM es mayor ó igual a 1 entonces
  Si CVEM es menor ó igual a LIM entonces
    inicia (Si)
    Presunta si la clave transmitida es la correcta
    Repite
    Lee respuesta (SN)
    Si SN es igual a 'S' ó a 's' entonces
      inicia (Si)
      I <-- CVEM
      II <-- I
      Activa PIDE DAT
      fin (Si)
    Sino
      Si SN es igual a 'N' ó a 'n' entonces
        SN <-- 'N'
      Sino
        Despliega mensaje de error y vuelve a solicitar el dato
        Hasta que SN sea igual a 'S' o a 'N'
      fin (Si)
    Sino
      Despliega mensaje de error de clave excedida del rango permitido
    sino
      Despliega mensaje de error de que la clave transmitida no existe
  Si BAND es diferente de un '*' entonces
    inicia (Si)
    Despliega mensaje
    'DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O => '
    Repite
    Lee respuesta (SN)
    Si SN es igual a 'S' ó a 's' entonces
      SN <-- 's'
    Sino
      Si SN es igual a 'N' ó a 'n' entonces
        SN <-- 'N'
      Sino
        Despliega mensaje de error y vuelve a solicitar el dato
        Hasta que SN sea igual a 'N' ó a 'n'
    fin (Si)
  Sino
    Despliega mensaje de error de que el archivo Maestro se encuentra vacío
  Despliega mensaje de terminación de programa
FIN (CARGPED)

```

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: PIDE DAT

PROGRAMA DONDE SE LOCALIZA: MODC

OBJETIVO: Solicitar los datos del medicamento al cual se le desean cargar nuevas unidades.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo Maestro, variable de tipo entero.
II	Apuntador para acceder el archivo de Nom. y Est. del Med., variable de tipo entero.

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
CUP	Cantidad de unidades pedidas, variable de tipo entero.
CVEP	Clave del proveedor, variable de tipo entero.
NPRECH	Nuevo precio del medicamento, variable de tipo real.
NIMPM	Nuevo impuesto del medicamento, variable de tipo real.
BAND1	Bandera para detectar algún error en los datos transmitidos variable de tipo caracter.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE

OBJETIVO

VERFECH Verificar fecha de caducidad de las unidades pedidas.

PSEUDOCODIGO

(* Inicia procedimiento PIDE DAT *)

INICIA (PIDE DAT)

CVEP <-- I

Lee el archivo de Nom. y Est. del Med.

Valida lectura

Lee el archivo Maestro

Valida lectura

II <-- CVEP

I <-- II

Solicitar los datos del medicamento de unidades pedidas, del precio del medicamento y del impuesto del medicamento.

Validar los datos y almacenar la información en el archivo Maestro

Activar VERFECH

Solicita clave de proveedor

Valida clave de proveedor

Almacena la clave en el archivo Maestro

Reescribe los datos en el archivo Maestro

Valida la reescritura

Reescribe la clave del proveedor en el archivo de Nom. y Est. del Med.

Valida la reescritura

FIN (PIDE DAT)

10.1.1 NOMBRE DEL PROCEDIMIENTO: VERFECH

PROGRAMA DONDE SE LOCALIZA: MODC

OBJETIVO: Validar fecha de caducidad de las unidades pedidas.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	ARCHIVO DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO

VARIABLES.

VARIABLES EXTERNAS: Ninguna

VARIABLES MODULARES: Ninguna

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
A	Año de la fecha de caducidad de las unidades cansadas, variable de tipo entero.
M	Mes de la fecha de caducidad de las unidades cansadas, variable de tipo entero.
D	Día de la fecha de caducidad de las unidades cansadas, variable de tipo entero.
BAND2	Bandera para detectar algún error en la transmisión de la fecha de caducidad, variable de tipo caracter.
TIPA	Tipo de año (Bisiesto) de la fecha de caducidad, variable de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguna

PSEUDOCODIGO

(* Inicia procedimiento VERFECH *)

INICIA (VERFECH)

Solicita fecha de caducidad

BAND2 <-- ' ' ' ' ' '

Repite

Si BAND2 es igual a '*' entonces

Despliega mensaje

'FAVOR DE VOLVER A DAR LA FECHA DE CADUCIDAD => '

BAND2 <-- ' ' ' ' ' '

Lee fecha de caducidad

Valida fecha de caducidad (año,mes,día)

Hasta que BAND2 sea igual a ' ' ' ' ' '

(* Asigna datos de fecha a el archivo Maestro *)

BF.AA <-- A

BF.MM <-- M

BF.DD <-- D

FIN (VERFECH)

III.2.2.11 DOCUMENTACION DEL MODULO: LEERAN

NOMBRE EN EL MENU : LISTA RANGOS DE TEMPORADAS
Y LISTA PROVEEDORES

PROGRAMA DONDE SE LOCALIZA: MODD

1. OBJETIVO. Controlar el proceso para la elaboración de reportes de ventas por temporada y datos estadísticos de precios y descuentos por proveedor.

2. FUNCIONES PRINCIPALES:

- Obtener un reporte de las ventas de los medicamentos por temporada.
- Obtener reporte que contenga el nombre del medicamento, así como también, la clave, precio y descuento de proveedores.

3. NARRATIVA DEL PROGRAMA:

Si el archivo Maestro no contiene información, el programa le despliega un mensaje de error, indicándole que el archivo se encuentra vacío y que es necesario, que ejecute antes el módulo ESCRIBE opción <F>, para cargar información en el archivo.

Si el archivo Maestro contiene información, el programa le despliega dos opciones, que le permiten obtener dos reportes, uno obtiene las ventas de los medicamentos por temporada y el otro algunos datos del medicamento con respecto al proveedor que lo surte.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
NH	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras a los archivos, tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento.

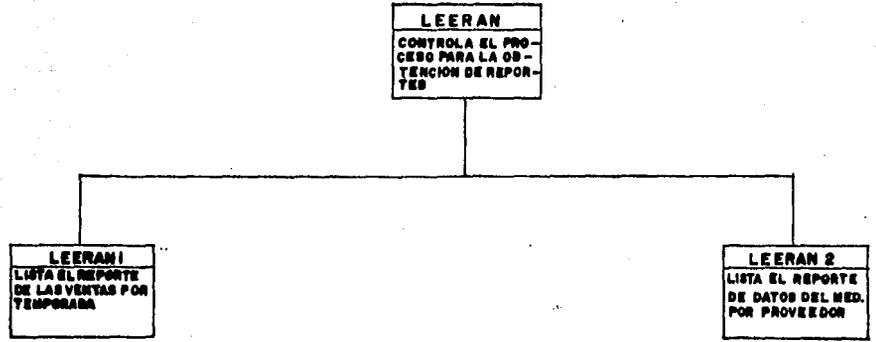
5.3 VARIABLES LOCALES:

NOMBRE	DESCRIPCION
SIG	Bandera para detectar el fin del proceso, del programa, variable de tipo caracter.
A6	Número de listado que se desea obtener, variable de tipo entero.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
LEERAN1	Obtener un reporte de las ventas de los medicamentos por temporada.
LEERAN2	Obtener un reporte de datos y nombre del medicamento con respecto al proveedor que lo surte.

DIAGRAMA JERARQUICO



C 213 3

9. PSEUDOCODIGO.

(* Inicia procedimiento LEERAN *)

INICIA (LEERAN)

Si LIM es mayor que cero entonces
inicia (Si)

SIG <-- ' '

Repite

Asigna el archivo de Nombres y Estadis. a la corrida

Despliega mensaje

' CUAL LISTADO DESEA OBTENER : '

' 1.- NOMBRE Y RANGOS '

' 2.- PROVEEDORES '

' ANOTE EL NUMERO DEL LISTADO QUE DESEA : '

Lee dato

En caso que A6

sea igual a 1: Activa LEERAN1

o

sea igual a 2: Activa LEERAN2

Sino

Despliega mensaje

' ANOTE UNICAMENTE 1 ó 2 '

fin (caso)

Despliega mensaje

' DESEA SACAR OTRO LISTADO S)I O N)O : '

Lee dato

Hasta que SIG sea igual a 'N'

I <-- 0

fin (Si)

Sino

Despliega mensaje

Error el archivo se encuentra vacio

FIN (LEERAN)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: LEERAN1

PROGRAMA DONDE SE LOCALIZA: MODD

OBJETIVO: Obtener un reporte de las ventas de los medicamentos por temporada.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos, variable de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento, variable de tipo entero.
A5	Espacios que se dejan en la línea de impresión para centrar información, variable de tipo entero.
A3	Nombre del medicamento, variable de tipo string.
A4	Longitud en caracteres del nombre del medic., variable de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
A1	Indice para recorrer el archivo de Nombres y Estadísticas del medicamento.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida las lecturas y escrituras a los archivos.

PSEUDOCODIGO.

(* Inicia procedimiento LEERAN1 *)

INICIA (LEERAN1)

Despliega el encabezado del listado de ventas por temporada
Haz desde A1=1 hasta LIM lo siguiente

inicia (Haz)

I <-- A1

Lee archivo de Nombres y Estadíst.

Valida lectura

Despliega o imprime línea de datos de venta de medicamentos por temporada

fin (Haz)

FIN (LEERAN1)

10.2 NOMBRE DEL PROCEDIMIENTO: LEERAN2

PROGRAMA DONDE SE LOCALIZA: MODD

OBJETIVO: Obtener un reporte de datos y nombre del medicamento con respecto al proveedor que los surte.

ARCHIVO QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADIS- TICAS

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIM	Número de la clave máxima que contiene información dentro del Sistema, variable de tipo entero.
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos, tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder el archivo de Nombres y Estadísticas del medicamento, variable de tipo entero.
A5	Espacios que se dejan en la línea de impresión para centrar información, variable de tipo entero.
A3	Nombre del medicamento, variable de tipo string.
A4	Longitud en caracteres del nombre del medicamento, variable de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
AI	Índice para recorrer el archivo de Nombres y Estadísticas del medicamento.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida las lecturas y escrituras a los archivos.

PSEUDOCODIGO.

(* Inicia Procedimiento LEERAN2 *)

INICIA (LEERAN2)

```

Despliega encabezado de datos de proveedor
Haz desde AI=1 hasta LIM lo siguiente
  inicia (Haz)
  I <-- AI
  Lee el archivo de Nombres y estadísticas
  Valida lectura
  despliega o imprime los datos del proveedor
  fin (Haz)
    
```

FIN (LEERAN2)

III.2.3 D O C U M E N T A C I O N
D E L O S P R O G R A M A S
D E L
S U B S I S T E M A
D E
C O N T R O L D E V E N T A S

III.2.3.1 DOCUMENTACION DEL MODULO: PROGRAMA PRINCIPAL.

PROGRAMA DONDE SE LOCALIZA: PROY1.

1. OBJETIVO: Asignar un módulo de acuerdo con la función escogida del menú principal. De hecho este programa controla todo el SUBSISTEMA CONTROL DE VENTAS.

2. FUNCIONES PRINCIPALES.
 - Despliega en la pantalla o impresora el menú principal de funciones del Subsistema de Control de Ventas.
 - Solicita la función a ejecutar.
 - Verifica la función seleccionada.
 - Despliega mensajes de errores.
 - Asigna la función deseada.
 - Valida fechas de trabajo.
 - Valida accesos a los archivos, ya sea para lectura o escritura.
 - Asigna y cierra archivos.

3. NARRATIVA DEL PROGRAMA:

La primera función que realiza este programa, es la de listar un mensaje de bienvenida al usuario; además le especifica todos los archivos que deben existir en el disco correspondiente para que el Subsistema de "CONTROL DE VENTAS", funcione adecuadamente.

Si por algún motivo falta un archivo de datos, entonces se da por terminada la sesión, listando los errores para que el usuario los corrija.

En caso de que se cumplan todas y cada una de las condiciones, entonces son asignados los archivos al sistema.

De los registros de alta seguridad de cada archivo, se leen los datos indispensables para el funcionamiento del sistema. Entre estos datos se cuentan: número máximo de registros en cada uno de ellos, fechas, claves de seguridad, identificación de la farmacia, etc.

De acuerdo con estos datos y si el Archivo Maestro y el de Nombres y Estadísticas contienen información, y si existe el Archivo de Trabajo, aun cuando esté vacío, entra de lleno al sistema.

Escribe los datos de identificación de la farmacia, así como la cantidad de registros que contienen información.

A continuación despliega el menú principal de funciones, es opcional, para que se elija alguna función, cada una de las cuales se encuentra en un módulo.

Todas las funciones tienen una clave, la cual es única y sirve para la activación del módulo correspondiente.

Verifica que la función seleccionada se encuentre entre las posibles; si no se cumplen determinadas condiciones, no permite el acceso a las funciones.

Cuando es seleccionada una función válida, entonces activa el módulo en el que se encuentra y transfiere el control.

En caso de error, lo detecta y especifica.

En el momento que ya no se desea ninguna función cierra los archivos y da por terminada la sesión.

En este programa, también existen procedimientos que leen y validan fechas, así como validación de accesos a los archivos, ya sea para lectura o escritura.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima del registro que contiene información en BF. Es de tipo entero.
LIMIVA	Número de la clave máxima del registro que contiene información en VA. Es de tipo entero.
BLANCNDM	Contiene el resultado de 30 menos la longitud del nombre de la empresa; es decir, son los espacios en blanco que quedan al final del nombre de la farmacia. Se usa para centrar la impresión del nombre. Es de tipo entero.
NOMBREMP	Contiene el nombre de la empresa. Es una cadena de 30 caracteres.
ANOEX	Año de la fecha a ser validada. Es de tipo entero.
MESEX	Mes de la fecha a ser validada. Es de tipo entero.
DIAEX	Día de la fecha a ser validada. Es de tipo entero.

VER Si la fecha es válida, toma valor 1; si es falsa toma el valor 0. Es de tipo entero.

ERROR Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
EXISTVA	Indica la existencia del archivo VA. Si vale 0 no existe; con otro valor si existe. Es de tipo entero.
LIMINM	Número de la clave máxima del registro que contiene información en NM. Es de tipo entero.
SIONO	Respuesta a la pregunta, para saber si desea ver el menú. Puede valer 'S' o 'N'. Es de tipo cadena de caracteres.
FUNDESEA	Función deseada; cada función tiene asociada una letra en el menú. Es de tipo caracter.
I	Apuntador para acceder a los registros de los archivos. Es de tipo entero.

5.3 VARIABLES LOCALES: Ninguna.

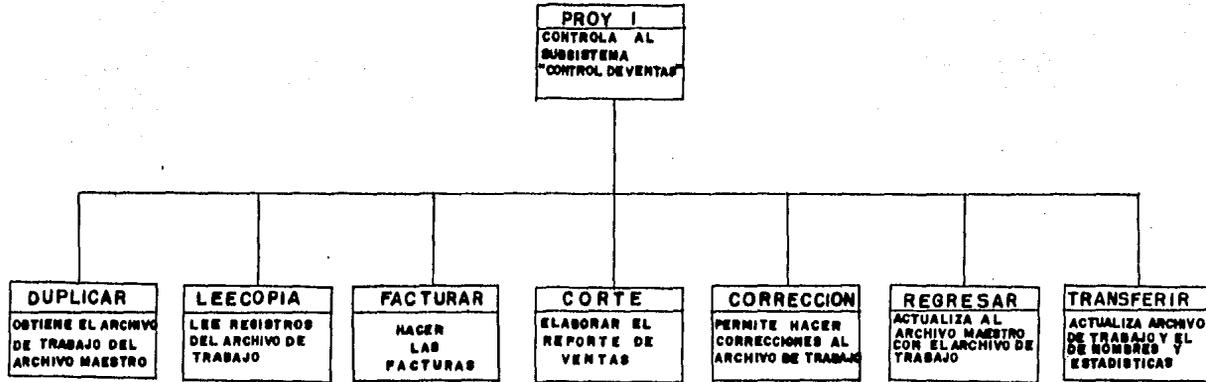
6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
FECHAUX	Lee los datos de la fecha y obtiene su validación.
CABEZA	Tomar del archivo NM los datos de identificación de la farmacia.
LETRA	Despliega el menú principal de funciones.
MENSAJE	Da la bienvenida al usuario y lista las condiciones mínimas para que el sistema trabaje.
ASIGNAR	Asignar los archivos al proceso y presentar la información del sistema contenida en el registro cero.
BUSCAFUN	Asigna la función seleccionada por el usuario.
BUSCAERR	Busca el error cometido, para especificarlo.
LISTAERR	Despliega el mensaje de error de acuerdo con el valor del parámetro.
DUPLICAR	Obtener una copia parcial del Archivo Maestro llamada Archivo de Trabajo, para ser usada en los módulos del Subsistema de Control de Ventas.
LECOPIA	Permitir la lectura de cualquier registro del Archivo de Trabajo del día, para hacer consultas rápidas.
FACTURAR	Elaborar la factura del cliente, actualizando el Archivo de Trabajo.

- CORTE** Elaborar el reporte del Corte de Caja del día, para obtener un listado con los datos más significativos para el usuario.
- REGRESA** Redresar información del Archivo de Trabajo al Archivo Maestro, con la información actualizada, después de una sesión de ventas.
- TRANSFERIR** Transferir información estadística de ventas del archivo BF al archivo NM. Actualiza campos del archivo BF.
- ACTUALI7** Hacer correcciones al Archivo de Trabajo, en caso de que se hayan presentado problemas al momento de hacer las facturas.
- SEEKREAD** Realiza lecturas de acceso directo a los archivos.
- SEEKWRITE** Realiza escrituras en los archivos.
- EXIT** Dar por terminada la sesión.

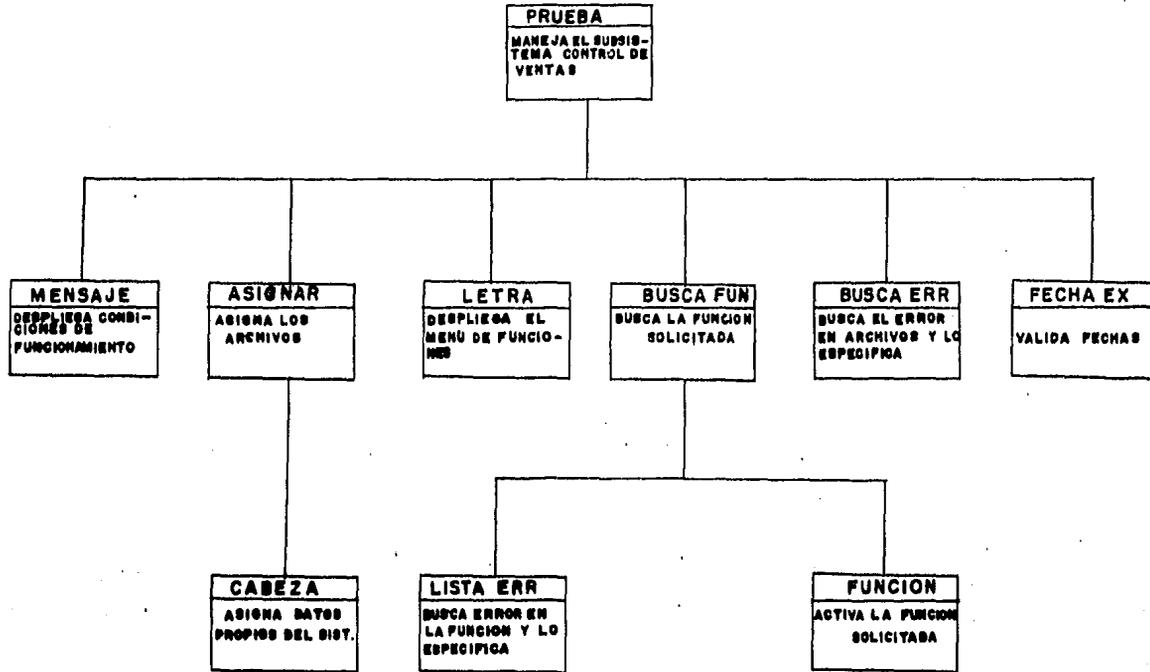
7. DIAGRAMA JERARQUICO.

DIAGRAMA JERARQUICO



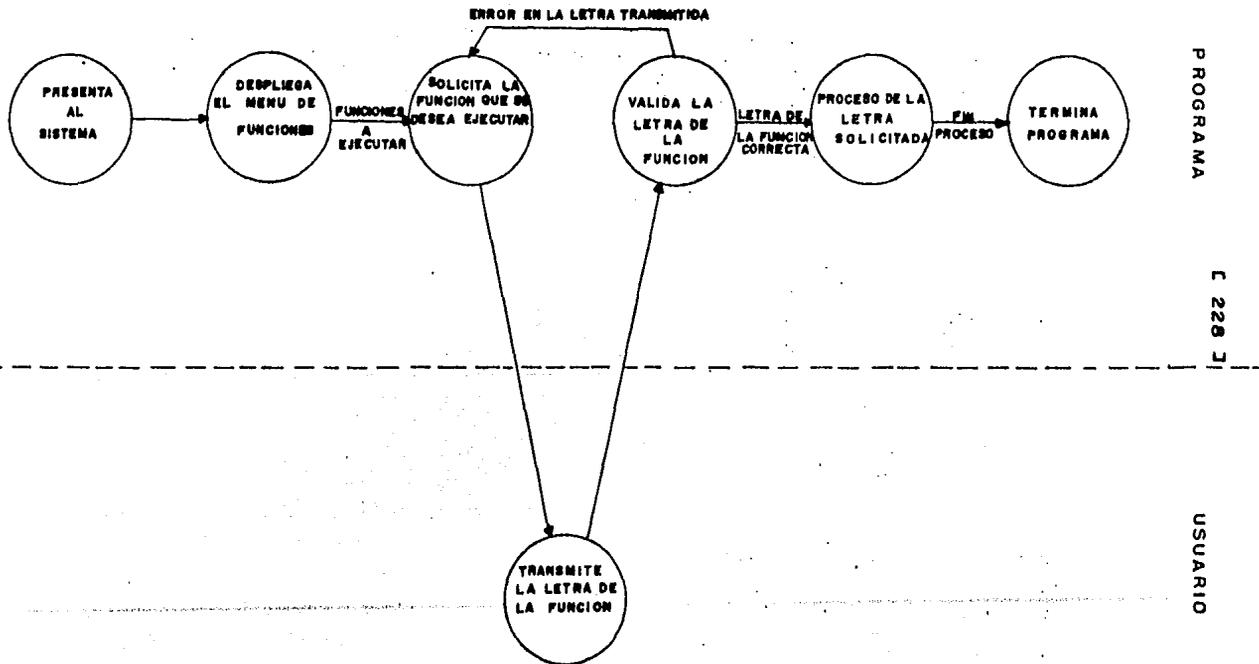
U 227 1

DIAGRAMA JERARQUICO



C 22713

DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO

(* Inicia el PROGRAMA PRINCIPAL *)

INICIA (PROGRAMA PRINCIPAL)

Activa MENSAJE

Activa ASIGNAR;

Despliega el mensaje:

*EL SISTEMA TIENE (valor de LIMMAST) MEDICAMENTOS *

Si LIMMAST es mayor que cero y

EXISTVA es mayor que cero y

LIMINM es mayor que cero entonces

Inicia (Si)

Repite

Mensaje:

*DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ?

ANOTE S)I O N)O : *

Lee SIONO

Si SIONO es igual a 'S' entonces

Activa LETRA

fin (Si)

Mensaje:

*FUNCION DESEADA : *

Lee FUNDESEA

Activa BUSCAFUN

Hasta que sea falso

fin (Si)

sino

Inicia (sino)

Activa BUSCAERR

fin (sino)

I <--- LIMMAST

Cierra los tres archivos

FIN (PROGRAMA PRINCIPAL)

10 DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: ERRCHK

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Validar lecturas y escrituras a los archivos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
--------	-------------

ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.
-------	---

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
--------	-------------

IDRESULT	Variable para hacer la validación de lecturas y escrituras en archivos. Es de tipo entero.
----------	--

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVUCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento ERRCHK *)

INICIA (ERRCHK)

ERROR <--- VERDADERO

En caso que IORESULT

sea igual a 0 : ERROR <--- FALSO

o

sea igual a 1 : Mensaje: "LEYENDO DONDE NO HAY DATOS"

o

sea igual a 2 : Mensaje: "ERROR DE CP/M"

o

sea igual a 3 : Mensaje: "BUSCANDO UN LIMITE EN BLANCO"

o

sea igual a 4 : Mensaje: "ERROR DE CP/M"

o

sea igual a 5 : Mensaje: "LLEGO AL FINAL DEL DISCO"

o

sino : Mensaje: "ERROR DE CODIGO DESCONOCIDO"

fin (caso)

FIN (ERRCHK)

10.2 NOMBRE DEL PROCEDIMIENTO: FECHAEX

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Lee los datos de la fecha y obtiene su validación.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
MESEX	Mes de la fecha a ser validada. Es de tipo entero.
DIAEX	Día de la fecha a ser validada. Es de tipo entero.
ANDEX	Año de la fecha a ser validada. Es de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
VER	Si la fecha es válida, toma valor 1; si es falsa toma el valor 0. Es de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
ANI	Sirve para ver si un año es bisiesto. Es de tipo entero.
DI2	Número de días que tiene el mes. Es de tipo entero.
SIG	Variable de control para proceso iterativo, el cual deja de hacerse cuando SIG sea igual a 0. Es de tipo caracter.

PROCEDIMIENTOS QUE INVOKA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento FECHAEX *)

INICIA (FECHAEX)

Mensaje: "ANOTE LA FECHA DE HOY : "

SIG <--- ' '

Repite

ANOEX <--- 0

MESEX <--- 0

DIAEX <--- 0

Solicita los datos del año, mes y día por medio de las variables ANOEX, MESEX y DIAEX

VER <--- 0

Si ANOEX es mayor o igual que 86 y

ANOEX es menor que 100 y

MESEX es mayor que 0 y

DIAEX es mayor que 0 entonces

Inicia (Si)

VER <--- 1

ANI <--- ANOEX MOD 4

En caso que MESEX

sea igual a 1,3,5,7,8,10,12: DI2 <---31

o

sea igual a 4,6,9,11 : DI2 <--- 30

o

sea igual a 2 : Inicia (MESEX=2)
 Si AN1 es igual a 0 entonces
 DI2 <--- 29
 sino
 DI2 <--- 28
 fin (Si)
 fin (MESEX=2)

sino : VER <--- 0

fin (caso)

Si VER es igual a 1 entonces

Inicia (si)

SI DIAEX es mayor que DI2 entonces

VER <--- 0

fin (si)

fin (si)

fin (Si)

Si VER es igual a 1 entonces

Inicia (Si)

SIG <--- 0

Mensaje: "ANOTO LA FECHA CORRECTAMENTE "

Escribe la fecha

fin (si)

sino

Inicia (sino)

Marca error y solicita la fecha nuevamente

SIG <--- ' '

fin (sino)

Hasta que SIG sea igual a '0'

FIN (FECHAEX)

10.3 NOMBRE DEL PROCEDIMIENTO: LETRA

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Despliega el menú principal de funciones.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
BLANCNOM	Contiene el resultado de 30 menos la longitud del nombre de la empresa; es decir, son los espacios en blanco que quedan al final del nombre de la farmacia. Se usa para centrar la impresión del nombre. Es de tipo entero.
NOMBREMP	Contiene el nombre de la empresa. Es una cadena de 30 caracteres.

VARIABLES MODULARES: Ninguna.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el Procedimiento LETRA *)

INICIA (LETRA)

NOMBRE DE LA FARMACIA

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO
B	LEER EL ARCHIVO DE TRABAJO DEL DIA
C	FACTURACION PARA CLIENTES
D	CORTE DE CAJA DE LAS VENTAS DEL DIA
E	CORRECCIONES AL ARCHIVO MAESTRO
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA :

FIN (LETRA)

10.4 NOMBRE DEL PROCEDIMIENTO: MENSAJE

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Da la bienvenida al usuario y especifica las condiciones minimas que deben existir para que funcione el programa.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES. Ninsuna.

PSEUDOCODIGO.

(* Inicia el procedimiento MENSAJE *)

INICIA (MENSAJE)

! ! ! BIENVENIDO A UNA NUEVA SESION DE TRABAJO ! ! !

EL SUBSISTEMA "CONTROL DE VENTAS", REQUIERE QUE
EXISTAN EN EL DISCO B. LOS SIGUIENTES ARCHIVOS:

- 1.- ARCHIVO MAESTRO
- 2.- ARCHIVO DE TRABAJO
- 3.- ARCHIVO DE NOMBRES Y ESTADISTICAS

FIN (MENSAJE)

10.5 NOMBRE DEL PROCEDIMIENTO: ASIGNAR

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Asignar los archivos al proceso y presentar la información del sistema contenida en el registro cero.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADÍSTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
--------	-------------

LIMMAST	Número de la clave máxima del registro que contiene información en BF. Es de tipo entero.
---------	---

LIMIVA Número de la clave máxima del registro que contiene información en VA. Es de tipo entero.

ERROR Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder a los registros de los archivos. Es de tipo entero.
LIMINM	Número de la clave máxima del registro que contiene información en NM. Es de tipo entero.
EXISTVA	Indica la existencia del archivo VA. Si vale 0 no existe; con otro valor si existe. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
CABEZA	Tomar del archivo NM los datos de identificación de la farmacia.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el Procedimiento ASIGNAR *)

INICIA (ASIGNAR)

```
Asigna el Archivo Maestro BF
Asigna el Archivo de Trabajo VA
Asigna el Archivo de Nombres y Estadísticas NM
Activa CABEZA
I <--- 0
LIMMAST <--- 0
LIMIYA <--- 0
LIMINM <--- 0
EXISTVA <--- 0
Lee el registro I del archivo BF
(* Valida la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  Activa a EXIT
fin (si)
LIMMAST <--- BF.MS1
Lee el registro I del archivo VA
(* Valida la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  Activa a EXIT
fin (si)
LIMIYA <--- VA.UNID
EXISTVA <--- VA.UVPD
Lee el registro I del archivo NM
(* Valida la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  Activa a EXIT
fin (si)
LIMINM <--- NM.CLAVE
```

FIN (ASIGNAR)

10.6 NOMBRE DEL PROCEDIMIENTO: LISTAERR

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Despliega el mensaje de error de acuerdo con el valor del parámetro.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
NUMERR	Parámetro por medio del cual se pasa el número del mensaje de error a ser listado. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el Procedimiento LISTAERR *)

INICIA (LISTAERR)

Mensaje:

* * * * E R R O R * * * *

En caso que NUMERR

sea igual a 1: Mensaje:
EL ARCHIVO MAESTRO NO EXISTE O ESTA VACIO

sea igual a 2: Mensaje:
EL ARCHIVO DE TRABAJO NO EXISTE O ESTA VACIO

sea igual a 3: Mensaje:
EL ARCHIVO DE NOM. Y EST. NO EXISTE O ESTA VACIO

sea igual a 4: Mensaje:

EL ARCHIVO DE TRABAJO NO CONTIENE INFORMACION
FAVOR DE LISTAR EL MENU PRINCIPAL DE FUNCIONES
Y ELIJA LA FUNCION <A> PARA LLENARLO

sea igual a 5: Mensaje:

UNICAMENTE SE ACEPTA A, B, C, D, E, F o Q

sea igual a 6: Mensaje:

ARCHIVO MAESTRO TIENE (valor de LIMMAST)
REGISTROS
ARCHIVO DE NOMBRES Y EST. TIENE (valor de
LIMINM) REGISTROS
ESTA USANDO ARCHIVOS EQUIVOCADOS

sea igual a 7: Mensaje:

ARCHIVO MAESTRO TIENE (valor de LIMMAST)
REGISTROS
ARCHIVO DE TRABAJO TIENE (valor de LIMIVA)
REGISTROS

ESTA USANDO UN ARCHIVO DE TRABAJO EQUIVOCADO

fin (caso)

FIN (LISTAERR)

10.7 NOMBRE DEL PROCEDIMIENTO: BUSCAERR

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Verifica el valor de las claves máximas de cada archivo en base a este dato, activa el listado del error.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima del registro que contiene información en BF. Es de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
EXISTVA	Indica la existencia del archivo VA. Si vale 0 no existe; con otro valor si existe. Es de tipo entero.
LIMINM	Número de la clave máxima del registro que contiene información en NM. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

10.8 NOMBRE DEL PROCEDIMIENTO: BUSCAFUN

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Toma el valor de FUNDESEA, lo valida y asigna el módulo seleccionado.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima del registro que contiene información en BF. Es de tipo entero.
LIMIWA	Número de la clave máxima del registro que contiene información en VA. Es de tipo entero.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
FUNDESEA	Función deseada; cada función tiene asociada una letra en el menú. Es de tipo caracter.
LIMINM	Número de la clave máxima del registro que contiene información en NM. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
DUPLICAR	Obtener una copia parcial del Archivo Maestro llamada Archivo de Trabajo, para ser usada en los módulos del Subsistema de Control de Ventas.
LECOPIA	Permitir la lectura de cualquier registro del Archivo de Trabajo del día, para hacer consultas rápidas.
FACTURAR	Elaborar la factura del cliente, actualizando el Archivo de Trabajo.
CORTE	Elaborar el reporte del Corte de Caja del día, para obtener un listado con los datos más significativos para el usuario.
REGRESA	Regresar información del Archivo de Trabajo al Archivo Maestro, con la información actualizada, después de una sesión de ventas.
TRANSFERIR	Transferir información estadística de ventas del archivo BF al archivo NM. Actualiza campos del archivo BF.
ACTUALI7	Hacer correcciones al Archivo de Trabajo, en caso de que se hayan presentado problemas al momento de hacer las facturas.
LISTAERR	Despliega el mensaje de error de acuerdo con el valor del parámetro.

PSEUDOCODIGO.

(* Inicia el procedimiento BUSCAFUN *)

INICIA (BUSCAFUN)

En caso que FUNDESEA

```
sea igual a 'A','a': Inicia (FUNDESEA='A')
                    Activa DUPLICAR
                    LIMIVA <--- LIMMAST
                    fin (FUNDESEA='A');
```

o

sea igual a 'Q','q': Da por terminada la sesión

sino

Inicia (sino)

Si LIMIVA es mayor que cero entonces

Inicia (si)

En caso que FUNDESEA

```
sea igual a 'B','b': Activa LECOPIA
```

o

```
sea igual a 'C','c': Activa FACT
```

o

```
sea igual a 'D','d': Activa CORTE
```

o

```
sea igual a 'E','e': Inicia (FUNDESEA='E')
                    Activa ACTUALI7
                    Dar por terminada la sesión
                    fin (FUNDESEA='E')
```

o

```
sea igual a 'F','f': Inicia (FUNDESEA='F')
  Si LIMIVA=LIMMAST entonces
    Inicia (si)
    Activa REGRESA
    Si LIMINM=LIMMAST entonces
      Activa TRANSFERIR
    sino
      Activa LISTAERR(6)
    fin (si)
  sino
    Inicia (sino)
    Activa LISTAERR(7)
    Dar por terminada la
    sesión
    fin (sino)
  fin (FUNDESEA='F')

sino
  Activa LISTAERR(5)
fin (caso)

fin (si)
sino
  Activa LISTAERR(4)
fin (sino)

fin (caso)

FIN (BUSCAFUN)
```

10.9 NOMBRE DEL PROCEDIMIENTO: CABEZA

PROGRAMA DONDE SE LOCALIZA: PROY1

OBJETIVO: Tomar del archivo NM los datos de identificación de la farmacia.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADÍSTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.
BLANCNOM	Contiene el resultado de 30 menos la longitud del nombre de la empresa; es decir, son los espacios en blanco que quedan al final del nombre de la farmacia. Se usa para centrar la impresión del nombre. Es de tipo entero.
NOMBREMP	Contiene el nombre de la empresa. Es una cadena de 30 caracteres.

VARIABLES MODULARES: Ninguna.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder al archivo NM. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada la sesión.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.

PSEUDOCODIGO.

(* Inicia el Procedimiento CABEZA *)

INICIA (CABEZA)

```
I <--- 0
Leer el registro I del archivo NM
(* Valida la lectura *)
Activa a ERRCHK
  Si ERROR es verdadero entonces
    Activa EXIT
  fin (si)
NOMBREMP <--- NM.NOMBRE
BLANCONOM <--- 30 - longitud de NOMBREMP
```

FIN (CABEZA)

III.2.3.2 DOCUMENTACION DEL MODULO: DUPLICAR

PROGRAMA DONDE SE LOCALIZA: MODU1

1. OBJETIVO: Obtener una copia parcial del Archivo Maestro para ser usada en los módulos del Subsistema "CONTROL DE VENTAS". El nuevo archivo se llama Archivo de Trabajo, grabado en el disco B con el identificador TRABAJO.FIL y reconocido por el sistema como VA.

2. FUNCIONES PRINCIPALES:

- Asignar los Archivos Maestro y de Trabajo al módulo DUPLICAR.
- Validar la fecha del Archivo de Trabajo.
- Duplicar parcialmente al Archivo Maestro. Incluye información considerada como de alta seguridad.
- Cerrar los dos archivos usados.

3. NARRATIVA DEL MODULO.

Para que el módulo DUPLICAR realice su función adecuadamente, deben estar presentes los archivos Maestro grabado en el disco con el nombre MAESTRO.FIL y reconocido por el sistema como BF y de Trabajo, grabado con el nombre TRABAJO.FIL y reconocido como VA.

Como primer punto, se analiza que el Archivo de Trabajo exista y que además esté vacío. Esto es muy importante, ya que no se permite encimar información, lo que pondría en Jueco la seguridad del sistema. Por lo tanto, el Archivo VA debe existir pero sin información. Para que se cumpla lo anterior, la variable LIMIVA (variable externa entera) debe ser igual a cero.

Si LIMIVA es igual a cero, entonces se procede a asignar los archivos BF y VA.

Hecho lo anterior, se tienen todas las condiciones para que trabaje el módulo de manera óptima. A continuación se inicia el proceso de duplicar parcialmente todos los registros contenidos en BF a VA.

El primer registro que procede a duplicar es el registro de alta seguridad, el cual contiene información de enorme importancia para el sistema. Cuando entra a este registro, el sistema procede a solicitar la fecha de ese día de trabajo, en la forma año, mes, día, un dato por renglón. La valida. Solicita la ratificación de la fecha por parte del usuario. Si la fecha es aceptada como verdadera, entonces continúa mientras no se acepte la fecha no podrá pasarse a otro sección. Es responsabilidad única del usuario dar la fecha correcta.

La fecha proporcionada por el usuario es grabada en el Archivo de Trabajo (VA) en el registro de alta seguridad y esta fecha será la que maneje el módulo en todos los procedimientos. Aparte de la fecha, el módulo duplica otros datos como claves de seguridad y de uso, etc, que están en el registro de alta seguridad del archivo BF.

Teniendo los archivos presentes y la fecha, entonces el sistema procede a duplicar parcialmente, todos los registros que contienen información, del Archivo Maestro al Archivo de Trabajo. La duplicación es parcial ya que únicamente son necesarios algunos campos para elaborar facturas y obtener los cortes de caja de un día de trabajo.

Finalmente se cierran los dos archivos usados.

Es responsabilidad del usuario, proporcionar los dos archivos al sistema.

4. ARCHIVOS QUE UTILIZA EL MODULO.

NOMBRE DEL PROGRAMA	ARCHIVO DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima que contiene información en el Archivo Maestro. Es de tipo entero.
LIMIWA	Número de la clave máxima que contiene información en el Archivo de Trabajo. Es de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo Maestro. Es de tipo booleano.
ANOEX	Año de la fecha en la que se está elaborando el Archivo de Trabajo. Es de tipo entero.
MESEX	Mes de la fecha en la que se está elaborando el Archivo de Trabajo. Es de tipo entero.
DIAEX	Día de la fecha en la que se está elaborando el Archivo de Trabajo. Es de tipo entero.

5.2 VARIABLES MODULARES.

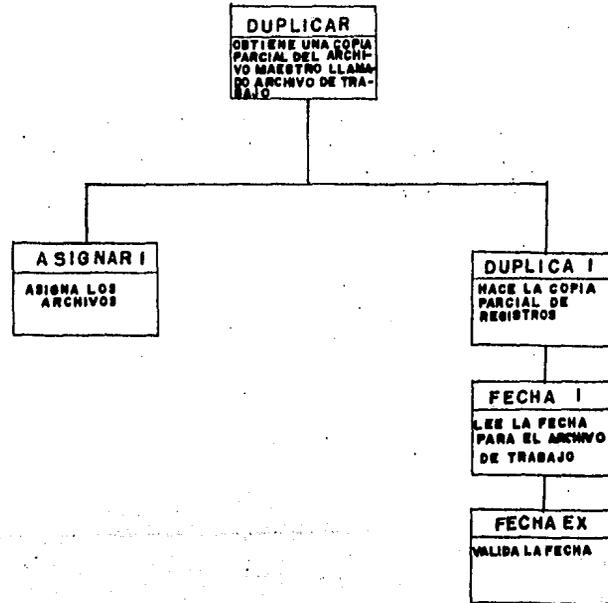
NOMBRE	DESCRIPCION
J	Apuntador para acceder a los dos archivos. Es de tipo entero.
VERIFI	Variable usada para ratificar la fecha. Puede valer S o N, afirmativo o negativo respectivamente. Es de tipo caracter.

5.3 VARIABLES LOCALES. Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

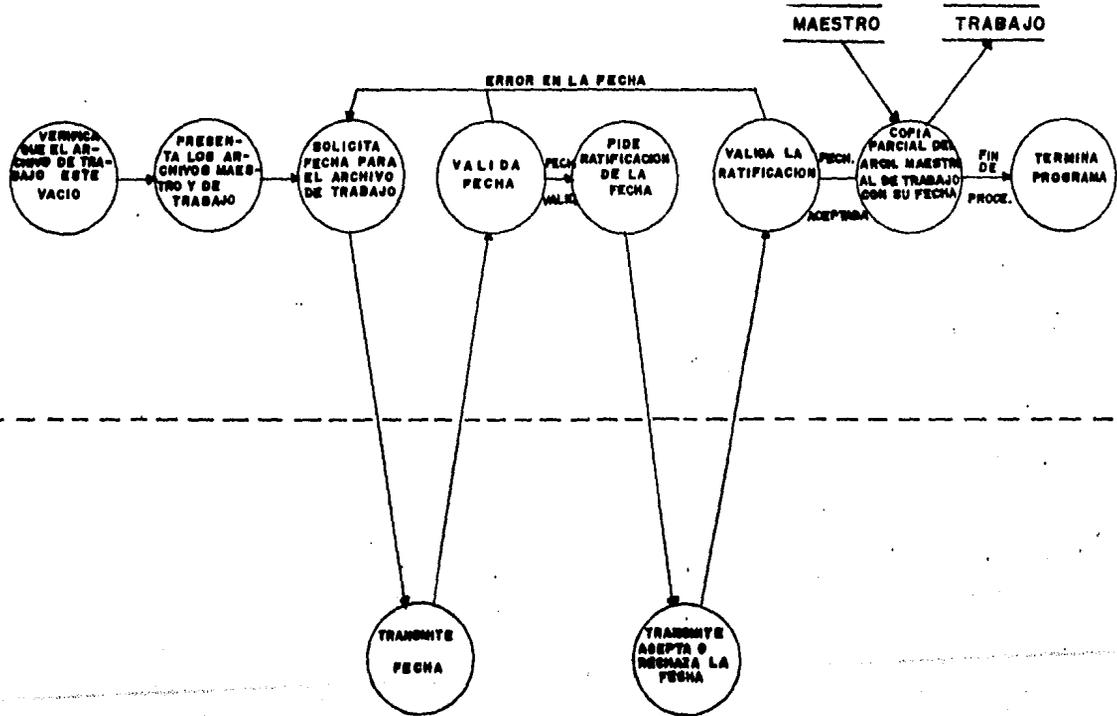
NOMBRE	OBJETIVO
ASIGNAR1	Asigna el Archivo Maestro (BF) y el Archivo de Trabajo (VA) al módulo para que realice sus funciones. Coloca los apuntadores en el primer registro disponible de cada archivo e inicializa variables.
FECHA1	Solicita la fecha del archivo, la valida activando a FECHAEX. Finalmente existe la ratificación de la fecha por parte del usuario, leyendo la variable VERIFI. Inicializa los valores del registro de alta seguridad, el cual contiene información de gran importancia para el sistema, como son las claves de seguridad y de uso, así como el máximo número de registros en el archivo.
DUPLICA1	Copia del Archivo Maestro la información contenida en los campos: CLAVE, MS1, MS2, MS3, PRECIO, UNID, IMP, UPED, UVP1 al Archivo de Trabajo, en los siguientes campos respectivamente: CLAVE, MEDS1, MEDS3, MEDS3, PRECIO, UNID, IMP, UPE1, UVPD, inicializando el campo UVEN de este archivo con cero. Escribe esta información. Cierra al Archivo de Trabajo.
FECHAEX	Leer y validar la fecha proporcionada por el usuario.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Da por terminada la sesión.

DIAGRAMA JERARQUICO



L 255 1

DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

E 286 J

USUARIO

9. PSEUDOCODIGO.

(* Inicia el módulo DUPLICAR *)

INICIA (DUPLICAR)

Si LIMIVA es igual a cero entonces

Inicia (Si)

Activa a ASIGNAR1

Activa a DUPLICAR1

fin (si)

sino

Inicia (sino)

Mensaje :

" *** E R R O R ***

EL ARCHIVO DE TRABAJO YA CONTIENE INFORMACION.
POR LO TANTO, NO PUEDE USAR ESTA FUNCION."

fin(sino)

FIN (DUPLICAR).

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: ASIGNAR1

PROGRAMA DONDE SE LOCALIZA: MODU1

OBJETIVO: Asigna el Archivo Maestro (BF) y el Archivo de Trabajo (VA) al módulo DUPLICAR para que realice sus funciones. Coloca los apuntadores en el primer registro disponible de cada archivo e inicializa variables.

ARCHIVOS QUE UTILIZA EL MODULO.

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS. Ninguna.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
VERIFI	Variable usada para ratificar la fecha. Puede valer S o N, afirmativo o negativo respectivamente. Es de tipo caracter.

VARIABLES LOCALES. Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento ASIGNAR1 *)

INICIA (ASIGNAR1)

(* Presenta al Archivo Maestro y al Archivo de Trabajo *)
Asigna al Archivo Maestro BF.
Asigna al Archivo de Trabajo VA.
(* Inicializa variables *)
VERIFI <---

FIN (ASIGNAR).

10.2 NOMBRE DEL PROCEDIMIENTO: FECHA1

PROGRAMA DONDE SE LOCALIZA: MODU1

OBJETIVO: Validar la fecha del Archivo de Trabajo. Grabar información trascendental para el subsistema en un registro de alta seguridad (registro cero).

ARCHIVOS QUE USA EL PROCEDIMIENTO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
ANDEX	Año de la fecha en la que se está elaborando el Archivo de Trabajo. Es de tipo entero.
MESEX	Mes de la fecha en la que se está elaborando el Archivo de Trabajo. Es de tipo entero.
DIAEX	Día de la fecha en la que se está elaborando el Archivo de Trabajo. Es de tipo entero.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
VERIFI	Variable usada para ratificar la fecha. Puede valer S o N, afirmativo o negativo respectivamente. Es de tipo caracter.

VARIABLES LOCALES. Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
FECHAEX	Leer y validar la fecha proporcionada por el usuario.

PSEUDOCODIGO.

(* Inicia el procedimiento FECHA1 *)

INICIA (FECHA1)

Repite

(* Lee y valida la fecha *)

Activa a FECHAEX

(* Solicita la ratificación de los datos *)

Escribe el mensaje:

'REALMENTE ESTA CORRECTA LA FECHA ? ANOTE S)I O N)O'

Lee VERIFI

Hasta que VERIFI sea igual a 'S' o 's'.

(* Inicializa los campos del registro cero, llamado de alta seguridad en el Archivo de Trabajo *)

VA.CLAVE <--- 0

(* Escribe los datos de la fecha en ese registro *)

VA.MEDS1 <--- ANDEX

VA.MEDS2 <--- MESEX

VA.MEDS3 <--- DIAEX

(* Inicializa campos *)

VA.PRECIO <--- 0

(* Copia el dato de la clave máxima contenida en el Archivo Maestro *)

VA.UNID <--- BF.MS1

(* Copia la primera contraseña de seguridad *)

VA.IMP <--- BF.IMP

(* Inicializa el número de la factura *)

VA.UVEN <--- 1

(* Copia la segunda contraseña de seguridad *)

VA.UPE1 <--- BF.UVP1

(* Inicializa el campo que maneja la clave de uso del Archivo de Trabajo *)

VA.UVPD <--- 1

FIN (FECHA1)

10.3 NOMBRE DEL PROCEDIMIENTO: DUPLICA1

PROGRAMA DONDE SE LOCALIZA: MODU1

OBJETIVO: Tomando como patrón al Archivo Maestro, obtiene una copia parcial de éste, llamado Archivo de Trabajo.

ARCHIVOS QUE USA EL MODULO.

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima que contiene información en el Sistema. Es de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo Maestro. Es de tipo booleano.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder a los dos archivos. Es de tipo entero.

VARIABLES LOCALES.

NOMBRE	DESCRIPCION
INDICE	Indice para un proceso iterativo; asume el valor desde cero hasta la máxima clave dada por la variable LIMMAST. Es de tipo entero.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Da por terminada la sesión.
FECHA1	Valida la fecha y graba información importante en el registro cero de VA.

PSEUDOCODIGO

(* Inicia el procedimiento DUPLICA1 *)

INICIA (DUPLICA1)

Escribe el mensaje:
"EL SISTEMA INICIA EL PROCESO PARA OBTENER EL
ARCHIVO DE TRABAJO A PARTIR DEL ARCHIVO MAESTRO"

```

*SE ESTA DUPLICANDO EL ARCHIVO.
ESPERE LA SENAL PARA PODER CONTINUAR ...*
Haz desde que INDICE = 0 hasta LIMMAST lo siguiente
  Inicia (Haz)
  Asigna a J el valor de INDICE
  Leer del Archivo BF el registro J
  (* Verifica la lectura *)
  Activa a ERRCHK
  (* Como resultado de la verificación resessa el valor de
  ERROR *)
  Si ERROR es verdadero entonces
    Activa a EXIT
  fin (Si)
  Si INDICE es diferente de cero entonces
    Inicia (Si)
    (* Inicia el proceso de copiado *)
    VA.CLAVE <--- BF.CLAVE
    VA.MEDS1 <--- BF.MS1
    VA.MEDS2 <--- BF.MS2
    VA.MED3 <--- BF.MS3
    VA.PRECIO <--- BF.PRECIO
    VA.UNID <--- BF.UNID
    VA.IMP <--- BF.IMP
    (* Inicializa el campo de Unidades Vendidas *)
    VA.UVEN <--- 0
    VA.UPE1 <--- BF.UPED
    VA.UVPD <--- BF.UVP1
    fin (Si)
  sino
    Inicia (sino)
    Activa a FECHA1
    fin (sino)
  (* Se graba en VA lo leído de BF *)
  Escribe en el registro J del archivo VA
  (* Verifica la escritura *)
  Activa a ERRCHK
  Si ERROR es verdadero entonces
    Activa a EXIT
  fin (si)
Fin (HAZ)
Cierra el Archivo VA
Escribe el mensaje:
'EL SISTEMA TERMINO DE DUPLICAR, PUEDE CONTINUAR'

```

FIN (DUPLICA1)

III.2.3.3 DOCUMENTACION DEL MODULO: LEECOPIA

PROGRAMA DONDE SE LOCALIZA : MODU2

1. OBJETIVO: Permitir la lectura de cualquier registro del Archivo de Trabajo del día, para hacer consultas rápidas.

2. FUNCIONES PRINCIPALES:

- Asignar el Archivo de Trabajo.
- Desplegar el encabezado de la información.
- Solicitar la clave del medicamento que desea consultar el usuario.
- Validar la clave de la medicina.
- Leer cualquier registro que señale el usuario, por medio de la clave del medicamento.
- Desplegar los datos del medicamento.

3. NARRATIVA DEL MODULO:

Antes de iniciar sus actividades, es necesario verificar la existencia del Archivo de Trabajo, grabado con el nombre TRABAJO.FIL en el disco B y reconocido por el sistema como VA. Sin este archivo, no trabaja el módulo LEECOPIA. El programa manejador PROY1 realiza esta función. Si el archivo existiese pero vacío, PROY1 lo detecta e impide el uso de este módulo.

Si existe el Archivo de Trabajo y contiene información, lo primero que hace el Sistema es asignarlo al módulo LEECOPIA. Coloca al apuntador en el primer registro disponible, y está listo para iniciar sus funciones.

Este módulo sirve para leer cualquier registro del Archivo de Trabajo. La lectura puede efectuarse desde pantalla o en impresora. Como es de consulta rápida, se susiere que la lectura se haga en pantalla.

Existe un registro de alta seguridad, en el cual se encuentran grabados datos de enorme importancia para el sistema. Este registro es de acceso restringido. El procedimiento REVISA tiene por objeto acceder al registro de alta seguridad para consultar su contenido. De esta manera, despliega la fecha del Archivo de Trabajo, para que el usuario la conozca. Este procedimiento obtiene el valor del límite máximo de registros con información contenida en VA y lo asigna a la variable LIMITE.

Por medio del procedimiento CABEZA2, el sistema despliega el encabezado del listado. En el primer renglón aparece el nombre de la empresa y la fecha del Archivo de Trabajo. El nombre es transferido al módulo por una variable externa. La fecha la toma del registro de alta seguridad. A continuación, escribe la identificación de las columnas que corresponden a los nombres de los campos del registro, cuya información aparecerá en el listado. Este encabezado es desplegado cada 13 lecturas o intentos de lecturas a los registros.

Después del encabezado, el sistema invoca a BUSCA2, el cual es el encargado de leer el registro que le interesa consultar al usuario. Así, anota el signo de interrogación y el de número (? é). Esta es la señal de que ya está listo para leer la clave del medicamento, cuya información desea conocer el usuario.

Tomando como dato la clave del medicamento, el sistema busca en el Archivo de Trabajo al registro que contiene esta clave y despliega la información correspondiente. La búsqueda es de acceso directo. Este procedimiento se realizará siempre y cuando la clave se encuentre dentro del rango permitido, el cual estará limitado por 1 en su valor mínimo y por la variable LIMITE, en su valor máximo. El sistema valida la lectura del registro, por lo que si se comete algún error, éste es detectado y especificado por el sistema valiéndose de la función ERRCHK.

El Módulo LEECOPIA dejará de realizarse cuando el usuario a la pregunta de cual clave de registro desea consultar, conteste con un caracter <Q>.

Por medio de este módulo, el sistema realiza consultas al Archivo de Trabajo, de una forma rápida, listando la información contenida en todos los campos del registro solicitado.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo de Trabajo. Es de tipo booleano.
BLANCNOM	Contiene el resultado de 30 menos la longitud del nombre de la empresa; es decir, son los espacios en blanco que quedan al final del nombre de la farmacia. Se usa para centrar la impresión del nombre. Es de tipo entero.
NOMBREMP	Contiene al nombre de la farmacia. Es de tipo cadena de caracteres.

5.2 VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder al Archivo de Trabajo. Es de tipo entero.
LIMITE	Número de la clave máxima que contiene información en el Archivo de Trabajo. Es de tipo entero.
SIG	Variable que se usa para presuntar si desea continuar con la lectura de registros. Es de tipo caracter.

5.3 VARIABLES LOCALES. Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
CABEZA2	Desplegar el encabezado, que incluye; Nombre de la empresa, fecha e identificadores de columnas.
REVISAS2	Realizar la lectura del registro de alta seguridad del Archivo de Trabajo para obtener el valor de la clave máxima que contiene información (LIMITER) y fecha. Resresa como parámetro el valor de LIMITER a LEECOPIA.
BUSCA2	Realizar la búsqueda del registro solicitado, para que de acuerdo con ciertas condiciones, se lleve a cabo la lectura. La condición principal que debe cumplirse, es que el registro solicitado debe estar dentro del rango.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.

DIAGRAMA JERARQUICO

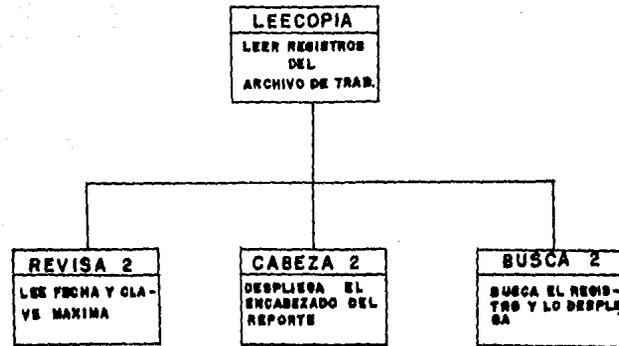
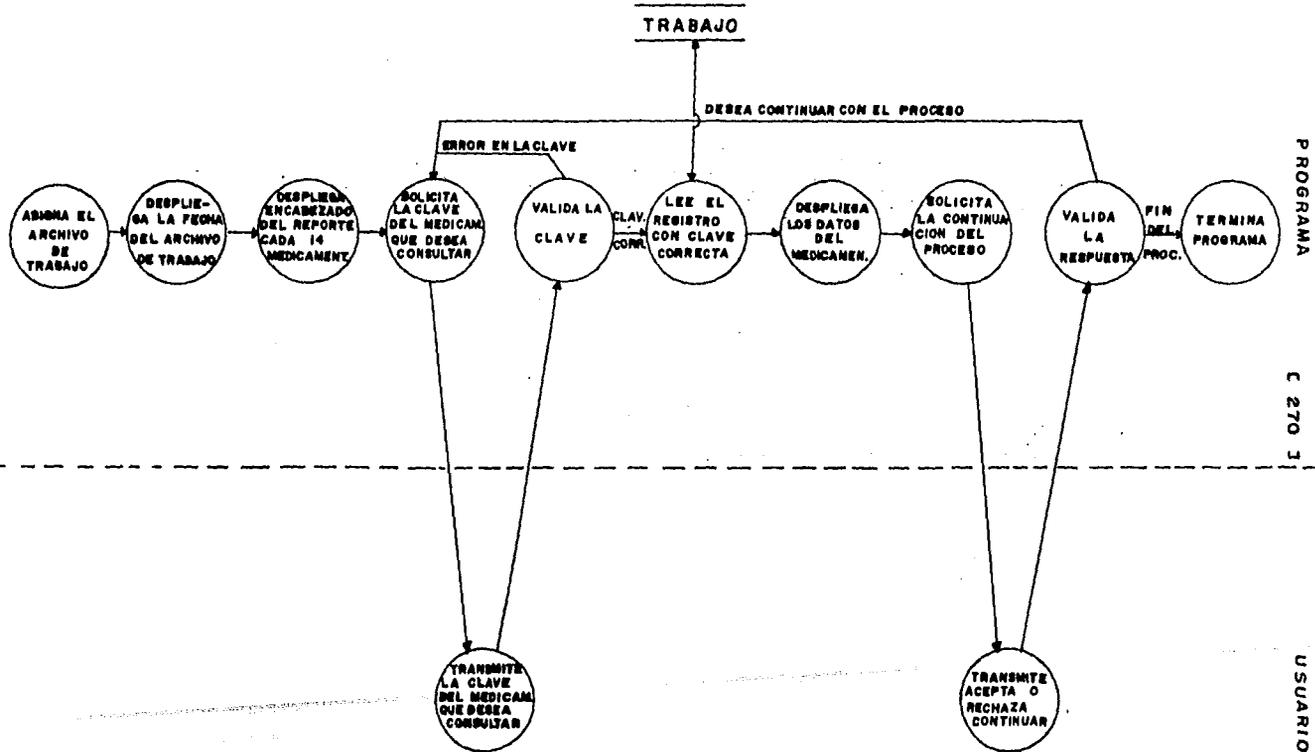


DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO.

(* Inicia el módulo LEECOPIA *)

INICIA (LEECOPIA)

(* Presenta el archivo VA y coloca el apuntador en su primer registro, listo para leer *)

Asigna el Archivo de Trabajo VA

(* Invoca al procedimiento REVISA2 pasando como parámetro el valor de LIMITE *)

Activa a REVISA2(LIMITE)

Despliega el mensaje:

"COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA"

"PARA CONTINUAR PRESIONE <RETURN>"

Lee SIG

(* Lista los encabezados del reporte *)

Activa a CABEZA2

(* Realiza la lectura de los registros que solicite el usuario *)

Activa a BUSCA2

FIN (LEECOPIA)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: REvisa2

PROGRAMA DONDE SE LOCALIZA: MODU2.

OBJETIVO: Realizar la lectura del registro cero, llamado de alta seguridad del Archivo de Trabajo para obtener el valor de la clave máxima que contiene información y la fecha. Como dato final se obtiene el valor de LIMITER.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo de Trabajo. Es de tipo booleano.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder al Archivo de Trabajo. Es de tipo entero.

VARIABLES LOCALES.

NOMBRE	DESCRIPCION
LIMITER	Número de la clave máxima que contiene información en el Archivo de Trabajo. Es un parámetro. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA :

NOMBRE	OBJETIVO
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
ERRCHK	Verificar lecturas y escrituras al archivo. Es de tipo externo.
EXIT	Dar por terminada la sesión. Es de tipo externo.

PSEUDOCODIGO.

(* Inicia el procedimiento REVISA2 *) *

INICIA (REVISA2)

```

(* Inicializa el valor de J *)
J <--- 0
(* Llama a un procedimiento de lectura *)
Leer del Archivo VA el registro J
(* Llama a un procedimiento que verifica la lectura *)
Activa a ERRCHK
(* Resresa el valor de la verificación por medio de ERROR *)
Si ERROR es verdadero entonces
  (* Si hay error en la lectura *)
  Activa a EXIT
fin (Si)
Despliega el mensaje:
"ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA : "
(* Los valores de ANO, MES y DIA están contenidos en los campos
MEDS1, MEDS2 y MEDS3 respectivamente del registro de sita
seguridad del Archivo de Trabajo VA *)
"ANO : (MEDS1) MES : (MEDS2) DIA : (MEDS3)"
(* Toma el valor de la clave máxima del campo UNID de VA *)
(* LIMITER es la variable que resresa al módulo que lo invoca *)
LIMITER <--- VA.UNID

```

FIN (REVISA2)

10.2 NOMBRE DEL PROCEDIMIENTO: CABEZA2.

PROGRAMA DONDE SE LOCALIZA: MODU2.

OBJETIVO: Desplegar el encabezado del reporte, que incluye: Nombre de la empresa, fecha e identificadores de columnas.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo de Trabajo. Es de tipo booleano.
BLANCNOM	Contiene el resultado de 30 menos la longitud del nombre de la empresa; es decir, son los espacios en blanco que quedan al final del nombre de la farmacia. Se usa para centrar la impresión del nombre. Es de tipo entero.
NOMBREMP	Contiene al nombre de la farmacia. Es de tipo cadena de caracteres.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
	Apuntador para acceder al Archivo de Trabajo. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
ERRCHK	Verifica lecturas y escrituras al archivo. Sin parámetros. Es de tipo externo.
EXIT	Dar por terminada la sesión. Es de tipo externo.

PSEUDOCODIGO

(* Inicia el procedimiento CABEZA2 *)

INICIA (CABEZA2)

```
(* Inicializa el valor de J *)
J <--- 0
(* Lee en el archivo VA el registro J *)
Leer del Archivo VA el registro J
(* Verifica la lectura *)
Activa a ERRCHK
(* ERRCHK regresa como resultado el valor de ERROR *)
Si ERROR es verdadera entonces
  (* Si la lectura fue erronea, se sale del procedimiento *)
  Activa a EXIT
fin (Si)
(* Escribe el encabezado del reporte *)
Despliega el mensaje:
(* El nombre de la empresa está en NOMBREMP *)
"(nombre de la empresa) ARCHIVO DE TRABAJO  FECHA : "
Despliega el mensaje:
```

```
=====
CLAVE:P R E C I O:IMP ; U N I D A D E S
      : D E V E N T A : % ; E N : V E N D I D A S : P E D I D A S : V E N D I D A S
      : S I N I M P C O N I M P ; E X I S T E N C I A : ( A C T ) : ( U L R E ) : D E L P E D .
=====
```

FIN (CABEZA2)

10.3 NOMBRE DEL PROCEDIMIENTO: BUSCA2.

PROGRAMA DONDE SE LOCALIZA: MODU2.

OBJETIVO: Realizar la búsqueda del registro solicitado, para que de acuerdo con ciertas condiciones, se lleve a cabo la lectura. La condición principal que debe cumplirse, es que el registro solicitado debe estar dentro del rango permitido por el valor de LIMITE.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas al Archivo de Trabajo. Es de tipo booleano.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder al Archivo de Trabajo. Es de tipo entero.
LIMITE	Valor de la clave máxima del registro que contiene información en el Archivo de Trabajo. Es de tipo entero.
SIG	Variable que se usa para preguntar si desea continuar con la lectura de registros. Es de tipo caracter.

VARIABLES LOCALES.

NOMBRE	DESCRIPCION
PRENET	Es el precio del medicamento, incluyendo el impuesto respectivo. Es de tipo real.
RENGLON	Lleva la cuenta del número de veces que se ejecuta o intenta hacer una lectura de registros; cada 13 lecturas o intentos, activa a CABEZA2. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CABEZA2	Desplegar el encabezado, que incluye: Nombre de la empresa, fecha e identificadores de columnas.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
ERRCHK	Verifica lecturas y escrituras al archivo. Sin parámetros. Es de tipo externo.
EXIT	Dar por terminada la sesión. Es de tipo externo.

PSEUDOCODIGO.

(* Inicia el Procedimiento BUSCA2 *)

INICIA (BUSCA2)

(* Inicializa el valor de SIG con un caracter blanco *)

SIG <--- ' '

REGLON <--- 1

(* Entra a un proceso repetitivo *)

Repite

Si REGLON es igual a 14 entonces

Inicia (si)

Activa a CABEZA2

REGLON <--- 1

fin (si)

(* Solicita el valor de la clave del medicamento a ser leído *)

Despliega el mensaje:

'? é'

(* Lee el valor de la clave del medicamento y asigna el valor a J *)

Lee J

(* El valor de J debe estar en el rango permitido, es decir debe ser mayor que 0 y menor o igual que LIMITE *)

Si J es mayor que 0 y menor o igual que LIMITE entonces

Inicia (si)

(* Si está dentro del rango se activa el proceso para leer *)

Leer del Archivo VA el registro J

(* Se verifica la lectura *)

Activa a ERRCHK

(* Como resultado de ERRCHK se obtiene el valor de ERROR *)

Si ERROR es verdadera entonces

(* En caso de error en la lectura se retorna al módulo manejador *)

Activa a EXIT

fin (si)

(* Si todo está correcto, lista el contenido de los campos del registro accesado *)

Despliega los valores contenidos en los campos:

CLAVE, PRECIO, PRECIO*IMP, IMP*100, UNID, UVEN, UPE1 Y UVPD

fin (si)

sino

Inicia (sino)

(* En caso de que se haya dado una clave fuera del rango permitido *)

Despliega el mensaje:

'NO EXISTE, EL REGISTRO (valor de J) ESTA FUERA DEL RANGO (valor de LIMITE)'

fin (sino)

REGLON <--- REGLON + 1

(* Sin escribir mensaje, solicita y lee el valor de SIG para saber si continúa con el procedimiento BUSCA2 o termina *)

Lee SIG

(* El procedimiento BUSCA2 dejará de ejecutarse hasta que

SIG tenga un valor de 'Q' o 'q'; en caso contrario sigue *)

Hasta que SIG sea igual a 'Q' o 'q'

FIN (BUSCA2)

III.2.3.4 DOCUMENTACION DEL MODULO: FACTURAR

PROGRAMA DONDE SE LOCALIZA: MODU3

1. OBJETIVO. Elaborar la factura del cliente, actualizando el Archivo de Trabajo.

2. FUNCIONES PRINCIPALES:

- Asignar el Archivo de Trabajo, verificando su existencia y contenido.
- Hacer los cálculos necesarios para elaborar las facturas.
- Manejar información de los medicamentos y artículos contenidos en el sistema.
- También pueden facturarse aquellos artículos que están fuera del sistema, llamados adicionales. Sus datos son proporcionados por el usuario.
- Proporcionar la factura del cliente.
- Actualizar el archivo con las ventas generadas durante una sesión.

3. NARRATIVA DEL MODULO.

Primero se asigna el archivo de trabajo, para que esté presente la información necesaria para elaborar las facturas. Esta función factura aquellos medicamentos cuyos datos están contenidos en el archivo, pero también lo hace de aquellos productos que están fuera del mismo.

Se verifica que este Archivo de Trabajo no haya sido usado enteramente para actualizar al Archivo Maestro, por medio de una clave de uso. Si ya fue usado para tal fin, entonces no se permite continuar con la función. En caso contrario, se procede a presuntar al usuario si el archivo le sirve, tomando como referencia la fecha y si desea continuar. Únicamente si las dos respuestas son afirmativas, continúa el proceso.

En el caso de los medicamentos, los únicos datos que recibe del exterior son, la clave de la medicina y la cantidad demandada, así como la confirmación de la venta. Todo lo demás lo toma del registro propio del medicamento. Realiza un acceso directo a los registros.

En caso de que el medicamento solicitado, esté agotado, se listan tres medicamentos similares, para que el cliente elija uno, si así lo desea.

Para los productos adicionales, es necesario que el usuario proporcione sus datos, como el precio, las unidades demandadas, el impuesto y la confirmación de la venta.

El módulo elabora la factura y actualiza en ese mismo instante al Archivo de Trabajo.

Se lleva un control sobre el número de facturas y el total acumulado por factura y de manera general.

4. ARCHIVOS QUE USA EL MODULO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
ERROR	Variable usada para validar las lecturas y escrituras al archivo de Trabajo. Es de tipo booleano.

5.2 VARIABLES MODULARES.

NOMBRE	DESCRIPCION
CONTINUA	Valor que se usa para ratificar algunos datos proporcionados del exterior. Es de tipo caracter.
J	Valor del indice para acceder a los registros. Es de tipo entero.
AH	Año de la fecha del Archivo de Trabajo. Es de tipo entero.
MH	Mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DH	Día de la fecha del Archivo de Trabajo. Es de tipo entero.

AXMXD	Valor del producto de AH por MH por DH. Es de tipo entero.
NUMFAC	Valor del contador de facturas; cada factura tiene un número. Es de tipo entero.
LIMIT3	Clave máxima del medicamento contenido en VA. Es de tipo entero.
USO	Contiene el valor de uso del archivo. Este valor cambia cuando VA actualiza a BF, con el fin de ya no permitir a este VA volver a facturar ni actualizar. Es de tipo entero.
UNIDACT	Valor de las unidades en existencia actuales. Es de tipo entero.
PRECMED	Valor del precio de la medicina. Es de tipo real.
IMP MED	Valor del porcentaje de impuesto. Es de tipo real.
UNIVEN	Valor de las Unidades Vendidas Actuales. Es de tipo entero.
UNIDEPE	Valor de las Unidades en Existencia del Pedido. Es de tipo entero.
UNVENPED	Valor de las Unidades Vendidas del Pedido. Es de tipo entero.
SIGUE	Valor de la respuesta del usuario a preguntas del sistema, para ratificar datos o continuar con el proceso. Es variable de control de tipo carácter.
TOTACUM	Total acumulado, después de la aceptación o rechazo de la venta. Es de tipo real.
SUBTPRO	Total acumulado por factura. Es de tipo real.

PREXUN Valor de PRECMED X UNISOLI. Es de tipo real.

PRXUNXIM Valor de PREXUN X IMPMED. Es de tipo real.

TOTXMED Valor de PREXUN + PRXUNPIM. Es de tipo real.

SUB Valor del total de la venta por producto. Es de tipo real.

ACCEPTA Valor que indica si la venta de la medicina o producto adicional es aceptada por el cliente. Es de tipo caracter.

UNIRES Resultado de hacer UNIDACT menos UNISOLI. Es de tipo entero.

TOTADGRA Contiene las ventas totales de productos adicionales, dato que se graba en disco. Es de tipo real.

TOTADXSE Ventas totales de productos adicionales por sesión, esto es posible, ya que un archivo VA puede servir para varias sesiones de ventas. Es de tipo real.

TOADXFA Valor acumulativo de los productos adicionales por factura. Es de tipo real.

UNISOLI Valor de las unidades demandadas por el cliente. Es de tipo entero.

SALTE Valor que usa la estructura de control repetitiva en la condición para verificar la cantidad. Es de tipo entero.

5.3 VARIABLES LOCALES. Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO.

NOMBRE	OBJETIVO
CHECA3	Realiza las lecturas al archivo y activa el procedimiento para validarlas.
ENTRADA3	Asignar el Archivo de Trabajo para iniciar la sesión de ventas. Toma información del registro de alta seguridad, para proporcionarla al módulo. Presenta la fecha del archivo e inicia o reinicia al contador de facturas.
DATOS3	Asigna los datos que existen en el registro leído a las variables que se usan en el proceso.
SIMILAR3	En caso de que no exista la medicina demandada por el cliente, despliega una lista de tres medicinas similares, para que el cliente decida cual de ellas le conviene. En caso de que solicite alguna de las presentadas, se verifican los datos tanto de clave como de cantidad, para ser procesadas por el procedimiento FACTMED3.
FACTMED3	Calcula la venta de cada producto y actualiza el registro correspondiente.
CEROS3	Inicializa variables con ceros.
ADICION3	Elabora la factura con productos adicionales, cuyos datos no están en el Archivo de Trabajo.
IMFINFAC	Termina la factura, obteniendo el total de la venta, recibiendo el dinero que pasa el cliente, calcula el cambio y despliega estos datos.
CERRAR3	Guarda los datos más relevantes en el registro cero.

FACTURA3 Inicializa las variables del proceso. Lista el encabezado de la factura. Activa a CEROS3. Lee los datos del medicamento, clave y cantidad, uno por uno. Valida los datos. Activa a FACTMED3. Continúa leyendo datos o elabora el final de la factura. Activa a ADICION3 si se desea continuar la facturación con productos adicionales. Activa a IMFINFAC para el total de la venta. Activa a CERRAR3.

ERRCHK Valida lecturas y escrituras a los archivos.

SEEKREAD Realiza lecturas de acceso directo a los archivos.

SEEKWRITE Realiza escrituras en los archivos.

EXIT Dar por terminada la sesión.

DIAGRAMA JERARQUICO

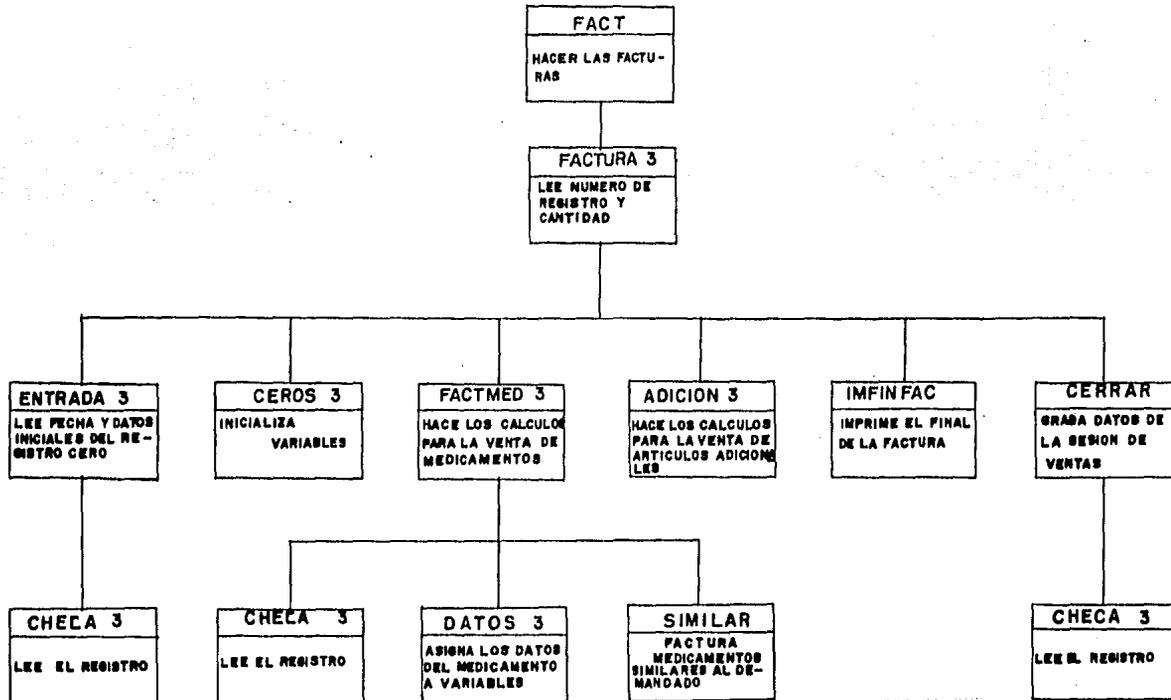


DIAGRAMA DE FLUJO DE DATOS

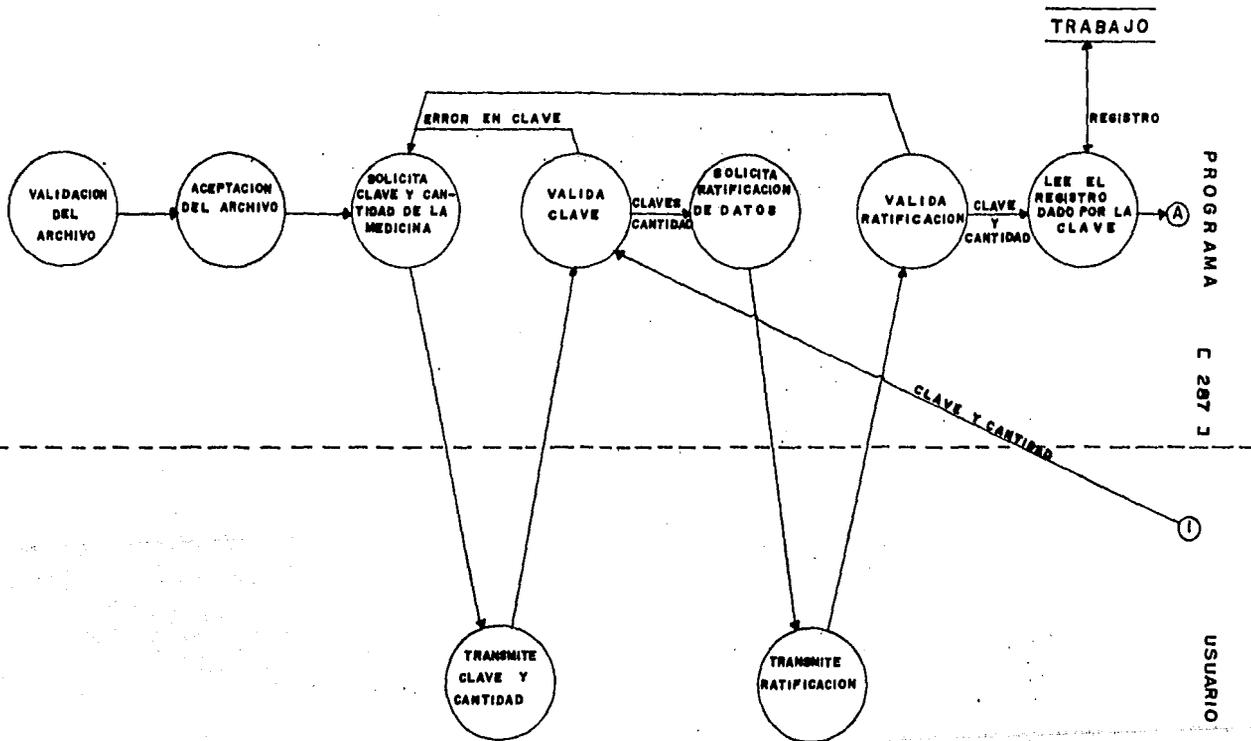
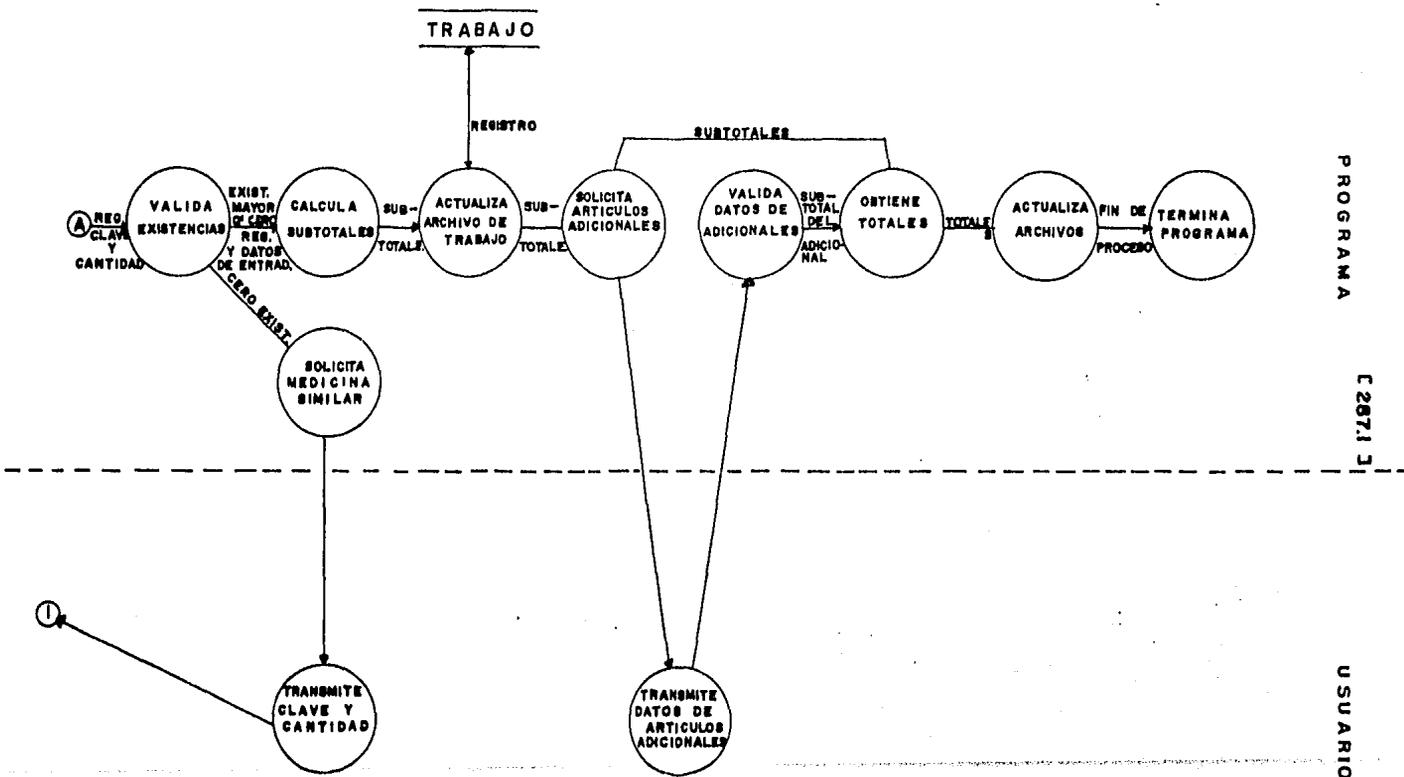


DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO,

(* Inicia el módulo FACTURAR *)

INICIA (FACTURAR)

(* Llama al procedimiento ENTRADA3 *)

Activa a ENTRADA3

(* Valida fecha y clave máxima *)

Si AXDXM es igual a cero o

LIMITE3 es igual a cero entonces

Despliega el mensaje:

"ERROR LA FECHA DEL ARCHIVO DE TRABAJO CONTIENE UN CERO.
LISTE EL MENU PRINCIPAL DE FUNCIONES, ELIJA LA
FUNCION <Q>, PARA SALIR DEL SUBSISTEMA Y CONSULTE
EL MANUAL CORRESPONDIENTE"

sino

Inicia (sino)

(* El archivo tiene la fecha correcta
y además contiene registros con información *)

(* Ahora se verifica que el archivo VA no ha sido usado para
actualizar al archivo BF, con la opción <F> *)

Si USO es igual a 5 entonces

(* El archivo VA ya fue usado para actualizar a BF *)

(* Por lo tanto ya no puede usarse para hacer facturas *)

Despliega el mensaje:

"ESTE ARCHIVO DE TRABAJO YA NO PUEDE USARSE
PORQUE CON EL CORRID LA OPCION <F>,
POR LO TANTO, DEBE CREAR OTRO ARCHIVO DE TRABAJO.
SI TIENE DUDA CONSULTE EL MANUAL"

sino

Inicia (sino)

(* El archivo VA es útil *)

(* Hace dos preguntas fundamentales para continuar *)

Despliega el mensaje:

"L E S I R V E ? S I o N O : "

(* Lee las respuesta con la variable SIGUE *)

Lee SIGUE

Despliega el mensaje:

"DESEA CONTINUAR ? S I o N O : "

(* Lee la respuesta con la variable CONTINUA *)

Lee CONTINUA

(* Verifica las dos respuestas *)

Si SIGUE es igual a 'S' y

CONTINUA es igual a 'S' entonces

(* Si SIGUE y CONTINUA son afirmativas *)

Activa a FACTURA3

sino

(* Cuando alguna de SIGUE o CONTINUA es negativa *)

Escribe el mensaje:

"ELIMINE ESTE ARCHIVO Y OBTENGA
EL ADECUADO INMEDIATAMENTE"

fin (Si)

fin (sino)

FIN (FACTURAR)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: CHECA3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Realiza las lecturas al archivo y activa el procedimiento para validarlas.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable usada para validar las lecturas y escrituras al archivo de Trabajo. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
J	Valor del indice para acceder a los registros. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO

(* Inicia el procedimiento CHECA3 *)

INICIA (CHECA3)

Lee del Archivo VA el registro J
(* Valida la lectura *)

ACTIVA ERRCHK

SI ERROR es verdadero entonces
Activa EXIT

fin (Si)

FIN (CHECA3)

10.2 NOMBRE DEL PROCEDIMIENTO: ENTRADA3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Asignar el Archivo de Trabajo para iniciar la sesión de ventas. Toma información del registro de alta seguridad, para proporcionarla al módulo. Presenta la fecha del archivo e inicia o reinicia al contador de facturas.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
J	Valor del índice para acceder a los registros. Es de tipo entero.

AH Año de la fecha del Archivo de Trabajo. Es de tipo entero.
 MH Mes de la fecha del Archivo de Trabajo. Es de tipo entero.
 DH Día de la fecha del Archivo de Trabajo. Es de tipo entero.
 AXMXD Valor del producto de AH por MH por DH. Es de tipo entero.
 NUMFAC Valor del contador de facturas; cada factura tiene un número. Es de tipo entero.
 LIMIT3 Clave máxima del medicamento contenido en VA. Es de tipo entero.
 USO Contiene el valor de uso del archivo. Este valor cambia cuando VA actualiza a BF, con el fin de ya no permitir a este VA volver a facturar ni actualizar. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CHECA3	Realiza las lecturas al archivo VA y activa el procedimiento para validarlas.

PSEUDOCODIGO

(* Inicia el procedimiento ENTRADA3 *)

INICIA (ENTRADA3)

```

Asigna el archivo VA
J <--- 0
Leer del Archivo VA el registro J
Activa a CHECA3
(* Asigna los datos de la fecha *)
AH <--- VA.MEDS1
MH <--- VA.MEDS2
DH <--- VA.MEDS3
(* Obtiene el producto *)
AXMXD <--- AH*MH*DH
(* Asigna el valor de la máxima clave de VA *)
LIMIT3 <--- VA.UNID
NUMFAC <--- VA.UVEN
(* Asigna la clave de uso del archivo *)
USO <--- VA.UVPD
Despliega el mensaje:
*ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA ;
ANO : XX, MES : XX, DIA : XX
  
```

FIN (ENTRADA3)

10.3 NOMBRE DEL PROCEDIMIENTO: DATOS3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Asignar los datos que existen en el registro leído a las variables que se usan en el proceso. El registro corresponde a la clave del medicamento.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
UNIDACT	Valor de las unidades en existencia actuales. Es de tipo entero.
PRECMED	Valor del precio de la medicina. Es de tipo real.
IMP MED	Valor del porcentaje de impuesto. Es de tipo real.
UNIVEN	Valor de las Unidades Vendidas Actuales. Es de tipo entero.
UNIDEPE	Valor de las Unidades en Existencia del Pedido. Es de tipo entero.
UNVENPED	Valor de las Unidades Vendidas del Pedido. Es de tipo entero.
SIGUE	Valor de la respuesta S o N. Es variable de control de tipo caracter.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO

(* Inicia el procedimiento DATOS3 *)

INICIA (DATOS3)

(* Asigna datos del medicamento, contenidos en el registro *)

(* Unidades en Existencia Actuales *)

UNIDACT = VA.UNID

(* Precio *)

PRECMED = VA.PRECIO

(* Impuesto *)

IMPMEP = VA.IMP

(* Unidades Vendidas Existentes *)

UNIVEN = VA.UVEN

(* Unidades Existentes del Pedido *)

UNIDEPE = VA.UPE1

(* Unidades Vendidas del Pedido *)

UNVENPED = VA.UVPD

SIGUE = 'S'

FIN (DATOS3)

10.4 NOMBRE DEL PROCEDIMIENTO: SIMILAR3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: En caso de que no exista la medicina demandada por el cliente, despliega una lista de tres medicinas similares, para que el cliente decida cual de ellas le conviene. En caso de que solicite alguna de las presentadas, se verifican los datos tanto de clave como de cantidad, para ser procesadas por el procedimiento FACTMED3.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS. Ninguna.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
J	Valor que indica la clave del medicamento. Es el indice por medio del cual se busca el registro. Es de tipo entero.
UNISOLI	Valor de las unidades demandadas por el cliente. Es de tipo entero.
SIGUE	Valor para saber si el cliente acepta o no el medicamento similar. Es de tipo entero.

VARIABLES LOCALES

NOMBRE	DESCRIPCION
SALTEP	Valor que usa la estructura de control repetitiva en la condición para verificar la cantidad. Es de tipo entero.
SIMIL1	Clave de la medicina similar número 1. Es de tipo entero.
SIMIL2	Clave de la medicina similar número 2. Es de tipo entero.
SIMIL3	Clave de la medicina similar número 3. Es de tipo entero.
CONDIC	Valor que usa la estructura de control repetitiva en la condición para verificar el dato clave. Es de tipo entero.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
CHECA3	Realiza las lecturas al archivo VA y activa el procedimiento para validarlás.
DATOS3	Asigna los datos que existen en el registro leído a las variables que se usan en el proceso.

PSEUDOCODIGO

(* Inicia el Procedimiento SIMILAR3 *)

INICIA (SIMILAR3)

(* Inicializa variables *)

CONDIC <--- 0

SALTEP <--- 1

(* Se asignan las medicinas similares del registro actual *)

SIMIL1 <--- VA.MEDS1

SIMIL2 <--- VA.MEDS2

SIMIL3 <--- VA.MEDS3

(* Se listan las claves de las tres medicinas SIMILARES,
en orden descendente *)

1: 2: 3:

Se despliega el mensaje:

'QUIERE ALGUNA S)I O N)O: '

Lee SIGUE

Si SIGUE igual a 'S' entonces

Inicia (Si)

(* El cliente aceptó pedir alguna medicina similar *)

Repetir

Repetir

Leer CLAVE de la medicina similar por medio de J

Si J igual a SIMIL1 o a SIMIL2 o a SIMIL3 entonces

Inicia (Si)

Activa a CHECA3

Activa a DATOS3

CONDIC <--- 3

fin (si)

sino

Saca mensaje de error y pide nuevamente la clave del
medicamento similar seleccionado

fin (Si)

Hasta que CONDIC sea igual a 3

Si UNIDACT es menor o igual a cero y

UNIDEPE es menor o igual a cero entonces

(* La similar solicitada también está asotada *)

Inicia (si)

Despliega el mensaje:

'NO LA TENEMOS

QUIERE OTRA SIMILAR S)I O N)O '

Lee SIGUE

SI SIGUE es 'N' entonces

SALTEP <--- 9

fin (Si)

fin (si)

sino

Inicia (sino)

SALTEP <--- 9

SIGUE <--- 'S'

Presunta y lee cuantas unidades quiere
por medio de UNISOLI

fin (sino)

HASTA que SALTEP sea igual a 9

FIN (SIMILAR3)

10.5 NOMBRE DEL PROCEDIMIENTO: FACTMED3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Calcular la venta de cada producto y actualizar el registro correspondiente.

ARCHIVOS QUE USA EL MODULO.

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
UNISOLI	Valor de las unidades demandadas por el cliente. Es de tipo entero.
SIGUE	Valor para saber si el cliente acepta o no el medicamento similar. Es de tipo caracter.
SALTE	Valor que usa la estructura de control repetitiva en la condición para verificar la cantidad. Es de tipo entero.
CONTINUA	Valor que indica la ratificación de los datos, B o M, para que continúe el proceso. Es de tipo caracter.
J	Indice para acceder a los registros del Archivo de Trabajo. Es de tipo entero.
UNIDACT	Valor de las unidades en existencia actuales. Es de tipo entero.
PREMED	Valor del precio de la medicina. Es de tipo real.

IMPME Valor del porcentaje de impuesto. Es de tipo real.
UNIVEN Valor de las Unidades Vendidas Actuales. Es de tipo entero.
UNIDEPE Valor de las Unidades en Existencia del Pedido. Es de tipo entero.
UNVENPED Valor de las Unidades Vendidas del Pedido. Es de tipo entero.
SUBTPRO Total acumulado por factura. Es de tipo real.
PREXUN Valor de PRECMED X UNISOLI. Es de tipo real.
PRXUNXIM Valor de PREXUN X IMPMED. Es de tipo real.
TOTXMED Valor de PREXUN + PRXUNPIM. Es de tipo real.
SUB Valor del total de la venta por producto. Es de tipo real.
ACEPTA Valor que indica si la venta de la medicina o producto adicional es aceptada por el cliente. Es de tipo caracter.
UNIRES Resultado de hacer UNIDACT menos UNISOLI. Es de tipo entero.
TOTACUM Total acumulado, después de la aceptación o rechazo de la venta. Es de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CHECA3	Realiza lecturas directas al archivo de trabajo VA.
DATOS3	Asigna el contenido de los campos del registro a ciertas variables.
SEEKWRITE	Realiza escrituras en los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO

(* Inicia el procedimiento FACTMED3 *)

INICIA (FACTMED3)

SALTE <--- 0

Activa a CHECA3

Activa a DATOS3

Si UNIDACT es menor o igual a 0 y
UNIDEPE es menor o igual a cero entonces
Activa a SIMILAR3(SALTE)

fin (Si)

Si UNISOLI es mayor que UNIDACT + UNIDEPE y
UNIDACT + UNIDEPE es mayor que 0 entonces
(* se pide más de lo que tiene *)

Inicia (Si)

Despliega el mensaje:

"SOLO TENGO :^{*} Imprime (UNIDACT + UNIDEPE)

"SE LAS LLEVA S)I O N)O :^{*}"

Lee la respuesta por medio de SIGUE

Si SIGUE es igual 'S' entonces

UNISOLI <--- UNIDACT + UNIDEPE

fin (Si)

SALTE <--- 3

fin (Si)

Si SALTE es diferente de 0 entonces

Inicia (Si)

Despliega el mensaje:

"B)IEN O M)AL"

Lee la respuesta por medio de CONTINUA

fin (Si)

Si CONTINUA es igual a B y

SIGUE es igual a S entonces

Inicia (Si)

(* Se realizan los cálculos del medicamento *)

PREXUN <--- PRECMED*UNISOLI

PRXUNXIM <--- PREXUN*IMP MED

TOTXMED <--- PREXUN+PRXUNXIM

(* Precio unitario con impuesto *)

PRECMED <--- TOTXMED/UNISOLI

(* Se incrementa el total del cliente *)

SUBTPRO <--- SUBTPRO+TOTXMED

(* Se imprimen los resultados en una línea *)

<SUBTPRO> <J> <PRECMED> <UNISOLI> <TOTXMED> <I>

Se lee la confirmación de la venta del medicamento
por medio de ACEPTA

Si ACEPTA es igual a * entonces

Inicia (Si)

(* La venta fue aceptada *)

SUB <---TOTXMED

UNIRES <--- UNIDACT-UNISOLI

Si UNIRES es mayor o igual a 0 entonces

Inicia (Si)

(* Se actualizan campos *)

VA.UVEN <--- UNIVEN + UNISOLI

VA.UNID <--- UNIRES

fin (Si)

sino

Inicia (sino)

(* Cuando con esta venta se terminan las unidades
restantes *)

VA.UNID <--- 0

VA.UVEN <--- UNIVEN + UNIDACT

VA.UPE1 <--- UNIDEFE + UNIRES

VA.UVPD <--- UNVENPED - UNIRES

fin (sino)

Escribir en el registro J del Archivo VA

(* Valida la escritura *)

Activa a ERRCHK

Si ERROR es verdadero entonces

Activa a EXIT

fin (si)

sino

SUBTPRO <--- SUBTPRO - TOTXMED

fin (Si)

TOTACUM <--- TOTACUM + SUB

Se imprime el valor del total acumulado TOTACUM

FIN (FACTMED3)

10.6 NOMBRE DEL PROCEDIMIENTO: CEROS3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Inicializar las variables que se usan en la mayor parte de los procedimientos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
SIGUE	Variable por medio de la cual se reciben respuestas del cliente, tales como aceptación o rechazo, continuación, etc. Es de tipo caracter.
UNIDACT	Valor de las unidades en existencia actuales. Es de tipo entero.
PRECMED	Valor del precio de la medicina. Es de tipo real.
IMP MED	Valor del porcentaje de impuesto. Es de tipo real.
UNIVEN	Valor de las Unidades Vendidas Actuales. Es de tipo entero.
UNIDEPE	Valor de las Unidades en Existencia del Pedido. Es de tipo entero.
UNVENPED	Valor de las Unidades Vendidas del Pedido. Es de tipo entero.
PREXUN	Valor de PRECMED X UNISOLI. Es de tipo real.
PRXUNXIM	Valor de PREXUN X IMPMED. Es de tipo real.
TOTXMED	Valor de PREXUN + PRXUNPIM. Es de tipo real.

SUB Valor del total de la venta por producto. Es de tipo real.

UNIRES Resultado de hacer UNIDACT menos UNISOLI; Es de tipo entero.

VARIABLES LOCALES. Ninguna.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO

(* Inicia el procedimiento CEROS3 *)

INICIA (CEROS3)

SIGUE <--- / /

(* Inicializa variables con valor cero *)

TOTXMED <--- 0

SUB <--- 0

UNIDACT <--- 0

PRECMED <--- 0

IMPVED <--- 0

UNIVEN <--- 0

UNIDEPE <--- 0

UNVENPED <--- 0

UNIRES <--- 0

PREXUN <--- 0

PRXUNIM <--- 0

FIN (CEROS3)

10.7 NOMBRE DEL PROCEDIMIENTO: ADICION3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO Elaborar la sección de la factura de productos cuyos datos no existen en el Archivo de Trabajo. Realiza sus propios cálculos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CONTINUA	Valor que indica si continua con el proceso. Es de tipo caracter.
ACEPTA	Valor que indica si la venta de la medicina o producto adicional es aceptada por el cliente. Es de tipo caracter.
SUBTPRO	Valor acumulativo general de la factura. Es de tipo real.

VARIABLES LOCALES

NOMBRE	DESCRIPCION
UNIADIC	Valor de las unidades del producto adicional solicitado. Es de tipo entero.
PREADIC	Valor del precio del producto adicional. Es de tipo real.
IMPADIC	Valor del impuesto del producto adicional. Es de tipo real.
PRXUNAD	Valor de UNIADIC X PREADIC. Es de tipo real.
PXUXIAD	Valor de PRXUNAD X IMPADIC. Es de tipo real.
TOTXPAD	Valor de PRXUNAD + PXUXIAD. Es de tipo real.

PROCEDIMIENTOS QUE INVOCA: Ninguno

PSEUDOCODIGO.

(* Inicia el Procedimiento ADICION3 *)

INICIA (ADICION3)

TOADXFA <--- 0

Mientras CONTINUA sea igual a A

Inicia (Mientras)

(* Pide los datos del producto *)

(* Precio Unidades Impuesto *)

Despliega el mensaje:

' PREC UNID IMP *)

Lee por medio de PREADIC, UNIADIC y IMPADIC

Ratifica los datos por medio de CONTINUA

Si CONTINUA es igual a B entonces

Inicia (Si)

Si IMPADIC es mayor que 1 entonces

Inicia (Si)

IMPADIC <--- IMPADIC/100

fin (Si)

(* Realiza los cálculos *)

(* Multiplica Precio por unidades *)

PRXUNAD <--- PREADIC * UNIADIC

(* Multiplica PRXUNAD por el impuesto *)

PXUXIAD <--- PRXUNAD * IMPADIC

(* Se obtiene el total por producto adicional *)

TOTXPAD <--- PRXUNAD + PXUXIAD

(* Se suma al total acumulado por factura *)

SUBTPRO <--- SUBTPRO + TOTXPAD

(* Se imprimen los resultados en una línea *)

<SUBTPRO> EX <PREADIC> <UNIADIC> <TOTXPAD> <:)

(* Pide y lee la confirmación de la venta *)

Lee ACEPTA

Si ACEPTA es igual a * entonces

Inicia (si)

(* Venta aceptada *)

TOADXFA <--- TOADXFA + TOTXPAD

(* Escribe el contenido de SUBTPRO *)

fin (si)

sino

Inicia (sino)

Si SUBTPRO es mayor que cero entonces

SUBTPRO <--- SUBTPRO - TOTXPAD

fin (Si)

fin (sino)

Despliega el mensaje:

'A)DIC o T)OT ?'

Lee el valor de CONTINUA

fin (Mientras)

FIN (ADICION3)

10.8 NOMBRE DEL PROCEDIMIENTO: IMFINFAC

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Termina la factura, obteniendo el total de la venta, recibiendo el dinero que paga el cliente, calcula el cambio y despliega estos datos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES

NOMBRE	DESCRIPCION
TOTACUM	Total acumulado, después de la aceptación o rechazo de la venta. Es de tipo real.
CONTINUA	Valor que indica la ratificación de los datos, B o M, para que continúe el proceso. Es de tipo caracter.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
DINERO	Cantidad monetaria que paga el cliente. Es de tipo real.
CAMBIO	Resultado de DINERO menos TOTACUM. Es de tipo real.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO

(* Inicia el procedimiento IMFINFAC *)

INICIA (IMFINFAC)

Despliega el mensaje:

(valor de TOTACUM) "T O T A L \$" (valor de TOTACUM)

Repite

Solicita y lee DINERO

Ratifica el dato con CONTINUA

Si DINERO es mayor o igual a TOTACUM y

CONTINUA es igual a B entonces

Inicia (si)

CAMBIO <--- DINERO - TOTACUM

CONTINUA <--- 'K'

Despliega el mensaje:

(valor de DINERO) "DINERO \$" (valor de DINERO)

(Valor de TOTACUM) "TOTAL \$" (valor de TOTACUM)

(valor de CAMBIO) "CAMBIO \$" (valor de CAMBIO)

"GRACIAS POR SU COMPRA"

fin (si)

sino

Despliega mensaje de error y solicita el dato

fin (Si)

Hasta que CONTINUA sea igual a 'K'

FIN (IMFINFAC)

10.9 NOMBRE DEL PROCEDIMIENTO: CERRAR3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Guarda los datos más relevantes de la sesión de ventas en el registro cero; cierra el archivo VA.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
J	Valor del índice para acceder a los registros. Es de tipo entero.
AX	Año de la fecha del Archivo de Trabajo. Es de tipo entero.
MH	Mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DH	Día de la fecha del Archivo de Trabajo. Es de tipo entero.
NUMFAC	Valor del contador de facturas. Es de tipo entero.
LIMIT3	Clave máxima del medicamento contenido en VA. Es de tipo entero.
USO	Contiene el valor de uso del archivo. Este valor cambia cuando VA actualiza a BF, con el fin de ya no permitir a este VA volver a facturar. Es de tipo entero.

TOTADGRA Contiene las ventas totales de productos adicionales, dato que se graba en disco. Es de tipo real.

TOTADXSE Ventas totales de productos adicionales por sesión, esto es, que un archivo VA puede servir para varias sesiones. Es de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CHECA3	Realiza lecturas directas al archivo de trabajo VA.
SEEKWRITE	Realiza escrituras en los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO

(* Inicia el procedimiento CERRAR3 *)

INICIA (CERRAR3)

```
J <--- 0
Activa a CHECA3
(* Asigna valores *)
TOTADGRA <--- VA.PRECIO
TOTADXSE <--- TOTADXSE + TOTADGRA
VA.MEDS1 <--- AH
VA.MEDS2 <--- MH
VA.MEDS3 <--- DH
VA.PRECIO <--- TOTADXSE
VA.UNID <--- LIMIT3
VA.UVEN <--- NUMFAC
VA.UPE1 <--- 1
VA.UVPD <--- USO
Escribe en el registro J del Archivo VA
(* Valida la escritura *)
Activa a ERRCHK
  Si ERROR es verdadero entonces
    Activa EXIT
  fin (Si)
J <--- LIMIT3
Cierra VA
```

FIN (CERRAR3)

10.10 NOMBRE DEL PROCEDIMIENTO: FACTURA3

PROGRAMA DONDE SE LOCALIZA: MODU3

OBJETIVO: Inicializa las variables del proceso. Lista el encabezado de la factura. Activa a CEROS3. Lee los datos del medicamento, clave y cantidad, uno por uno. Valida los datos. Activa a FACTMED3. Continúa leyendo datos o elabora el final de la factura. Activa a ADICION si se desea continuar la facturación con productos adicionales. Activa a IMFINFAC para el total de la venta. Activa a CERRAR3.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
J	Valor del índice para acceder a los registros. Es de tipo entero.
AX	Año de la fecha del Archivo de Trabajo. Es de tipo entero.
MH	Mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DH	Día de la fecha del Archivo de Trabajo. Es de tipo entero.
NUMFAC	Valor del contador de facturas. Es de tipo entero.
LIMIT3	Clave máxima del medicamento contenido en VA. Es de tipo entero.

USO Contiene el valor de uso del archivo. Este valor cambia cuando VA actualiza a BF, con el fin de ya no permitir a este VA volver a facturar. Es de tipo entero.

TOTADGRA Contiene las ventas totales de productos adicionales, dato que se graba en disco. Es de tipo real.

TOTADXSE Ventas totales de productos adicionales por sesión, esto es, que un archivo VA puede servir para varias sesiones. Es de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
CEROS3	Inicializar las variables que se usan en la mayor parte de los procedimientos.
FACTMED3	Calcular la venta de cada producto y actualizar el registro correspondiente.
IMFINFAC	Termina la factura, obteniendo el total de la venta, recibiendo el dinero que paga el cliente, calcula el cambio y despliega estos datos.
CHECA3	Realiza lecturas directas al archivo de trabajo VA.
SEEKWRITE	Realiza escrituras en los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO

(* Inicia el Procedimiento FACTURA3 *)

INICIA (FACTURA3)

Despliega el mensaje:

'COLOGUE EL PAPEL AL INICIO DE UNA NUEVA HOJA
PRESIONE <RETURN>'

Lee SIGUE

TOTADXSE <--- 0

TOTADGRA <--- 0

Repite

SIGUE <--- ' '

TOTACUM <--- 0

SUBTPRO <--- 0

Despliega el encabezado de la factura

Repite

Activa a CEROS3

Pide datos de clave y cantidad

Lee J y UNISOLI

Pide ratificación de datos

Lee CONTINUA

Si CONTINUA es igual a B entonces

Inicia (Si)

(* Los datos fueron ratificados *)

Si J es mayor que 0 y

es menor o igual que LIMIT3 entonces

Activa a FACTMED3

sino

Despliega mensaje de error

fin (Si)

Solicita terminar o seguir por medio de SIGUE

Lee SIGUE

Hasta que SIGUE sea igual a T

TOTADXFA <--- 0

CONTINUA <--- 'C'

SOLICITA terminar o adicionales por medio de CONTINUA

Lee CONTINUA

Si CONTINUA es igual a 'A' entonces

Inicia (Si)

Activa a ADICION3(SUBTPRO,TOADXFA)

TOTACUM <--- TOTACUM + TOADXFA

TOADXSE <--- TOADXSE + TOADXFA

fin (Si)

Si TOTACUM es mayor que cero entonces

(* Si hubo ventas *)

Activa a IMFINFAC

fin (Si)

Pregunta si desea otra factura por medio de SIGUE

Lee SIGUE

Hasta que SIGUE sea igual a 'N'

Activa a CERRAR3

FIN (FACTURA3)

III.2.3.5 DOCUMENTACION DEL MODULO: CORTE

PROGRAMA DONDE SE LOCALIZA: MODU4

1. OBJETIVO: Elaborar el Reporte del Corte de Caja del Dia, para obtener un listado con los datos más significativos para el usuario.

2. FUNCIONES PRINCIPALES :
 - Asignar los Archivos de Trabajo y el de Nombres y Estadísticas del Medicamento al módulo.
 - Desplegar el encabezado del reporte, así como la identificación de la farmacia y la fecha del corte.
 - Realizar los cálculos necesarios para cada medicamento que tuvo ventas, con el fin de elaborar el corte de caja de ventas de una sesión de trabajo.
 - Incluir en el reporte los totales generados por ventas de artículos fuera del sistema, es decir, productos adicionales.
 - Cerrar archivos de información.
 - Desplegar notas con indicaciones para el usuario.

3. NARRATIVA DEL MODULO.

Este programa es el encargado de controlar a todos los procedimientos que interesan al módulo CORTE.

El módulo puede correrse en pantalla o impresora. Para tener un reporte concreto, se debe usar la impresora.

Activa al procedimiento ABRIR4, para que asigne los Archivos de Trabajo y el de Nombres y Estadísticas, reconocidos por el sistema por VA y NM, grabados en disco como TRABAJO.FIL Y NOMYEST.FIL, respectivamente.

En cuanto han sido asignados los dos archivos, se dejan listos para ser accedidos, ya sea para lectura o escritura. Del registro cero del archivo VA, llamado de alta seguridad, se toman la fecha del mismo y el límite máximo de registros que contienen información.

Posteriormente, se activa al procedimiento CABEZA4, el cual es el encargado de desplegar el encabezado de los reportes del corte de caja. Este encabezado se repite cada 40 registros.

A continuación, realiza un proceso iterativo, barriendo todo el archivo VA, desde 1 hasta su límite marcado por LIMIT4, por medio del procedimiento llamado CORTE4, el cual realiza los cálculos correspondientes para hacer el reporte. Únicamente se trabaja con aquellos medicamentos que hayan tenido movimientos en sus ventas. Se revisa el campo de unidades vendidas.

Una vez que se han hecho los cálculos correspondientes, y se ha barrido todo el archivo VA, entonces, se procede a obtener los totales generales, ya sea de los medicamentos contenidos en el sistema así como de los adicionales, por medio del procedimiento TOTALES4.

Finalmente, se despliegan unas notas para que el usuario esté informado de lo que debe hacer con las opciones del menú y con los archivos. Esto lo realiza el procedimiento NOTAS4.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL PROGRAMA	ARCHIVO DISCO	DESCRIPCION
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.
BLANCNM	Espacios que se dejan en blanco para manejar correctamente la impresión del nombre de la farmacia. Es de tipo entero.
NOMBREMP	Contiene el nombre de la farmacia. Es una cadena de 30 caracteres.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
JJ	Apuntador para acceder a los registros de los archivos VA y NM. Es de tipo entero.
ANOCORTE	Contiene el valor del año de la fecha del Archivo de Trabajo. Es de tipo entero.
MESCORTE	Contiene el valor del mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIACORTE	Contiene el valor del día de la fecha del Archivo de Trabajo. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO.

NOMBRE	OBJETIVO
CORTE	Es el programa manejador, por medio del cual se activan todos los procedimientos.
ABRIR4	Este procedimiento se encarga de asignar los archivos TRABAJO.FIL y NOMYEST.FIL, los cuales serán usados por varios procedimientos del módulo CORTE. En caso de que no existan, detecta el error y lo comunica al usuario. Una vez que los asigna, se posiciona en el primer registro de cada uno. Activa a CHECLEVA4. Lee al registro cero del archivo TRABAJO.FIL para saber cuál es la fecha del archivo sobre el cual se va a hacer el corte de caja. Resresa el dato de la fecha al procedimiento que lo activó.
CABEZA4	El procedimiento CABEZA4 tiene como objetivo imprimir los datos de identificación de la farmacia y la fecha del corte. Esta fecha está grabada en el registro cero del archivo TRABAJO.FIL, el cual fue leído por el procedimiento ABRIR4. A continuación escribe el nombre de cada columna que aparece en los reportes.
CORTE4	Realiza todos los cálculos necesarios para elaborar los reportes.
TOTALES4	Calcular los totales de las columnas.
NOTAS4	Proporciona una guía para indicar al usuario lo que debe hacer con los archivos VA y NM.
CHECLEVA4	Realizar las lecturas al Archivo VA.
CHECLENM4	Realizar las lecturas al Archivo NM.
ERRCHK	Verificar las lecturas a los archivos.
SEEKREAD	Realizar lecturas de acceso directo a los archivos.
SEEKWRITE	Realizar escrituras en los archivos.
EXIT	Dar por terminada la sesión.

DIAGRAMA JERARQUICO

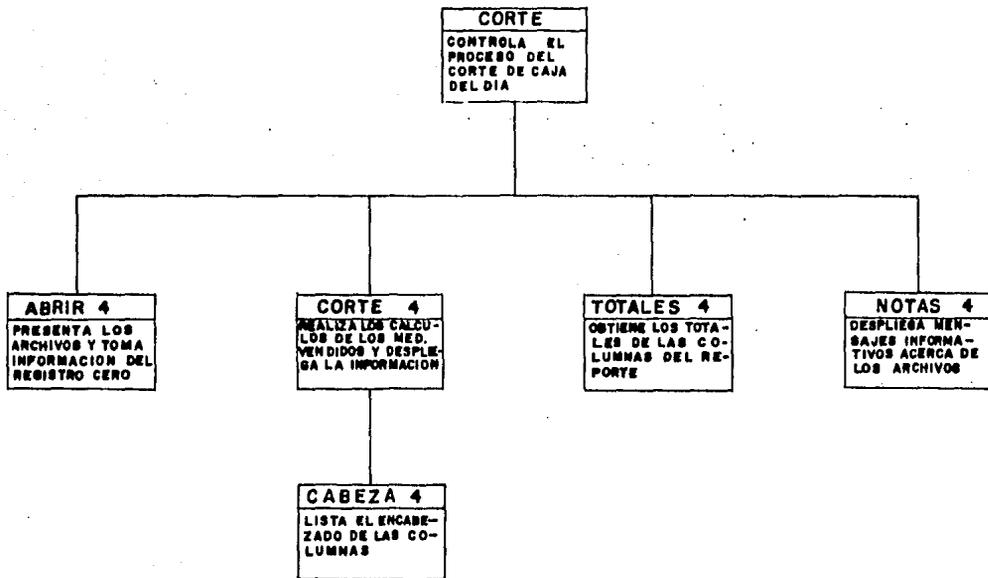
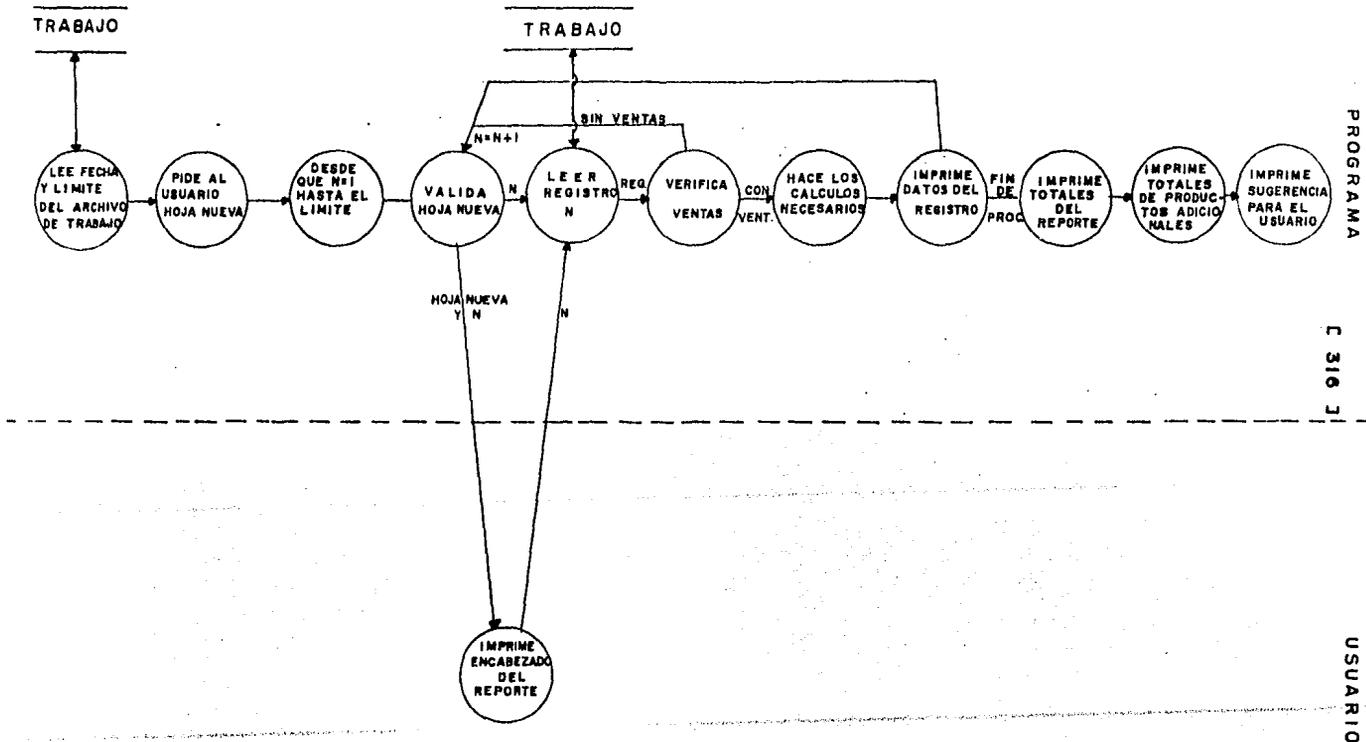


DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

[316]

USUARIO

9. PSEUDOCODIGO.

(* Inicia el Módulo CORTE *)

INICIA (CORTE)

Activa ABRIR4

Despliega el mensaje:

'COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA'

'PRESIONE <RETURN>'

Lee la variable SEGUIR

Activa CABEZA4

Activa CORTE4

Activa TOTALES4

Activa NOTAS4

FIN (CORTE)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO.

10.1 NOMBRE DEL PROCEDIMIENTO: ABRIR4.

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Asignar los dos archivos que usa el módulo y leer de ellos la fecha de trabajo y el número de la clave máxima del archivo VA.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADÍSTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
JJ	Apuntador para acceder a los dos archivos. Es de tipo entero.
ANDCORTE	Contiene el valor del año de la fecha del Archivo de Trabajo. Es de tipo entero.
MESCORTE	Contiene el valor del mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIACORTE	Contiene el valor del día de la fecha del Archivo de Trabajo.
LIMIT4	Valor de la máxima clave del Archivo de Trabajo. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CHECLEVA	Realizar las lecturas al Archivo VA.

PSEUDOCODIGO.

(* Inicia el procedimiento ABRIR4 *)

INICIA (ABRIR4)

Asigna al Archivo NM.
Se posiciona en el primer registro de NM.
Asigna al Archivo VA.
Se posiciona en el primer registro de VA.
Activa a CHECLEVA4.
Toma datos del registro cero.
(* Lee la fecha y la clave máxima del archivo VA, contenida
en el registro de alta seguridad *)

ANOCORTE <--- VA.MEDS1
MESCORTE <--- VA.MEDS2
DIACORTE <--- VA.MEDS3
LIMIT4 <--- VA.UNID

FIN (ABRIR4)

10.2 NOMBRE DEL PROCEDIMIENTO CABEZA4.

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Listar el encabezado del Reporte correspondiente al Corte de Caja.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO
NM NOMEYEST.FIL	ARCHIVO DE NOMBRES Y ESTADISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
BLANCNOM	Espacios que se dejan en blanco para manejar correctamente la impresión del nombre de la farmacia. Es de tipo entero.
NOMBREMP	Contiene el nombre de la farmacia. Es una cadena de 30 caracteres.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
ANOCORTE	Contiene el valor del año de la fecha del Archivo de Trabajo. Es de tipo entero.
MESCORTE	Contiene el valor del mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIACORTE	Contiene el valor del día de la fecha del Archivo de Trabajo. Es de tipo entero.

VARIABLES LOCALES: Ninsuna.

PROCEDIMIENTOS QUE INVOCA: Ninsuno.

PSEUDOCODIGO.

(* Inicia el procedimiento CABEZA4 *)

INICIA (CABEZA4)

(* Escribe identificación de FARMACIA Y FECHA *)

NOMEMP (identificación de la farmacia)

Usa BLANCNOM para centrar NOMBREMP

FECHA: ANOCORTE, MESCORTE, DIACORTE

(* Escribe el encabezado de las COLUMNAS *)

MEDICINA: incluye CLAVE y NOMBRE

UNIDADES VENDIDAS

PRECIO DE VENTA

SUBTOTAL \$

IVA \$

TOTAL \$

PRECIO DE COMPRA

PRECIO DE COMPRA X UNIDADES VENDIDAS

GANANCIA NETA \$

FIN (CABEZA4)

10.3 NOMBRE DEL PROCEDIMIENTO: CORTE4.

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Realizar todos los cálculos necesarios para elaborar los reportes.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
VA TRABAJO.FIL	ARCHIVO DE TRABAJO
NM NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTA DISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
BLANCNOM	Espacios que se dejan en blanco para manejar correctamente la impresión del nombre de la farmacia. Es de tipo entero.
NOMBREMP	Contiene el nombre de la farmacia. Es una cadena de 30 caracteres.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
JJ	Apuntador para acceder a los dos archivos. Es de tipo entero.
ANOCORTE	Contiene el valor del año de la fecha del Archivo de Trabajo. Es de tipo entero.
MESCORTE	Contiene el valor del mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIACORTE	Contiene el valor del día de la fecha del Archivo de Trabajo. Es de tipo entero.

SUMAVP	Es la suma total de las ventas parciales de los medicamentos. Es de tipo real.
SUMAPTOT	Es la suma total de la venta de todos los productos. Es de tipo real.
SUMAIMP	Contiene el valor global del impuesto a pagar de todas las ventas; este resultado aparece como la suma de los impuestos en la columna de IMPUESTO. Es de tipo real.
SUMAPXUN	Es la suma global de PRE1XUNI. Es de tipo real.
SUMAGANA	Obtiene el total de las ganancias por sesión de ventas. Es de tipo real.
VENPARC	Es el producto de las unidades vendidas totales (UNIVENT) por el precio.
IMPUESTO	Es el resultado de multiplicar el porcentaje de impuesto por el subtotal de cada medicina. Es de tipo real.
PRECTOT	Es la suma de la venta parcial (VENPARC) más su respectivo impuesto; se obtiene por artículo. Es de tipo real.
PRE1XUNI	Es el resultado de multiplicar el precio de compra por las unidades vendidas de cada medicamento. Es de tipo real.
GANANCIA	Es la diferencia entre PRECTOT menos PRE1XUNI; se obtiene la ganancia neta de cada producto. Es de tipo real.
UNIVENT	Es la suma de los contenidos de los campos UVEN + UVPD, es decir, indica el total de unidades vendidas actuales más las del pedido. Es de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
H	Variable de control para un proceso iterativo. Vale desde 1 hasta el número de la máxima clave del archivo. Es de tipo entero.
RENGLON	Es un contador de renglones, es decir de medicamentos que han sido listados. Sirve para controlar la impresión del encabezado del reporte. Es de tipo entero.
NOMBMED	Contiene el nombre de la medicina. Es una cadena de 30 caracteres.
BLANCOS	Cantidad de blancos que se dejan al finalizar el nombre del medicamento, para centrar a NOMBMED en el reporte. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Verificar las lecturas a los archivos.
CABEZA4	Listar el encabezado del Reporte correspondiente al Corte de Caja.
CHECLEVA4	Realizar las lecturas al Archivo VA.
CHECLENM4	Realizar las lecturas al Archivo NM.

PSEUDOCODIGO.

(* Inicia el procedimiento CORTE4 *)

INICIA (CORTE4)

(* Inicializa valores *)

SUMAVF <--- 0
 SUMAPTOT <--- 0
 SUMAIMF <--- 0
 SUMAPXUN <--- 0
 SUMAGANA <--- 0
 RENGLON <--- 1

```
(* Entra a un Proceso iterativo *)
Haz desde H = 1 hasta LIMITE4 lo siguiente
  Inicia (Haz)
  JJ <--- H
  Activa a CHECLEVA4
  Activa a CHECLENM4
  (* Verifica que se hayan los cálculos de sueltos
  registros que hayan tenido ventas *)
  Si VA.UVEN es mayor que cero o
  VA.UVPD es mayor que cero entonces
    Inicia (Si)
    VENPARC <--- 0
    (* Suma de los campos de unidades vendidas *)
    UNIVENT <--- VA.UVEN + VA.UVPD
    (* Calcula la venta parcial *)
    VENPARC <--- UNIVENT * VA.PRECIO
    (* Incrementa las ventas parciales *)
    SUMAVP <--- SUMAVP + VENPARC
    (* Calcula el impuesto *)
    IMPUESTO <--- VA.IMP * VENPARC
    (* Incrementa el impuesto acumulado *)
    SUMAIMP <--- SUMAIMP + IMPUESTO
    (* Calcula el precio total *)
    PRECOT <--- VENPARC + IMPUESTO
    (* Incrementa la suma de ventas totales *)
    SUMAPTOT <--- SUMAPTOT + PRECOT
    (* Calcula la venta del producto *)
    PREIXUNI <--- NM.PRE1 * UNIVENT
    (* Calcula la ganancia por producto *)
    GANANCIA <--- VENPARC - PREIXUNI
    (* Incrementa la ganancia total *)
    SUMAGANA <--- SUMAGANA + GANANCIA
    (* Calcula el número de espacios en blanco
    para centrar a NOMBMED *)
    NOMBMED <--- NM.NOMBRE
    BLANCOS <--- 30 - longitud de NOMBMED
    Despliega el mensaje!
    (* Anota todos los datos de la medicina
    en su columna correspondiente *)
    CLAVE, NOMBRE, UNIVENT, PRECIO, VENPARC,
    IMPUESTO, PRECOT, PRE1, PREXUNI, GANANCIA
    (* Incrementa el número del renglón *)
    RENGLON <--- RENGLON + 1
    Si RENGLON es igual a 40 entonces
      Inicia (Si)
      Activa a CABEZA4
      RENGLON <--- 1
      (* Reinicia el conteo de líneas *)
      fin (Si)
    fin (Si)
  fin (Haz)

FIN (CORTE4)
```

10.4 NOMBRE DEL PROCEDIMIENTO: TOTALES4.

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Calcula los totales de las columnas; guarda información importante en el registro de alta seguridad y cierra los archivos.

ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO
NM NONYEST.FIL	ARCHIVO DE NOMBRES Y ESTA DISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
JJ	Apuntador para acceder a los dos archivos. Es de tipo entero.
ANOCORTE	Contiene el valor del año de la fecha del Archivo de Trabajo. Es de tipo entero.
MESCORTE	Contiene el valor del mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIACORTE	Contiene el valor del día de la fecha del Archivo de Trabajo. Es de tipo entero.

SUMAVP Es la suma total de las ventas parciales de los medicamentos. Es de tipo real.

SUMAPTOT Es la suma total de la venta de todos los productos. Es de tipo real.

SUMAIMP Contiene el valor global del impuesto a pagar de todas las ventas; este resultado aparece como la suma de los impuestos en la columna de IMPUESTO. Es de tipo real.

SUMAPXUN Es la suma global de PREIXUNI. Es de tipo real.

SUMAGANA Obtiene el total de las ganancias por sesión de ventas. Es de tipo real.

FUE Venta total del día de artículos fuera del sistema o inventario. Es de tipo real.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Verificar las lecturas a los archivos.
CHECLEVA4	Realizar las lecturas al Archivo VA.
SEEKWRITE	Realizar escrituras en los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento TOTALES4 *)

INICIA (TOTALES4)

Despliega el mensaje:

'VENTA TOTAL DEL DIA *'

JJ <---0

Activa a CHECLEVA

(* Se posiciona en el registro cero*)

(* Asigna el contenido de los campos en las variables *)

FUE <--- VA.PRECIO

VA.MEDS1 <--- ANOCORTE

VA.MEDS2 <--- MESCORTE

VA.MEDS3 <--- DIACORTE

(* Asigna clave de uso del archivo *)

VA.UPE1 <--- 1

(* Escribe en VA el registro JJ *)

Escribe el registro JJ en el Archivo VA

(* Valida la escritura *)

Activa a ERRCHK

Si ERROR es verdadero entonces

Activa a EXIT

fin (Si)

Cierra los archivos

Despliega el mensaje:

'VENTA TOTAL DEL DIA DE ARTICULOS FUERA DE INVENTARIO:' (FUE)

FIN (TOTALES4)

10.5 NOMBRE DEL PROCEDIMIENTO: NOTAS4.

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Desplegar en una nota, las actividades que debe hacer el usuario para con sus archivos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO. Ninguno.

VARIABLES. Ninguna.

PROCEDIMIENTOS QUE INVOCA. Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento NOTAS4 *)

INICIA (NOTAS4)

Desplega el mensaje:

* N O T A : NO OLVIDE HACER LO SIGUIENTE:

1o. REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. USE LA OPCION <F> DEL MENU. RECUERDE QUE LA OPCION <F> SOLO PUEDE USARSE UNA VEZ CON CADA ARCHIVO DE TRABAJO. TENGA CUIDADO.

2o. PASE EL ARCHIVO DE TRABAJO AL DISCO ALMACENADOR*

FIN (NOTAS4).

10.6 NOMBRE DEL PROCEDIMIENTO: CHECLEVA4.

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Realizar las lecturas al Archivo VA.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO.

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
JJ	Apuntador para acceder a los los registros de los archivos VA y NM. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Verificar las lecturas de los archivos.
SEEKREAD	Realizar lecturas de acceso directo a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento CHECLEVA4 *)

INICIA (CHECLEVA4)

Leer el registro JJ en el Archivo VA

(* Valida la lectura *)

Activa a ERRCHK

Si ERROR es verdadero entonces

Activa a EXIT

fin (Si)

FIN (CHECLEVA4)

10.7 NOMBRE DEL PROCEDIMIENTO: CHECLENM4

PROGRAMA DONDE SE LOCALIZA: MODU4.

OBJETIVO: Realizar las lecturas al Archivo NM.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
NM NM	ARCHIVO DE NOMBRES Y ESTA DISTICAS DEL MEDICAMENTO
	NDMYEST.FIL

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
JJ	Apuntador para acceder a los dos archivos. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Verificar las lecturas de los archivos.
SEEKREAD	Realizar lecturas de acceso directo a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento CHECLENM4 *)

INICIA (CHECLENM4)

```

Leer el registro JJ del Archivo NM
(* Valida la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  Activa a EXIT
fin (Si)
    
```

FIN (CHECLENM4)

III.2.3.6 DOCUMENTACION DEL MODULO: REGRESAR

PROGRAMA DONDE SE LOCALIZA: MODUS

1. OBJETIVO: Actualizar al Archivo Maestro, enviando los nuevos datos que se generaron durante el día de ventas y que están en el Archivo de Trabajo.

2. FUNCIONES PRINCIPALES:

- Asignar a los archivos BF y VA al módulo para que pueda trabajar.
- Verificar que el archivo VA no haya sido usado anteriormente para actualizar al archivo BF. No está permitido que el archivo BF sea actualizado en más de una ocasión por el archivo VA.
- Transferir la información de un archivo a otro.
- Escribir en el registro de alta seguridad, específicamente en el campo de Unidades Vendidas del Pedido, del registro cero, un carácter que sirve como clave para evitar que se use el mismo archivo VA para actualizar al archivo BF en otra ocasión.
- Cerrar los dos archivos.

3. NARRATIVA DEL MODULO.

El módulo REGRESAR tiene como finalidad, la de pasar o transferir la información del archivo VA al archivo BF.

El archivo TRABAJO.FIL contiene todos los movimientos que se hicieron durante el día de trabajo. Así, los registros han cambiado su contenido en la mayoría de los campos debido a las ventas que se hicieron durante el día, las cuales alteraron a las Unidades Vendidas Actuales, las Unidades en Existencia Actuales, las Unidades Vendidas del Pedido, las Unidades en Existencia del Pedido y el total de las ventas de artículos fuera del sistema.

Por lo anterior, algunos campos del archivo VA tienen información actualizada, la cual debe pasar íntegra a los campos equivalentes del archivo BF. Por otra parte, existen otros campos del archivo TRABAJO.FIL que suman su información al campo correspondiente del archivo BF. No olvidar que el archivo TRABAJO.FIL es una copia parcial del archivo BF.

La actualización, transferencia o regreso de información del Archivo de Trabajo al Maestro se hace registro por registro.

El archivo VA tiene un dato clave, contenido en el campo UVPD del registro cero, el cual es alterado en el momento que se realiza la transferencia de información, por lo que si por error, se desea actualizar al Archivo Maestro con la misma copia más de una vez, el sistema no lo permite y marca error en el proceso.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO		DESCRIPCION
PROGRAMA	DISCO	
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
ERROR	Variable usada en las lecturas de los archivos. Si hay un error en la lectura, entonces adquiere un valor verdadero. Es de tipo booleana.
LIMMAST	Número de la clave máxima del registro que contiene información en BF. Es de tipo entero.

5.2 VARIABLES MODULARES.

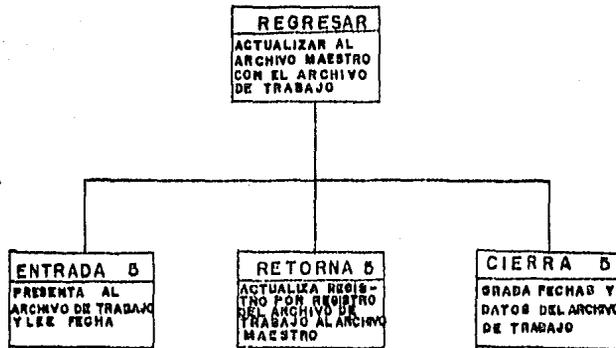
NOMBRE	DESCRIPCION
J	Apuntador para acceder a los Archivos Maestro y de Trabajo. Es de tipo entero.
ANO	Contiene el año de la fecha del Archivo de Trabajo. Es de tipo entero.
MES	Contiene el mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIA	Contiene el día de la fecha del Archivo de Trabajo. Es de tipo entero.

5.3 VARIABLES LOCALES. Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

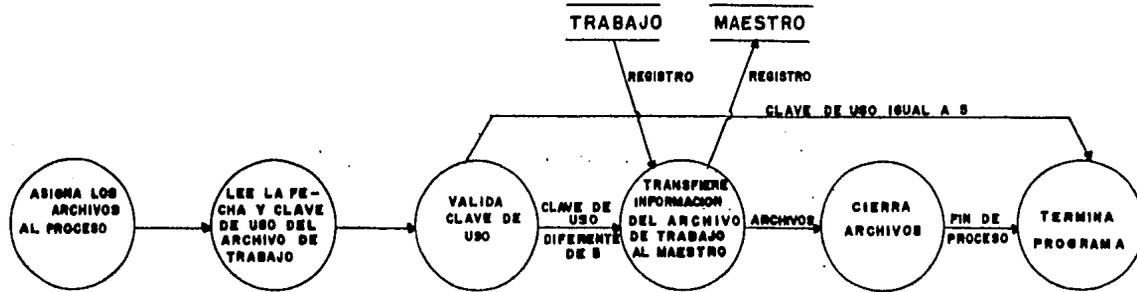
NOMBRE	OBJETIVO
ENTRADAS	Asigna los dos archivos (Maestro y de Trabajo) al proceso. Lee la fecha del archivo de Trabajo y la despliega en la pantalla. De esta manera el usuario tiene una referencia del Archivo de Trabajo que está usando. Resresa el control al módulo REGRESAR.
RETORNAS	Valida al Archivo de Trabajo. En caso que el Archivo de Trabajo sea el adecuado para resresar información, es decir que no haya sido usado anteriormente para tal fin, entonces transfiere la información al Archivo Maestro, a partir del registro i hasta el registro máximo según sea el valor de LIMMAST. Si no es el adecuado entonces marca error y lo comunica al usuario. En los dos casos cierra al Archivo Maestro. Resresa el control al módulo REGRESAR.
CIERRAS	Escribe en el registro de alta seguridad del Archivo de Trabajo la fecha y la clave de uso del mismo, validando esta acción. Cierra al Archivo de Trabajo.
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Dar por terminada la sesión.

DIAGRAMA JERARQUICO



E 336 1

DIAGRAMA DE FLUJO DE DATOS



PROGRAMA

E 337 J

USUARIO

9. PSEUDOCODIGO.

(* Inicia el módulo REGRESAR *)

INICIA (REGRESAR)

Activa a ENTRADAS
Activa a RETORNAS
Activa a CERRAR5

FIN (REGRESAR)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: ENTRADAS

PROGRAMA DONDE SE LOCALIZA: MODUS

OBJETIVO: Asignar los archivos Maestro y de Trabajo al proceso. Leer del Archivo de Trabajo la fecha para desplegarla en la pantalla.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
BF MAESTRO.FIL	ARCHIVO MAESTRO
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder a los Archivos Maestro y de Trabajo. Es de tipo entero.
ANO	Contiene el año de la fecha del Archivo de Trabajo. Es de tipo entero.
MES	Contiene el mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIA	Contiene el día de la fecha del Archivo de Trabajo. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el Procedimiento ENTRADAS *)

INICIA (ENTRADAS)

```

Asigna al Archivo de Trabajo VA
Asigna al Archivo Maestro BF
J <--- 0
Leer el registro J del Archivo VA
(* Verifica la lectura *)
Activa a ERRCHK
(* Reseta el valor de ERROR *)
Si ERROR es verdadera entonces
  (* Se da por terminada la sesión *)
  Activa a EXIT
fin (Si)
(* Asigna la fecha del Archivo de Trabajo a las variables *)
ANO <--- VA.MEDS1
MES <--- VA.MEDS2
DIA <--- VA.MEDS3
Despues el mensaje:
'EL ARCHIVO DE TRABAJO QUE ACTUALIZARA AL MAESTRO ES DE FECHA :
DIA: (Valor de DIA) MES: (Valor de MES) ANO: (Valor de ANO)'

```

FIN (ENTRADAS)

10.2 NOMBRE DEL PROCEDIMIENTO: RETORNAS

PROGRAMA DONDE SE LOCALIZA: MODUS

OBJETIVO: Verifica que el archivo de Trabajo sea el adecuado. Realiza un proceso iterativo para copiar desde el primer registro hasta el último marcado por la variable LIMMAST.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable usada en las lecturas de los archivos. Si hay un error en la lectura, entonces adquiere un valor verdadero. Es de tipo booleana.
LIMMAST	Número de la clave máxima del registro que contiene información en BF. Es de tipo entero.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder a los Archivos Maestro y de Trabajo. Es de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
INDICEJ	Indice para realizar la operación iterativa de copiar; rango: de 1 hasta el valor de LIMMAST. Es de tipo entero.

PROCEDIMIENTOS QUE INVOKA:

ERRCHK Valida lecturas y escrituras a los archivos.
SEEKREAD Realiza lecturas de acceso directo a los
 archivos.
SEEKWRITE Realiza escrituras en los archivos.
EXIT Dar Por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento RETORNAS *)

INICIA (RETORNAS)

(* Verifica que los Archivos Maestro y de Trabajo, tengan el mismo
número de registros, comparando el valor del campo UNID de
VA con la variable externa LIMMAST, para hacer el resreso *)
Si VA .UNID es igual a LIMMAST entonces
(* Inicia el proceso de resresar información *)
Inicia (SI)
(* Valida al archivo de Trabajo por medio de la información
contenida en el campo UVPD para saber si ya fue usado con
el mismo fin *)
SI UVPD es diferente de 5 entonces
(* Se cumplieron todas las condiciones, por lo tanto se
inicia el proceso *)
Inicia (SI)
J <--- 1
Despliega el mensaje:
"EL ARCHIVO DE TRABAJO ES EL CORRECTO "

"SE ESTA ACTUALIZANDO EL ARCHIVO MAESTRO"

"ESPERE LA SENAL PARA PODER CONTINUAR ..."
(* Inicia el proceso iterativo para copiar registro por
registro desde el registro 1 hasta el registro
correspondiente al valor de LIMMAST *)
HAZ desde que INDICEJ = 1 hasta LIMMAST lo siguiente
Inicia (Haz)
J <--- INDICEJ
(* Leer del archivo VA el registro J *)
Leer el registro J del Archivo VA
(* Verificar la lectura *)
Activa a ERRCHK

```
Si ERROR es verdadero entonces
  (* Error en la lectura, se termina la sesión *)
  Activa a EXIT
fin (Si)
(* Leer del archivo BF el registro J *)
Leer el registro J del Archivo BF
(* Verificar la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  (* Error en la lectura, se termina la sesión *)
  Activa a EXIT
fin (Si)

(* Realiza las siguientes actualizaciones campo por campo:
un punto y el nombre del campo *)
(* Asigna las unidades en existencia *)
BF.UNID <--- VA.UNID

(* Asigna la suma de las unidades vendidas actuales (BF)
más las unidades vendidas actuales (VA) a las unidades
vendidas actuales (BF) *)
BF.UVEN <--- BF.UVEN + VA.UVEN

(* Asigna el contenido de unidades en existencia del
pedido (VA) a las unidades en existencia del pedido
(BF) *)
BF.UPEI <--- VA.UPEI

(* Asigna al campo de unidades vendidas del pedido (BF)
la suma de las unidades vendidas del pedido (BF) más
las unidades vendidas del pedido (VA) *)
BF.UVPI <--- BF.UVPI + VA.UVPI

(* Escribir en el archivo BF el registro J *)
Escribir en el registro J del Archivo BF
(* Verifica la escritura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  (* Error en la escritura, se termina la sesión *)
  Activa a EXIT
fin (Si)
Fin (HAZ)

Cierra el archivo BF
Despliega el mensaje:
'EL SISTEMA TERMINO DE ACTUALIZAR AL MAESTRO'
fin (si)
sino
```

```

Inicia (sino)
  Cierra el archivo BF
  Despliega el mensaje:
  *ESTE ARCHIVO DE TRABAJO YA FUE USADO UNA VEZ*
  *NO ESTA PERMITIDO ACTUALIZAR DOS VECES*
  *CON EL MISMO ARCHIVO DE TRABAJO*
  fin (sino)
fin (Si)
sino
  Inicia (sino)
  Despliega el mensaje:
  *ESTA USANDO UN ARCHIVO DE TRABAJO EQUIVOCADO*
  *YA QUE NO TIENE EL MISMO NUMERO DE REGISTROS*
  *QUE EL ARCHIVO MAESTRO, VERIFIQUE LOS ARCHIVOS.*
  Fin (sino)

```

FIN (RETORNAS)

10.3 NOMBRE DEL PROCEDIMIENTO: CIERRAS

PROGRAMA DONDE SE LOCALIZA: MODUS

OBJETIVO: Escribir en el registro de alta seguridad del Archivo de Trabajo la fecha y la clave de uso del mismo, validando esta acción por medio de ERRCHK. Cerrar el Archivo de Trabajo.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
J	Apuntador para acceder a los Archivos Maestro y de Trabajo. Es de tipo entero.
ANO	Contiene el año de la fecha del Archivo de Trabajo. Es de tipo entero.
MES	Contiene el mes de la fecha del Archivo de Trabajo. Es de tipo entero.
DIA	Contiene el día de la fecha del Archivo de Trabajo. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza escrituras en los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el Procedimiento CERRAR5 *)

INICIA (CERRAR5)

J <--- 0

(* Guarda la fecha en los campos del registro de alta seguridad *)

VA.MEDS1 <--- ANO

VA.MEDS2 <--- MES

VA.MEDS3 <--- DIA

(* Asigna el valor de 5 al campo UVPD de VA para indicar que el
Archivo VA ya fue usado para actualizar a BF *)

VA.UVPD <--- 5

(* Escribe en el Archivo VA el registro J *)

Escribir en el registro J del Archivo VA

(* Verifica la escritura *)

Activa a ERRCHK

Si ERROR es verdadero entonces

(* Error en la escritura, se termina la sesión *)

Activa a EXIT

Fin (Si)

Cierra el Archivo VA

FIN (CERRAR5)

III.2.3.7 DOCUMENTACION DEL MODULO: TRANSFERIR

PROGRAMA DONDE SE LOCALIZA: MODU6

1. OBJETIVO: Transferir información del Archivo Maestro al Archivo de Nombres y Estadísticas y actualizar algunos campos del Archivo Maestro.

2. FUNCIONES PRINCIPALES:

- Validar la fecha en la que se pretenden hacer las actualizaciones.
- Tomando como base la fecha de transferencia, se obtiene el número de días transcurridos en el año, para poder calcular la estación correspondiente, ya sea primavera, verano, otoño o invierno.
- Verificar las existencias actuales de cada medicamento, ya que si este dato es igual a cero, se procederá a realizar la transferencia y actualización de información.
- De acuerdo con el número de días transcurridos, se vaciarán los datos del medicamento, para llevar un registro estadístico de ventas.
- Se actualizarán los datos correspondientes a UNID, UPED, UVP1, ANN, AA, MES, MM, DIA y DD de cada medicamento en BF.

3. NARRATIVA DEL MODULO.

Como primer punto, el módulo asigna y posiciona en apuntador en el primer registro de los Archivos Maestro y el de Nombres y Estadísticas. Inicializa la variable de control SIG. Activa a FECHAEX.

Una vez que la fecha ha sido aceptada como correcta entonces el sistema procede a hacer la transferencia de información de la siguiente manera:

Por medio de un mensaje, le indica al usuario que está transfiriendo información, por lo que se debe esperar a que termine para poder continuar.

Cómo hace la transferencia el sistema ?

Primero obtiene el total de días que han transcurrido del año en curso. Este dato lo obtiene tomando como base la fecha. El dato de días transcurridos DITO, le servirá para saber a que estación del año deberá enviar los datos estadísticos de ventas de cada medicina.

Sobre cuáles registros realiza la transferencia ?

Únicamente podrá transferirse información de los registros que están dentro del rango reconocido por el sistema, en este caso de 1 a LIMMAST. Posteriormente se verifica que el campo de la unidades en existencia actual sea igual a cero. Por qué ?. Porque únicamente la transferencia se hace sobre aquellos medicamentos que ya agotaron sus existencias actuales (UNID) de BF. Los demás no interesan, ya que las estadísticas son sobre las ventas reales no ficticias.

Bien, una vez que se verifica que el campo de UNID del registro es igual a cero entonces se procede a buscar a cuál temporada se cargan esas ventas.

Los rangos de cada estación o temporada son los siguientes:

Primavera (R1): De 81 a 172 días.

Verano (R2): De 173 a 264 días. Otoño (R3): De 265 a 355 días.

Invierno (R4): Menor que 81 o mayor que 355 días.

De esta manera, en cuanto encuentra el rango, estación o temporada a la que pertenecen las ventas, entonces procede a:

Sumar el contenido del rango a las unidades vendidas existentes y el resultado lo guarda en el mismo campo del rango.

Esta transferencia la hace tomando en cuenta dos archivos: el Maestro y el de Nombres y Estadísticas.

Ahora, procede a transferir datos dentro del mismo Archivo Maestro. El contenido del campo de Unidades Pedidas (UPED) lo pasa al campo de Unidades en Existencia Actuales (UNID) y hace UPED igual a cero. El contenido del campo de Unidades Vendidas del Pedido (UVP1) lo envía al campo de Unidades Vendidas Actuales (UVEN) y hace UVP1 igual a cero.

Transfiere las fechas de los medicamentos del pedido a las unidades existentes actuales, haciendo que las fechas del pedido sean iguales a cero.

Una vez que hace todo lo anterior, manda grabar cada registro en su respectivo archivo.

Todo esto lo hace únicamente con aquellos registros que cumplan con las dos condiciones iniciales.

Finalmente cierra los dos archivos, comunica al usuario que ya terminó y regresa el control al programa manejador.

4. ARCHIVOS QUE UTILIZA EL MODULO.

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTADÍSTICAS DEL MEDICAMENTO

5. VARIABLES.

5.1 VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima del registro que contiene información en el Sistema. Es de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.
MESEX	Contiene el mes de la fecha en la que se está realizando la transferencia de información. Es de tipo entero.
DIAEX	Contiene el día de la fecha en la que se está realizando la transferencia de información. Es de tipo entero.
VER	Variable por medio de la cual se valida la fecha; 1 es verdadero, 0 es falso. Es de tipo entero.

5.2 VARIABLES MODULARES.

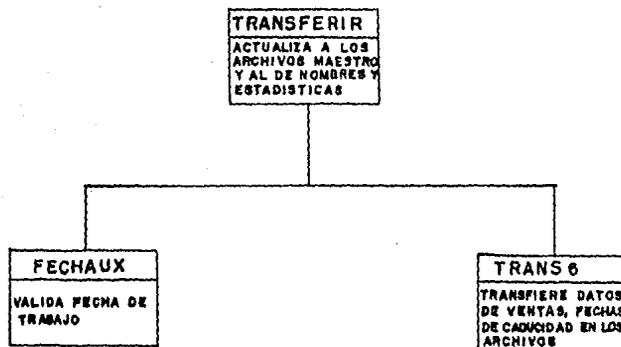
NOMBRE	DESCRIPCION
DITO	Número de días transcurridos del año. Se obtiene haciendo ((MES-1) por 30) menos 8. Es de tipo entero.
I	Apuntador para acceder a los registros de los archivos. Es de tipo entero.
SIG	Opción para salir de este procedimiento. Es de tipo carácter.

5.3 VARIABLES LOCALES. Ninguna.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

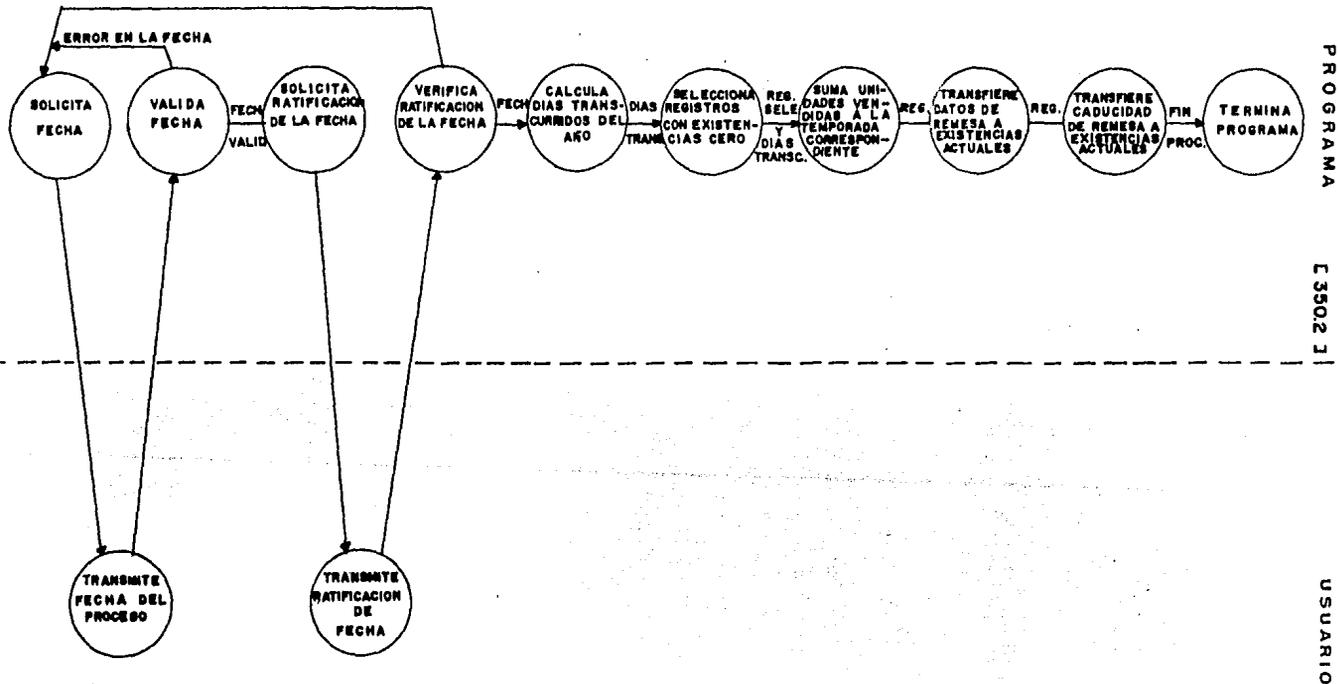
NOMBRE	OBJETIVO
FECHAEX	Validar la fecha, evaluando el año, mes y día. Si la fecha es correcta, asigna a VER el valor 1; Por el contrario, si la fecha tiene una error, entonces VER tomará el valor 0.
TRANS1	Hacer la transferencia de información, únicamente sobre aquellos registros que cumplan con ciertas condiciones; la transferencia se puede hacer dentro del mismo registro, o a otro registro de archivo diferente.
ERRCHK	Valida las lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Escribir datos en el registro de un archivo.
EXIT	Dar por terminada la sesión.

DIAGRAMA JERARQUICO



C 3501 2

DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO.

(* Inicia el Módulo TRANSFERIR *)

INICIA (TRANSFERIR)

Asigna el Archivo BF y coloca el apuntador en el Primer registro.
Asigna el Archivo NM y coloca el apuntador en el Primer registro.
SIG <--- ' '

Repite

Activa FECHAEX

(* Si la fecha está correcta entra al proceso de transferir *)

Pide la ratificación de la fecha por medio de SIG

Hasta que SIG sea igual a 'S'

Si VER es igual a 1 entonces

Activa a TRANSI

fin (Si)

(* Asigna el valor de la máxima clave a la variable I *)

I <--- LIMMAST

Cierra los archivos NM y BF

Despliega el mensaje:

'EL SISTEMA TERMINO.

PUEDE CONTINUAR'

FIN (TRANSFERIR)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO.

10.1 NOMBRE DEL PROCEDIMIENTO: TRANS6

PROGRAMA DONDE SE LOCALIZA: MODU6.

OBJETIVO: Hacer la transferencia de información, únicamente sobre aquellos registros que cumplan con ciertas condiciones; la transferencia se puede hacer dentro del mismo registro, o a otro registro de archivo diferente.

ARCHIVOS QUE UTILIZA EL MODULO.

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
BF	MAESTRO.FIL	ARCHIVO MAESTRO
NM	NOMYEST.FIL	ARCHIVO DE NOMBRES Y ESTA DISTICAS DEL MEDICAMENTO

VARIABLES.

VARIABLES EXTERNAS.

NOMBRE	DESCRIPCION
LIMMAST	Número de la clave máxima del registro que contiene información en el Sistema. Es de tipo entero.
ERROR	Es la variable por medio de la cual se verifican las lecturas y escrituras a los archivos. Es de tipo booleano.
MESEX	Contiene el mes de la fecha en la que se está realizando la transferencia de información. Es de tipo entero.
DIAEX	Contiene el día de la fecha en la que se está realizando la transferencia de información. Es de tipo entero.

VARIABLES MODULARES.

NOMBRE	DESCRIPCION
DITO	Número de días transcurridos del año. Se obtiene haciendo ((MES-1) por 30)) menos 8. Es de tipo entero.
I	Apuntador para acceder a los registros de los archivos. Es de tipo entero.

VARIABLES LOCALES.

NOMBRE	DESCRIPCION
IT	Variable iterativa; rango de 1 a LIMMAST. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
ERRCHK	Valida las lecturas y escrituras a los archivos.
SEEKREAD	Realiza lecturas de acceso directo a los archivos.
SEEKWRITE	Escribir datos en el registro de un archivo.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento TRANS6 *)

INICIA (TRANS6)

```
Despliega el mensaje!
"EL SISTEMA ESTA TRANSFIRIENDO INFORMACION
 ESPERE LA SENAL PARA PODER CONTINUAR ..."
MESEX <-- MESEX - 1
(* Obtiene el total de días transcurridos en el año *)
DITO <-- (MESEX*30) + DIAEX
DITO <-- DITO - 8
Haz desde IT isual a 1 hasta LIMMAST lo siguiente
Inicia (Haz)
I <--- IT
Lee el registro I del Archivo BF
(* Valida la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
  Activa a EXIT
fin (si)
```

```

Si el contenido del campo UNID de BF es igual a cero entonces
  Inicia (si)
  Lee el registro I del Archivo NM
  (* Valida la lectura *)
  Activa a ERRCHK
  Si ERROR es verdadero entonces
    Activa a EXIT
  fin (Si)
  Si DITO es >= a 81 y es <= a 172 entonces
    NM.R1 <--- NM.R1 + BF.UVEN
  fin (Si)
  Si DITO es >= a 173 y es <= a 264 entonces
    NM.R2 <--- NM.R2 + BF.UVEN
  fin (Si)
  Si DITO es >= a 265 y es <= a 355 entonces
    NM.R3 <--- NM.R3 + BF.UVEN
  fin (Si)
  Si DITO es < a 81 y es >= a 356 entonces
    NM.R4 <--- NM.R4 + BF.UVEN
  fin(Si)
  Escribe en el registro I del Archivo NM
  (* Valida la escritura *)
  Activa a ERRCHK
  Si ERROR es verdadero entonces
    Activa a EXIT
  fin (Si)
  (* Realiza las siguientes asignaciones, transfiriendo
  información de un campo a otro del mismo registro
  y además inicializa el campo fuente *)
  (* Unidades pedidas a unidades en existencia *)
  BF.UNID <--- BF.UPEB
  BF.UPEB <--- 0

  (* Unidades vendidas del pedido a
  unidades vendidas actuales *)
  BF.UVEN <--- BF.UVP1
  BF.UVP1 <--- 0
  (* Fecha de caducidad de las unidades pedidas a
  las unidades en existencia actuales *)
  BF.ANN <--- BF.AA
  BF.AA <--- 0
  BF.MES <--- BF.MM
  BF.MM <--- 0
  BF.DIA <--- BF.DD
  BF.DD <--- 0
  Escribir en el registro I del Archivo BF
  (* Valida la escritura *)
  Activa a ERRCHK
  Si ERROR es verdadero entonces
    Activa a EXIT
  fin (Si)
fin (si)
fin (Haz)
FIN (TRANS6)

```

III.2.3.8 DOCUMENTACION DEL MODULO: ACTUALI7

PROGRAMA DONDE SE LOCALIZA: MODU7

1. **OBJETIVO:** Hacer correcciones al Archivo de Trabajo, en caso de que se hayan presentado problemas al momento de hacer las facturas.

2. **FUNCIONES PRINCIPALES:**
 - Asignar el archivo de trabajo.
 - Presentar los datos de identificación del archivo.
 - Validar las claves personales para permitir el acceso a este módulo.
 - Validar la existencia del registro que se desea actualizar.
 - Desplegar el listado de los campos susceptibles a ser modificados.
 - Desplegar el dato viejo.
 - Solicitar el dato nuevo.
 - Ratificar las actualizaciones.
 - Hacer las correcciones en el archivo VA.
 - Cerrar el archivo de trabajo.

3. NARRATIVA DEL MODULO.

Para entrar el módulo ACTUALI7, se elige la opción <E> del menú principal de funciones.

El sistema presenta el archivo. Accesa la información que existe en el registro cero llamado de alta seguridad, para tomar la fecha y las claves de acceso al sistema o de seguridad.

Despliega el dato de la fecha y si el usuario acepta al archivo como útil para sus intereses entonces continua trabajando; en caso contrario, se sale del módulo y se regresa el control al programa manejador principal.

Si continuamos con el módulo, entonces se solicitan las dos claves o contraseñas de seguridad, las cuales deben coincidir con las que tiene almacenadas el sistema para permitir continuar. En caso contrario se rechaza la solicitud del usuario y se regresa el control al programa manejador.

Si las contraseñas coinciden con las claves del sistema, entonces se continua en el módulo.

Una vez dentro, es decir, que el usuario fue aceptado, se procede a presuntar el número del registro que se desea modificar. Este dato debe estar dentro del rango permitido, es decir, debe ser mayor de 0 y menor o igual que LIMIT7. Si no se cumple con este requisito, el sistema marca el error. Por otra parte si el registro se encuentra en el sistema entonces, procede a buscarlo de manera directa, para poder acceder a su información. El acceso es directo.

Si el registro es localizado, se lista la identificación (número y nombre) de los campos susceptibles de ser modificada su información.

A continuación se pide el número del campo a ser modificado. El sistema verifica que este dato se encuentre dentro del rango permitido (mayor que 0 y menor que 6). Si hay error lo detecta y denuncia.

Finalmente, si no se presentan problemas, se hace la corrección en el campo del registro seleccionado.

4. ARCHIVOS QUE UTILIZA EL MODULO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

5. VARIABLES.

5.1 VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable para verificar las lecturas y escrituras al archivo. Es de tipo booleano.

5.2 VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CLASECP	Segunda clave de seguridad; se pasa como parámetro al procedimiento ACCESO. Es de tipo entero.
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.
X	Contiene el número del campo que se desea modificar. Es de tipo entero.
UNI	Valor contenido en algunos campos susceptibles a ser modificados. Es de tipo entero.
AXMXD	Es el resultado de multiplicar AND por MES por DIA. Es de tipo entero.
ACEP	Variable por medio de la cual se comunica la aceptación o el rechazo del usuario que desea hacer modificaciones; es aceptado si toma el valor de 1, en caso contrario se rechaza la solicitud del usuario. Es de tipo entero.
LIMIT7	Contiene el valor del máximo registro útil del archivo de Trabajo. Es de tipo entero.
CLAPRIP	Primera clave de seguridad; se pasa como parámetro al procedimiento ACCESO. Es de tipo real.

OTRO Variable por medio de la cual el usuario tiene comunicación con el sistema para responder a presuntas del mismo. Puede valer S o N u otro caracter reconocido. Es de tipo caracter.

5.3 VARIABLES LOCALES: No tiene.

6. PROCEDIMIENTOS Y FUNCIONES QUE UTILIZA EL MODULO:

NOMBRE	OBJETIVO
CABEZA7	Desplegar la identificación (número y nombre) de los campos que son susceptibles de ser modificados.
CHECA7	Realiza las lecturas al archivo de trabajo, las verifica por medio del procedimiento ERRCHK.
ASIGNAR7	Asigna el archivo de Trabajo; activa a CHECA7; lee información confidencial del registro de alta seguridad (registro cero).
ACCESO7	Solicita al usuario, una a una, las claves de seguridad; verifica que coincidan con los datos almacenados en el registro de alta seguridad. En caso afirmativo asigna a ACEPC el valor de 1, lo que significa que ha sido aceptado el usuario para hacer actualizaciones.
MODIFI7	Presenta la información contenida en el campo a ser modificado, identificada como "dato viejo" y solicita el dato nuevo para hacer la actualización. Exige la ratificación del nuevo dato.
MODIFI7A	Función similar a MODIFI7, pero exclusivo para el registro cero.
SIMODIF7	Actualiza "físicamente" la información en el campo del registro determinado por el usuario.
CUALREG7	Solicita el número del registro a ser modificado; verifica y acepta o rechaza el dato.
CUALCAM7	Pide el número del campo a ser modificado; verifica y acepta o rechaza los datos.
SEEKREAD	Realiza las lecturas de acceso directo a los archivos.
SEEKWRITE	Realiza las escrituras a los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada una sesión.

DIAGRAMA JERARQUICO

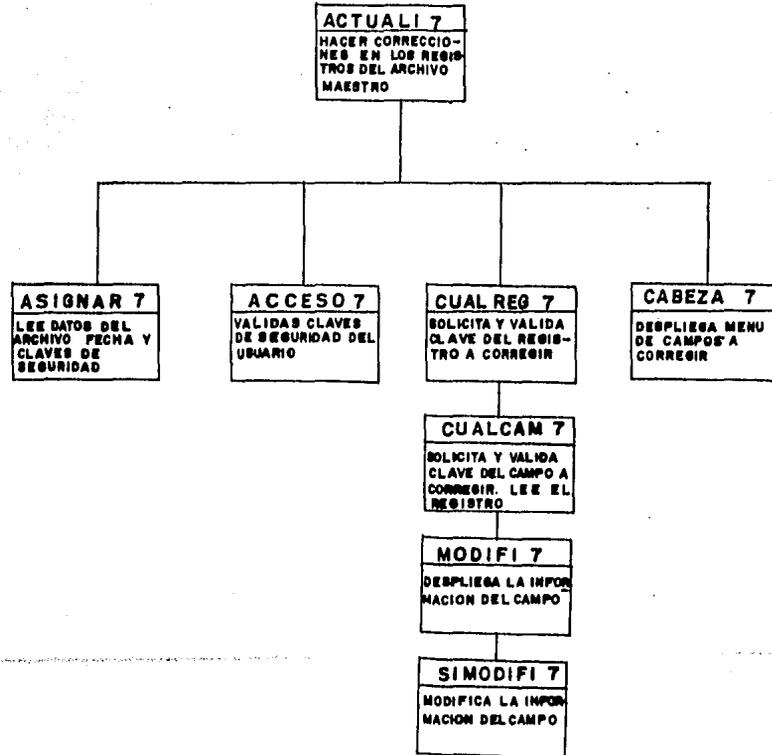


DIAGRAMA DE FLUJO DE DATOS

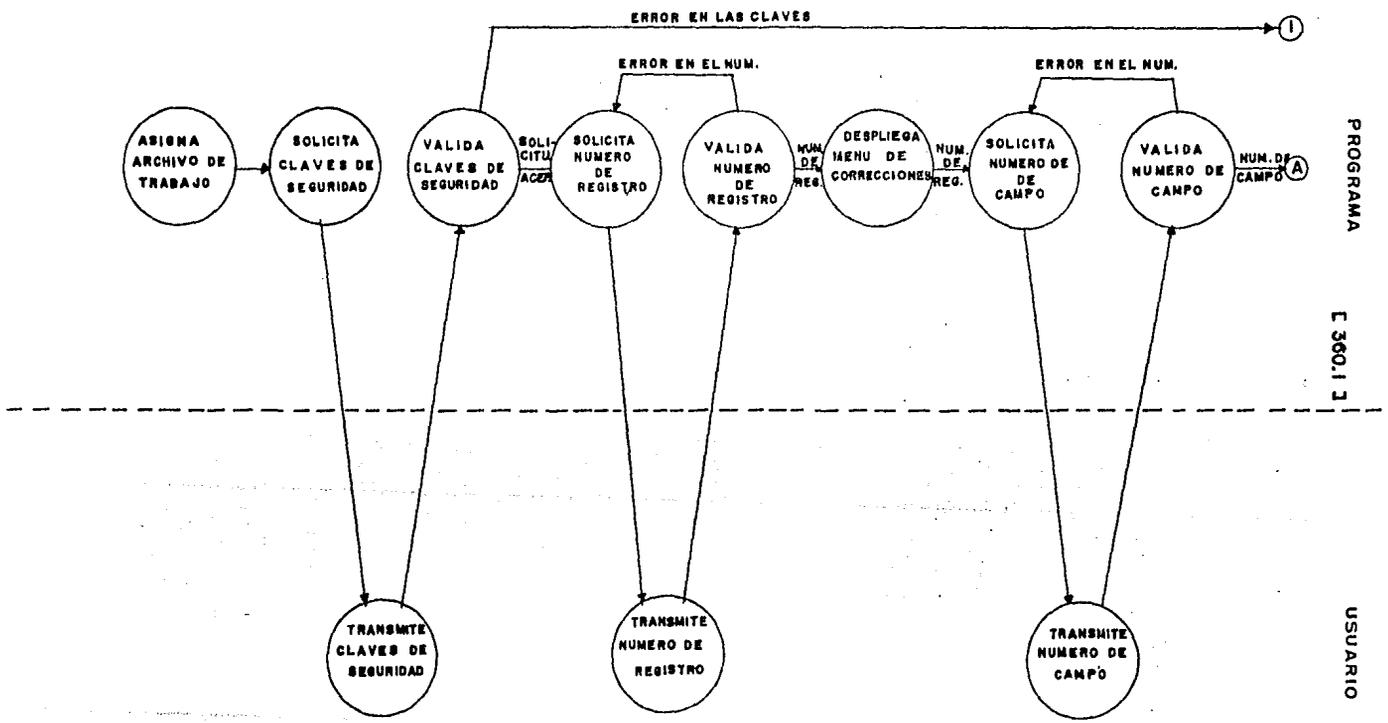
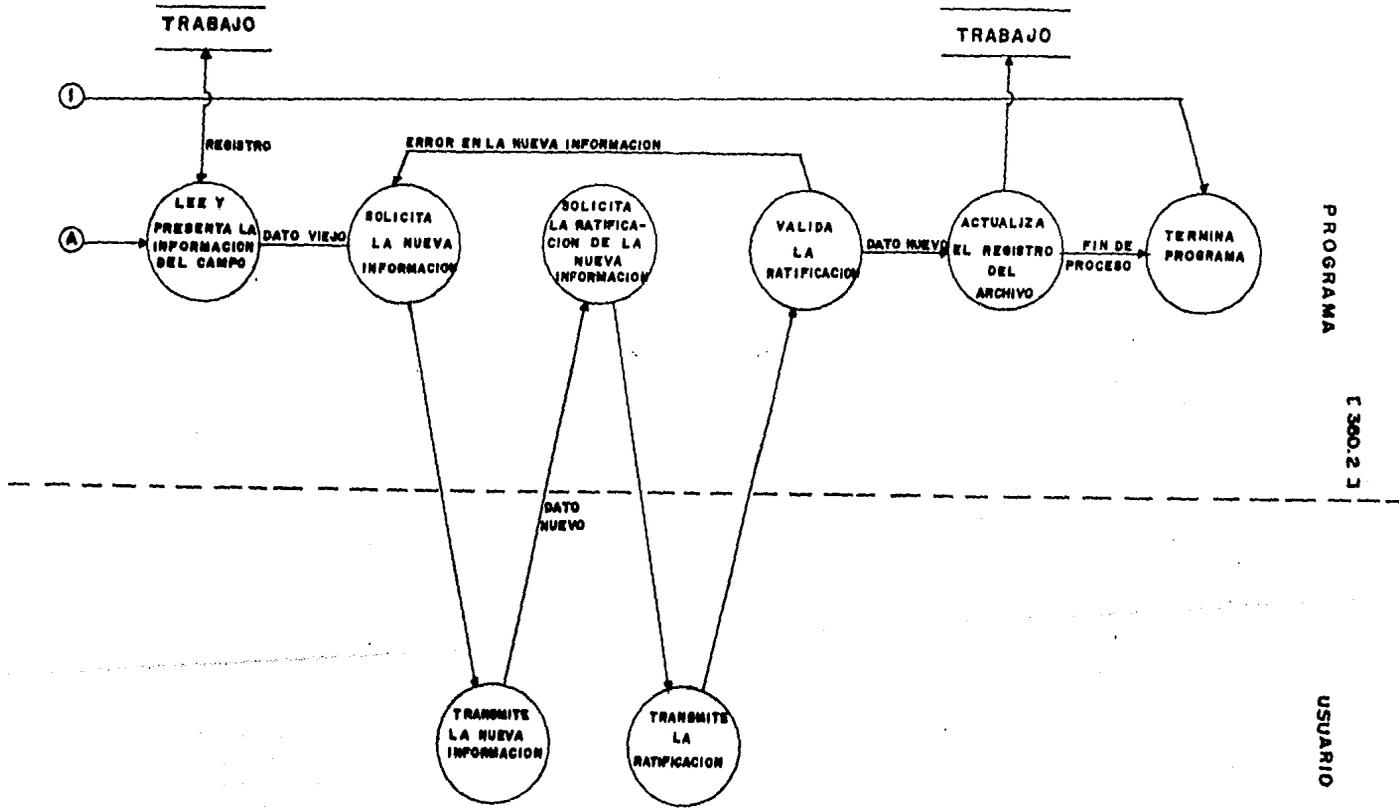


DIAGRAMA DE FLUJO DE DATOS



9. PSEUDOCODIGO.

(* Inicia el Módulo ACTUALI7 *)

INICIA (ACTUALI7)

Activa ASIGNAR (AXMXD, CLASECP, CLAPRIP)
Despliega el mensaje:

"ESTE ES EL ARCHIVO DE TRABAJO QUE USTED BUSCA ?"
"ANOTE S)I o N)O => "

Lee la respuesta (OTRO)

Si OTRO es 'S' o 's' entonces

Activa a ACCESO7 (CLASECP, ACEP, CLAPRIP)

Si ACEP es igual al 1 entonces

Activa ACEPTADO

sino

Despliega el mensaje:

"USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES"

fin (Si)

fin (Si)

I <--- LIMIT7

Cierra el archivo VA

FIN (ACTUALI7)

10. DESCRIPCION DE PROCEDIMIENTOS DEL MODULO:

10.1 NOMBRE DEL PROCEDIMIENTO: CABEZA7

PROGRAMA DONDE SE LOCALIZA: MODU7

OBJETIVO: Desplegar la identificación (número y nombre) de los campos que son susceptibles de ser modificados.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES: Ninguna.

VARIABLES LOCALES: Ninguna.

PSEUDOCODIGO:

(* Inicia el procedimiento CABEZA7 *)

INICIA (CABEZA7)

Despliega el siguiente mensaje:

"LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES :

C	A	M	P	O		
é	N	O	M	B	R	E

=====

- 1 UNIDADES EN EXISTENCIA ACTUALES.
 - 2 UNIDADES VENDIDAS ACTUALES.
 - 3 UNIDADES PEDIDAS.
 - 4 UNIDADES VENDIDAS DEL PEDIDO.
 - 5 NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)
 - 6 VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)
- =====

FIN (CABEZA7)

10.2 NOMBRE DEL PROCEDIMIENTO: CHECA7

PROGRAMA DONDE SE LOCALIZA: MODU7

OBJETIVO: Realiza las lecturas al archivo de trabajo; las verifica por medio del Procedimiento ERRCHK.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS:

NOMBRE	DESCRIPCION
ERROR	Variable para verificar las lecturas y escrituras al archivo. Es de tipo booleano.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
SEEKREAD	Realiza las lecturas de acceso directo a los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento CHECA7 *)

INICIA (CHECA7)

Leer el registro I del Archivo VA
(* Valida la lectura *)
Activa a ERRCHK
Si ERROR es verdadero entonces
Activa a EXIT
fin (Si)

FIN (CHECA7)

10.3 NOMBRE DEL PROCEDIMIENTO: ASIGNAR7.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Asigna el archivo de Trabajo; activa a CHECA7; lee información confidencial del registro de alta seguridad (registro cero).

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA DISCO	DESCRIPCION
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CLASECA	Segunda clave de seguridad, leida del registro de alta seguridad; se resresa el valor como parámetro (CLASECP) al procedimiento que lo activo.. Es de tipo entero.
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.
AXMXD	Es el resultado de multiplicar AND por MES por DIA. Parámetro de tipo entero.
LIMIT7	Contiene el valor del máximo registro útil del archivo de Trabajo; valor leido del registro cero. Es de tipo entero.
CLAPRIA	Primera clave de seguridad; leida del registro de alta seguridad; el valor se resresa como parámetro (CLAPRIP) al procedimiento que lo activo. Es de tipo real.

10.4 NOMBRE DEL PROCEDIMIENTO: ACCES07.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Solicita al usuario, una a una, las claves de seguridad; verifica que coincidan con los datos almacenados en el registro de alta seguridad. En caso afirmativo asigna a ACEPC el valor de 1, lo que significa que ha sido aceptado el usuario para hacer actualizaciones.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
CLAPRIC	Valor de la primera clave de seguridad; Parámetro por medio del cual se pasa a este procedimiento el valor de la primera clave (CLAPRIP), contenida en el registro de alta seguridad. Parámetro de tipo real.
CLASECC	Valor de la segunda clave de seguridad; Parámetro por medio del cual se pasa a este procedimiento el valor de la segunda clave (CLASECP), contenida en el registro de alta seguridad. Parámetro de tipo entero.
ACEPC	Si las claves proporcionadas por el usuario, coinciden con las claves guardadas en el registro de alta seguridad, toma el valor de 1; en caso contrario asume un valor diferente, es un parámetro (ACEP). Parámetro de tipo entero.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
CLAPRIM	Se le asigna el valor leído correspondiente a la primera clave de seguridad, proporcionada por el usuario. Es de tipo entero.
CLASECU	Se le asigna el valor leído correspondiente a la segunda clave de seguridad, proporcionada por el usuario. Es de tipo entero.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento ACCES07 *)

INICIA (ACCES07)

```
ACEPC <--- 2
Despliega el mensaje:
"ANOTE LA CLAVE PRIMARIA DE SEGURIDAD => "
Lee el valor de CLAPRIM
Despliega el mensaje:
"ANOTE LA CLAVE SECUNDARIA DE SEGURIDAD => "
Lee el valor de CLASECU
SI CLAPRIM es igual a CLAPRIC y
CLASECU es igual a CLASECC entonces
(* El usuario es acertado *)
ACEPC <--- 1
fin (Si)
```

FIN (ACCES07)

10.5 NOMBRE DEL PROCEDIMIENTO: MODIFI7.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Presenta la información contenida en el campo a ser modificado, identificada como "dato viejo" y solicita el dato nuevo para hacer la actualización. Exige la ratificación del nuevo dato.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.
X	Contiene el número del campo que se desea modificar. Es de tipo entero.
UNI	Valor contenido en algunos campos susceptible de ser modificados. Es de tipo entero.
OTRO	Variable por medio de la cual el usuario tiene comunicación con el sistema para responder a preguntas del mismo. Puede valer S o N u otro caracter reconocido. Es de tipo caracter.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
MODIFI7A	Función similar a MODIFI7, pero exclusivo para el resisro cero.
SIMODIF7	Actualiza "físicamente" la información en el campo del registro determinado por el usuario.

PSEUDOCODIGO.

(* Inicia el procedimiento MODIFI7 *)

INICIA (MODIFI7)

OTRO <--- 'N'

Si X es mayor que 0 entonces

Inicia (1)

En caso que X

sea igual a 1 Inicia (1)

Despliega el mensaje:

'DATO VIEJO : '

(valor del campo UNID) "ANOTE EL DATO NUEVO : '

Lee el valor de UNI

VA.UNID <--- UNI

fin (1)

sea igual a 2 Inicia (2)

Despliega el mensaje:

(valor del campo UVEN) "ANOTE EL DATO NUEVO : '

Lee el valor de UNI

VA.UVEN <--- UNI

fin (2)

sea igual a 3 Inicia (3)

Despliega el mensaje:

(valor del campo UPE1) "ANOTE EL DATO NUEVO : '

Lee el valor de UNI

VA.UPE1 <--- UNI

fin (3)

sea igual a 4 Inicia (4)

Despliega el mensaje:

(valor del campo UVPD) "ANOTE EL DATO NUEVO : '

Lee el valor de UNI

VA.UVPD <--- UNI

fin (4)

sino

Inicia (sino)

Mensaje:

'UNICAMENTE PUEDE CORREGIR LOS CAMPOS DEL 1 AL 4'

OTRO <--- 'C'

fin (sino)

sino

(* Para modificar el registro cero *)

Activa a MODIFI7A

fin (Si)

Si OTRO es diferente de 'C' entonces

Inicia (Si)

Despliega el mensaje:

'ESTA CORRECTO ? ANOTE S)I o N)O : '

Lee el valor de OTRO

Si OTRO es igual a 'S' o 's' entonces

(* Las modificaciones fueron ratificadas por el usuario,
entonces se realiza la actualización en el registro *)

Activa a SIMODIF7

fin (Si)

fin Si)

FIN (MODIFI7)

10.6 NOMBRE DEL PROCEDIMIENTO: MODIFI7A.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Presenta la información contenida en el campo a ser modificado, identificada como "dato viejo" y solicita el dato nuevo para hacer la actualización. Existe la ratificación del nuevo dato. Es exclusivo del registro cero.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO	DESCRIPCION
PROGRAMA, DISCO	
VA TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
X	Contiene el número del campo que se desea modificar. Es de tipo entero.
UNI	Valor contenido en algunos campos susceptibles de ser modificados. Es de tipo entero.
OTRO	Variable por medio de la cual el usuario tiene comunicación con el sistema para responder a preguntas del mismo. Puede valer S o N u otro caracter reconocido. Es de tipo caracter.

VARIABLES LOCALES:

NOMBRE	DESCRIPCION
TOTFUE	Se usa en caso de que se quiera modificar el valor de las ventas totales del día de artículos fuera del sistema. Es proporcionado por el usuario. Se usa únicamente cuando se pretende modificar el campo PRECIO del registro cero. Es de tipo real.

PROCEDIMIENTOS QUE INVOCA: Ninguno.

PSEUDOCODIGO.

(* Inicia el procedimiento MODIFI7A *)

INICIA (MODIFI7A)

```

En caso que X
  sea igual a 5  Inicia (5)
                  Despliega el mensaje:
                  "DATO VIEJO"
                  (valor del campo UVEN)  "ANOTE EL DATO NUEVO : "
                  Lee el valor de UNI
                  UVEN <--- UNI
                  fin (5)

  sea igual a 6  Inicia (6)
                  Despliega el mensaje:
                  "DATO VIEJO"
                  (valor del campo PRECIO) "ANOTE EL DATO NUEVO : "
                  Lee el valor de TOTFUE
                  PRECIO <--- TOTFUE
                  fin (6)

  sea diferente de 5 o 6  Inicia (diferente)
                          Despliega el mensaje:
                          "UNICAMENTE PUEDE CORREGIR LOS CAMPOS 5 Y 6"
                          OTRO <--- 'C'
                          fin (diferente)

fin (caso)

```

FIN (MODIFI7)

10.8 NOMBRE DEL PROCEDIMIENTO: SIMODIF7.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Actualiza 'físicamente' la información en el campo del registro determinado por el usuario.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO:

NOMBRE DEL ARCHIVO PROGRAMA	DISCO	DESCRIPCION
VA	TRABAJO.FIL	ARCHIVO DE TRABAJO

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOKA:

NOMBRE	OBJETIVO
SEEKWRITE	Realiza las escrituras a los archivos.
ERRCHK	Valida lecturas y escrituras a los archivos.
EXIT	Dar por terminada la sesión.

PSEUDOCODIGO.

(* Inicia el procedimiento SIMODIF7 *)

INICIA (SIMODIF7)

Escribir en el registro I del Archivo VA
Activa a ERRCHK
Si ERROR es verdadero entonces
Activa a EXIT
fin (si)

FIN (SIMODIF7)

10.7 NOMBRE DEL PROCEDIMIENTO: CUALREG7.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Solicita el número del registro a ser modificado; verifica y acepta o rechaza el dato.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguno.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.
LIMIT7	Contiene el valor del máximo registro útil del archivo de Trabajo. Es de tipo entero.
OTRO	Variable por medio de la cual el usuario tiene comunicación con el sistema para responder a preguntas del mismo. Puede valer 'Q' o cualquier caracter reconocido. Es de tipo caracter.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CUALCAM7	Pide el número del campo a ser modificado; verifica y acepta o rechaza los datos.

PSEUDOCODIGO:

(* Inicia el procedimiento CUALREG7 *)

INICIA (CUALREG7)

OTRO <--- ' '

Repite

 Despliega el mensaje:

 ' CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => '

 Lee el valor de I

 Si I es mayor o igual a cero

 y I es menor o igual a LIMIT7 entonces

 (* El registro solicitado esta dentro del rango *)

 Activa a CUALREG7

 sino

 Despliega el mensaje:

 ' * * * E R R O R * * * '

 NO EXISTE EL REGISTRO (valor de I),

 ESTÁ FUERA DEL RANGO (valor de LIMIT7) '

 fin (Si)

 Despliega el mensaje:

 'SI DESEA SEGUIR ACTUALIZANDO TRANSMITA <RETURN>

 EN CASO CONTRARIO ANOTE LA LETRA Q => '

 Lee el valor de OTRO

 Hasta que OTRO sea igual a Q o a q

FIN (CUALREG7)

10.8 NOMBRE DEL PROCEDIMIENTO: CUALCAM7.

PROGRAMA DONDE SE LOCALIZA: MODU7.

OBJETIVO: Pide el número del campo a ser modificado; verifica y acepta o rechaza los datos.

ARCHIVOS QUE UTILIZA EL PROCEDIMIENTO: Ninguna.

VARIABLES.

VARIABLES EXTERNAS: Ninguna.

VARIABLES MODULARES:

NOMBRE	DESCRIPCION
X	Contiene el número del campo que se desea modificar. Es de tipo entero.
I	Apuntador para acceder a los registros del Archivo de Trabajo VA. Es de tipo entero.
OTRO	Variable por medio de la cual el usuario tiene comunicación con el sistema para responder a preguntas del mismo. Puede valer N o cualquier caracter reconocido. Es de tipo caracter.

VARIABLES LOCALES: Ninguna.

PROCEDIMIENTOS QUE INVOCA:

NOMBRE	OBJETIVO
CHECA7	Realiza las lecturas al archivo de trabajo, las verifica por medio del procedimiento ERRCHK.
CABEZA7	Desplegar la identificación (número y nombre) de los campos que son susceptibles de ser modificados.
MODIFI7	Presenta la información contenida en el campo a ser modificado, identificada como "dato viejo" y solicita el dato nuevo para hacer la actualización. Exige la ratificación del nuevo dato.

PSEUDOCORRIGO.

(* Inicia el Procedimiento CUALCAM7 *)

INICIA (CUALCAM7)

Activa a CHECA7
Activa a CABEZA7

Repite

Despliega el mensaje:

"CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => "

Lee el valor de X

Si X es mayor que 0 y menor o igual a 6 entonces

(* es un campo valido *)

Activa a MODIFI7

sino

Despliega el mensaje:

" * * * E R R O R * * *

NO EXISTE EL CAMPO (valor de X)

ESTA FUERA DEL RANGO "

fin (Si)

Despliega el mensaje:

"DESEA CAMBIAR OTRO CAMPO DEL REGISTRO (valor de I) ?

ANOTE SI o NO => "

Lee el valor de OTRO

Hasta que OTRO sea igual a N o n

FIN (CUALCAM7)

[378]

C A P I T U L O I V

M A N U A L D E O P E R A C I O N

D E L

S I S T E M A

IV.1 M A N U A L D E O P E R A C I O N
D E L S U B S I S T E M A D E
A C T U A L I Z A C I O N
Y
C O N S U L T A
D E
R E Q U E R I M I E N T O S

IV.1.1 MANUAL DE OPERACION DEL MODULO SABADO

MODULO PRINCIPAL DEL SUBSISTEMA DE ACTUALIZACION
Y CONSULTA DE REQUERIMIENTOS

1. PANORAMA ESPECIFICO.

OBJETIVO: Controlar el proceso de ejecución del Subsistema de Actualización y Consulta de Requerimientos.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL.

RESTRICCIONES: Es importante que cada vez que actualize información del Sistema se salga y vuelva a entrar, para así obtener una mejor protección en su información.

2. PROCEDIMIENTO:

Para entrar al Subsistema de Actualización y Consulta de Requerimientos, transmite la siguiente palabra.

SABADO

Si el archivo Maestro no se encuentra en el disco que está utilizando como el de datos, entonces el programa le desplegará el siguiente mensaje:

**** ERROR EL DISCO NO CONTIENE EL ARCHIVO MAESTRO DE DATOS ****

Y lo sacará de sesión, por lo tanto asegúrese de trabajar con el disco adecuado que contenga los archivos correspondientes, para que funcione el programa, los cuales son los siguientes: ARCHIVO MAESTRO (MAESTRO.FIL), ARCHIVO DE NOMBRES Y ESTADISTICAS (NOMYEST.FIL), ARCHIVO DE TRABAJO (TRABAJO.FIL), ARCHIVO DE CLAVES DISPONIBLES (CVEDISP.FIL) y el ARCHIVO DE PEDIDOS (PEDIDOS.FIL).

Si el archivo Maestro se encuentra presente en el disco, donde contiene sus archivos de datos para el Sistema, entonces al transmitir la palabra SABADO, queda automáticamente dentro del Subsistema, e inmediatamente el programa le da la bienvenida de la siguiente manera:

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

**** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS ****

** FECHA DEL ULTIMO DIA DE ACCESO, DIA: ND MES: NM AÑO: NA **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

Donde ND es el día, NM es el mes y NA es el año en que se consultó, por última vez el subsistema.

Después el programa le pedirá le transmita la fecha de trabajo del día en que está consultando al Subsistema, de la siguiente manera:

DEME AÑO => AA

Donde AA es un número de dos dígitos, y debe ser igual o mayor en una unidad a NA. En caso que usted transmita un número menor o mayor a 2 unidades que NA, el programa le desplegará el siguiente mensaje de error y le pedirá, vuelva a transmitir el dato.

ERROR : EL AÑO DEBE SER MAYOR O IGUAL AL DE LA FECHA DE LA
ULTIMA CONSULTA AL SUBSISTEMA Y NO MAYOR A UN AÑO
DE DIFERENCIA, FAVOR DE VOLVER A TRANSMITIR EL AÑO

DEME AÑO => AA

En caso que usted transmita un año mayor al de la fecha de la última consulta y la diferencia de meses no sea menor o igual a dos, entonces el programa le desplegará el siguiente mensaje de error.

ERROR : FAVOR DE DAR EL AÑO IGUAL AL DE LA ULTIMA CONSULTA
DEBIDO A QUE LA DIFERENCIA DE MESES SERA MAYOR A DOS

DEME AÑO => AA

Transmitido el año correctamente, el programa le pedirá ahora, transmita el mes de la siguiente manera,

DEME MES => MM

Donde MM es un número comprendido entre el 1 y el 12, y debe ser igual o menor a dos unidades que MM. En caso que usted transmita un número menor o mayor a dos unidades del mes de la última consulta, el programa le desplegará el siguiente mensaje de error y le pedirá vuelva a transmitir el dato.

ERROR : EL MES DEBE SER MAYOR O IGUAL AL DE LA FECHA DE LA
ULTIMA CONSULTA AL SUBSISTEMA Y NO MAYOR A DOS ME-
SES DE DIFERENCIA, FAVOR DE VOLVER A DAR EL MES

DEME MES => MM

Si usted transmite un número no comprendido entre el 1 y el 12, el programa le desplegará el siguiente mensaje de error y le pedirá, vuelva a transmitir el dato.

ERROR : EL MES MM NO EXISTE, FAVOR DE VOLVER A TRANSMITIR
EL MES

DEME MES => MM

Si usted ha transmitido un año más, que el de la última consulta, entonces deberá dar como máximo el segundo mes de ese año, en caso contrario el programa le desplegará el siguiente mensaje de error y le pedirá vuelva a transmitir el dato.

ERROR : EL MES DEBE SER EL PRIMERO O MAXIMO EL SEGUNDO DEL
AÑO QUE ACABA DE TRASMITIR

DEME MES => MM

Transmitido el mes correctamente el programa le pedirá transmita el día de la siguiente manera.

DEME DIA => DD

Donde DD es un número comprendido entre el 1 y el 31. Si usted transmite un número menor al día de la última consulta, el programa le desplegará el siguiente mensaje de error y le pedirá vuelva a transmitir el dato.

ERROR : EL DIA DEBE SER MAYOR O IGUAL AL DE LA FECHA DE LA
ULTIMA CONSULTA AL SUBSISTEMA, FAVOR DE VOLVER A
TRANSMITIR EL DIA

DEME DIA => DD

Si transmite un día que no existe en el mes, el programa le desplegará el siguiente mensaje de error y le volverá a pedir transmita nuevamente el dato.

ERROR : EL DIA DD DEL MES MM NO EXISTE, FAVOR DE VOLVER A TRANSMITIR EL DIA

DEME DIA => DD

Transmitido el día correctamente el programa le desplegará la siguiente pregunta:

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Si transmite cualquier caracter diferente de S o de N , el programa le desplegará los siguientes mensajes:

ERROR TRANSMITA UNICAMENTE <S> O <N>.

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Si transmite una letra N, el programa le despliega la siguiente pregunta:

DEME LA FUNCION QUE DESEA =>

Pero si transmite una S, el programa le desplegará el menú del Subsistema de Actualización y Consulta de Requerimientos, de la siguiente manera.

FUNCION	OPERACION
---------	-----------

A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGO DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SUBSISTEMA

DEME LA FUNCION QUE DESEA =>

Si transmite un caracter diferente a las letras de la función que se le indican, el programa le desplegará el siguiente mensaje de error y le pedirá vuelva a transmitir el dato.

*** E R R O R ***

TRANSMITA UNICAMENTE DE LA <A> A LA <K>

DEME LA FUNCION QUE DESEA =>

Si transmite correctamente la letra de la función que desea ejecutar, el subsistema le permitirá entrar al programa indicado.

A partir de aquí le sugerimos consulte los manuales correspondientes a cada uno de los programas que realizan las funciones señaladas.

Para salirse del Subsistema transmita la letra <K>, y quedará fuera del programa e inmediatamente le desplegará el siguiente mensaje de terminación.

*** TERMINA PROGRAMA DE ACTUALIZACION Y CONSULTA ***
*** DE REQUERIMIENTOS ***

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Este programa no genera un reporte, sino que despliega el menú de las funciones que se pueden ejecutar dentro del Subsistema. Este menú puede aparecer ya sea por pantalla o por impresora.

4. COMENTARIOS GENERALES.

Es muy importante que la fecha que transmite al iniciar su trabajo, vaya de acuerdo a la fecha real del día, ya que si transmite una fecha no mayor a dos meses de diferencia, de la fecha de la última consulta, el Subsistema la aceptará.

IV.1.2 MANUAL DE OPERACION DEL MODULO SUGPED

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:
SUGERIR PEDIDOS

1. PANORAMA ESPECIFICO.

OBJETIVO: Sugerir pedidos de los medicamentos, a los cuales sus unidades en existencia están abajo del 20 %.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL y Archivo de Pedidos, llamado PEDIDOS.FIL .

RESTRICCIONES: Es importante señalar que si hay alguna falla de corriente eléctrica en el momento en que está haciendo pedidos, todo los pedidos que se hayan hecho se habrán perdido y hay que volver a iniciar el proceso de sugerir pedidos.

2. PROCEDIMIENTO:

Después que el programa le haya desplegado el menú de las funciones que puede usted ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le despliega el siguiente mensaje)

DEME LA FUNCION QUE DESEA =>

Cuando esto suceda transmita la letra A, y quedará dentro del programa de Sugerir Pedidos.

Si el archivo Maestro no contiene información, el programa le desplegará los siguientes mensajes :

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO.

*** TERMINA PROGRAMA DE PEDIDOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Con relación a esta pregunta le sugerimos, consulte el manual del Programa Principal, donde se le explica lo que debe hacer ante esta situación.

En caso que el archivo Maestro contenga información, el programa le desplegará por pantalla alguna información de los medicamentos que están abajo del 20 % de sus unidades en existencia. Estos datos se le proporcionan con la finalidad de que tenga, una mejor panorámica acerca de la venta del producto, que en ese momento le está apareciendo en pantalla. Por ejemplo los datos que se le proporcionan son; el nombre del medicamento, la clave del medicamento, las unidades que hay en ese momento en existencia, apareciéndole abajo la fecha del día en que está siendo consultado el Subsistema, también se le proporcionan las unidades que había en el momento de empezarse a vender esa remesa, con la fecha del día en que se inicio la venta del medicamento, se le proporciona los días que lleva de venta, se le dan también los precios y descuentos que le hacen los proveedores, además se le sugiere la cantidad de medicamentos a comprar, esta sugerencia se hace en base a acompletar el 100 %, de las unidades que había en existencia antes de empezarse a vender la remesa, y por último se le proporciona la clave del proveedor que le surte ese medicamento. Toda esta información se le dá con la finalidad de que tenga una mejor opción a la hora de hacer la compra de algún medicamento.

Esta información le será listada en base al número de la clave del medicamento, en orden ascendente y le pararecerá como sigue (los datos que le aparecen por pantalla son de ejemplo):

NOMBRE DEL MEDICAMENTO : XXXXXXXXXXXXXXXXXXXXXXXXXXXXX

CLAVE	UNIDADES EN EXIT. (31/12/84)	UNIDADES QUE HABIA (31/01/85)	DIAS DE VENTA TRANSCURRIDOS DE ESTA CANTIDAD DE UNIDADES		
1	10	100	31 DIAS		
	PRECIO X UNIDAD		DESCUENTO DEL PROVEEDOR		
PROVEEDOR1	PROVEEDOR2	PROVEEDOR3	1	2	3
\$ 1000.00	\$ 1200.00	\$ 1100.00	10 %	5.0 %	1.0 %

CANTIDAD DE MEDICAMENTOS
QUE SE RECOMIENDAN COMPRAR

ACTUALMENTE ESTE ARTICULO
ES SURTIDO POR EL PROVEEDOR

90

1

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO S)I O N)O =>

Si transmite un caracter diferente de S ó N el programa le desplegará el siguiente mensaje de error, y le volverá a pedir de nuevo el dato;

ERROR TRANSMITA UNICAMENTE <S> O <N>
FAVOR DE VOLVER A TRANSMITIR S)I O N)O =>

En caso que transmita la letra N, el programa le desplegará la siguiente pregunta;

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó N, le aparecerá por pantalla de nuevo el mismo mensaje de error visto ya anteriormente. Si transmite una letra N, el programa le desplegará los siguientes mensaje;

*** NO GRABO NINGUN PEDIDO ***

*** TERMINA PROGRAMA DE PEDIDOS ***

E inmediatamente le despliega el siguiente mensaje;

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Ante esta situación le sugerimos regresar a la parte correspondiente a esta pregunta.

Continuando con la siguiente pregunta;

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

Ya hemos visto lo que sucede cuando, se le transmite un caracter diferente de S ó N, ó cuando se le trasmite una letra N, ahora veamos que ocurre cuando se le transmite una letra S.

Si transmite una S, el programa le listará la información del siguiente medicamento, al cual sus unidades en existencia se encuentran abajo del 20 %, e inmediatamente después de esto, el programa le desplegará la siguiente pregunta;

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O =>

Con respecto a esta pregunta, anteriormente ya se explicó lo que sucede cuando se le transmite un caracter diferente de S ó N, y cuando se le transmite una letra N, veamos ahora lo que ocurre cuando se le transmite una letra S.

Si transmite una letra S, el programa le desplegará la siguiente pregunta:

DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD SUGERIDA?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó N, el programa le desplegará el mensaje de error visto ya anteriormente.

Si transmite una letra S, el pedido queda grabado con la información desplegada anteriormente y le aparece por pantalla los siguientes mensajes:

GRABA PEDIDO

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA AHORA PEDIDO?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S o N, el programa le desplegará el mensaje de error visto ya anteriormente.

Si transmite una S, el programa le desplegará el monto de lo que lleva pedido, hasta ese momento, de la siguiente manera:

HASTA AHORA LLEVA PEDIDO.- \$ 99999.99

Y después de este mensaje, el programa le despliega la siguiente pregunta vista ya anteriormente:

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

A partir de esta pregunta se repite el proceso de suscribir pedidos, así que le sugerimos regrese a esta pregunta, donde se le explica lo que debe hacer.

Ahora volvamos a la siguiente pregunta:

DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD SUGERIDA?, TRANSMITA S)I O N)O =>

Como ya anteriormente se ha analizado lo que ocurre, cuando se le transmite un caracter diferente de S ó N, ó cuando se le transmite una letra N, ahora veamos que sucede, cuando se le transmite una letra S.

Cuando se transmite una letra S, el programa le despliega el siguiente mensaje:

A QUE PROVEEDOR DESEA HACER EL PEDIDO?
DEME NUMERO DE PROVEEDOR => N

Donde N debe ser un número comprendido entre 1 y 3, en este caso se está suponiendo, que se cuenta con solo tres proveedores, es importante recalcar que si usted transmite, cualquier otro número el programa le mandará un mensaje de error y le volverá a solicitar el número de proveedor.

Después de haber transmitido correctamente el número de proveedor, el programa de SUGPED le desplegará por pantalla el siguiente mensaje:

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó de N, el programa le desplegará de nuevo el mensaje de error visto ya anteriormente.

Si transmite una letra S, el programa le despliega la siguiente pregunta:

CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO? => NU

Donde NU es el número de unidades que debe transmitir, transmitido el número de unidades, el programa le despliega la siguiente pregunta:

DEME PRECIO POR UNIDAD => P

Donde P es el precio que debe transmitir, transmitido el precio el programa le despliega la siguiente pregunta:

DEME EL DESCUENTO QUE LE HACE EL PROVEEDOR => D

Donde D es el descuento que le hace el proveedor, puede ser un número real o entero. Si es entero el número debe estar comprendido entre el 1 y el 99, sino desplegará el siguiente mensaje de error y le volverá a solicitar de nuevo el dato:

ERROR EL DATO ESTA EQUIVOCADO REBASA EL 100 %
FAVOR DE DAR EL DESCUENTO =>

Si es real, el número debe estar comprendido entre un número mayor que cero y un número menor que uno, sino volverá a desplegar el mismo mensaje de error y de nuevo volverá a solicitar el dato.

Transmitido el descuento correctamente, el programa le volverá a desplegar el siguiente mensaje:

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA AHORA PEDIDO?, TRANSMITA S)I O N)O =>

Como esta situación ya ha sido explicada anteriormente, le sugerimos redrese a consultar de nuevo esta presunta.

Continuando con el desarrollo del proceso, redresemos a la siguiente presunta:

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O =>

Como ya anteriormente se ha explicado lo que sucede, cuando se le transmite un caracter diferente de S ó de N, ó cuando se le transmite una letra S, ahora veamos que ocurre cuando se le transmite una letra N.

Cuando transmite una letra N, el programa le despliega por pantalla la siguiente presunta:

DESEA DAR LA CANTIDAD DE MEDICAMENTOS A COMPRAR? TRANSMITA, S)I O N)O =>

Si transmite un caracter diferente de S o de N, el programa le volverá a desplegar el mensaje de error explicado ya anteriormente.

Si transmite una letra N, el programa le desplegará el siguiente mensaje:

SE TOMA LA CANTIDAD SUGERIDA

Pero si transmite una letra S, le desplegará la siguiente presunta:

CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO? => NUS

Donde NUS es el número de unidades solicitadas, transmitido el número de unidades solicitadas, el programa le despliega la siguiente presunta:

DESEA DAR EL PRECIO POR MEDICAMENTO?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó de N, el programa le volverá a desplegar de nuevo el mensaje de error explicado ya anteriormente.

Si transmite una letra N, el Programa le desplega el siguiente mensaje;

SE TOMA EL PRECIO DEL PROVEEDOR ESCOGIDO

Pero si transmite una letra S, el programa le desplegará la siguiente pregunta;

DEME PRECIO POR UNIDAD => P

Donde P es número entero o real. Transmitido el precio el programa le desplega la siguiente pregunta;

DESEA DAR EL DESCUENTO DEL PROVEEDOR?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó de N, el programa le desplegará el mensaje de error explicado anteriormente.

Si transmite una letra N, el programa le desplegará el siguiente mensaje;

SE TOMA EL DESC. DEL PROVEEDOR ESCOGIDO

Pero si transmite una letra S, el programa le desplegará la siguiente pregunta;

DEME EL DESCUENTO QUE LE HACE EL PROVEEDOR => D

Donde D es un número real o entero, si es entero debe estar comprendido entre el 1 y el 99, Pero si es real debe ser mayor que 0 y menor que 1, en caso que no transmita correctamente el descuento, el programa le desplegará el siguiente mensaje de error explicado ya anteriormente;

ERROR EL DATO ESTÁ EQUIVOCADO REBASA EL 100 %
FAVOR DE DAR EL DESCUENTO =>

Como esta situación ya fue explicada, continuaremos con la descripción del programa de Suscribir Pedidos,

Transmitido correctamente el descuento del proveedor, el programa le volverá a desplegar el siguiente mensaje;

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO, TRANSMITA S)I N)O =>

Como esta situación ya fue explicada anteriormente, le suserimos redrese a la parte correspondiente donde se describe esta situación.

Este proceso se repite cuantas veces lo requiera, o hasta que se acaben los medicamentos que tienen sus unidades abajo del 20 %.

Transmitido el descuento correctamente, el programa le volverá a desplegar el siguiente mensaje:

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA AHORA PEDIDO?, TRANSMITA S)I O N)O =>

Como esta situación ya ha sido explicada anteriormente, le sugerimos regresar a consultar de nuevo esta pregunta.

Continuando con el desarrollo del proceso, regresemos a la siguiente pregunta:

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O =>

Como ya anteriormente se ha explicado lo que sucede, cuando se le transmite un caracter diferente de S ó de N, ó cuando se le transmite una letra S, ahora veamos que ocurre cuando se le transmite una letra N.

Cuando transmite una letra N, el programa le despliega por pantalla la siguiente pregunta:

DESEA DAR LA CANTIDAD DE MEDICAMENTOS A COMPRAR? TRANSMITA, S)I O N)O =>

Si transmite un caracter diferente de S o de N, el programa le volverá a desplegar el mensaje de error explicado ya anteriormente.

Si transmite una letra N, el programa le desplegará el siguiente mensaje:

SE TOMA LA CANTIDAD SUGERIDA

Pero si transmite una letra S, le desplegará la siguiente pregunta:

CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO? => NUS

Donde NUS es el número de unidades solicitadas, transmitido el número de unidades solicitadas, el programa le despliega la siguiente pregunta:

DESEA DAR EL PRECIO POR MEDICAMENTO?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó de N, el programa le volverá a desplegar de nuevo el mensaje de error explicado ya anteriormente.

Si transmite una letra N, el programa le desplegará el siguiente mensaje;

SE TOMA EL PRECIO DEL PROVEEDOR ESCOGIDO

Pero si transmite una letra S, el programa le desplegará la siguiente pregunta;

DEME PRECIO POR UNIDAD => P

Donde P es número entero o real. Transmitido el precio el programa le desplegará la siguiente pregunta;

DESEA DAR EL DESCUENTO DEL PROVEEDOR?, TRANSMITA S)I O N)O =>

Si transmite un caracter diferente de S ó de N, el programa le desplegará el mensaje de error explicado anteriormente.

Si transmite una letra N, el programa le desplegará el siguiente mensaje;

SE TOMA EL DESC. DEL PROVEEDOR ESCOGIDO

Pero si transmite una letra S, el programa le desplegará la siguiente pregunta;

DEME EL DESCUENTO QUE LE HACE EL PROVEEDOR => D

Donde D es un número real o entero, si es entero debe estar comprendido entre el 1 y el 99, pero si es real debe ser mayor que 0 y menor que 1, en caso que no transmita correctamente el descuento, el programa le desplegará el siguiente mensaje de error explicado ya anteriormente;

ERROR EL DATO ESTÁ EQUIVOCADO REBASA EL 100 %
FAVOR DE DAR EL DESCUENTO =>

Como esta situación ya fue explicada, continuaremos con la descripción del programa de Sugerir Pedidos.

Transmitido correctamente el descuento del proveedor, el programa le volverá a desplegar el siguiente mensaje;

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO, TRANSMITA S)I N)O =>

Como esta situación ya fue explicada anteriormente, le sugerimos regresar a la parte correspondiente donde se describe esta situación.

Este proceso se repite cuantas veces lo requiera, o hasta que se acaben los medicamentos que tienen sus unidades abajo del 20 %.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Este programa no genera un reporte en especial, sino que despliega por pantalla algunos datos estadísticos (los cuales ya han sido explicados en el manual de procedimiento) del medicamento, estos datos también pueden ser listados por impresora en caso que así se requiera.

4. COMENTARIOS GENERALES.

Este programa puede ser utilizado tanto por pantalla como por impresora, pero se le recomienda que lo utilice por pantalla, ya que es más rápido su proceso.

IV.1.3 MANUAL DE OPERACION DEL MODULO LISPED

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

LISTAR PEDIDOS

1. PANORAMA ESPECIFICO.

OBJETIVO: Listar la información del archivo de Pedidos, de los pedidos que se han suserido comprar.

ARCHIVOS QUE USA: Archivo de Pedidos, llamado PEDIDOS.FIL y el Archivo de Nombres y Estadísticas, llamado NOMYEST.FIL .

RESTRICCIONES: El archivo de Pedidos siempre mantendrá la última información de los pedidos hechos, hasta que se hayan vuelto hacer nuevos pedidos.

2. PROCEDIMIENTO:

Después que el programa le haya desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Cuando esto suceda transmita una letra B y si el archivo de Pedidos aún no contiene información el programa le desplegará los siguientes mensajes:

*** ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION ***
*** FAVOR DE VOLVER A LISTAR EL MENU Y EJECUTE LA OPCION ***
*** <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO ***

*** TERMINA PROGRAMA DE LISTAR PEDIDOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Ante esta pregunta le sugerimos consulte el manual de descripción del Programa Principal, donde se le explica que debe hacer ante esta situación.

Si el archivo de Pedidos contiene información, el programa le empezará a listar por proveedor los siguientes datos de los medicamentos que se han sugerido comprar, iniciando primeramente con el encabezado del reporte.

REPORTE DE PEDIDOS DE MEDICAMENTOS DEL DIA DD DE XXXXXXXX DE 19AA

PROVEEDOR : Nombre del Proveedor

CLAVE	NOMBRE DEL MEDICAMENTO	NUM DE UNID.	PRECIO	DESCUENTO	PAGO NETO
-------	------------------------	--------------	--------	-----------	-----------

Al final de cada lista de los medicamentos, que surte cada proveedor, le desplegará el monto total de lo que se pidió, de la siguiente manera:

TOTAL DEL PEDIDO A ESTE PROVEEDOR.- \$ 9999.99

Así para cada proveedor le aparecerá un listado, con estos datos.

Al final del programa de listar pedidos, le desplegará el siguiente mensaje de terminación de programa:

*** TERMINA PROGRAMA DE LISTAR PEDIDOS ***

E inmediatamente después el programa le desplega el siguiente mensaje visto ya anteriormente:

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Y a partir de aquí le sugerimos, consulte la parte correspondiente a esta descripción, que se localiza en el manual del Programa Principal.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Esta función genera el reporte de los pedidos que se han solicitado, y la información que se despliega en el listado es la siguiente:

PROVEEDOR. Nombre del proveedor, que surte este medicamento.

CLAVE. Número de clave del medicamento que surte este proveedor

NOMBRE DEL MEDICAMENTO. Nombre del medicamento.

NUM DE UNID. Número de unidades que se desean pedir del medicamento.

PRECIO. Precio de compra del medicamento al proveedor.

DESCUENTO. Descuento que hace el proveedor al medicamento.

PAGO NETO. Pago neto por medicamento.

4. COMENTARIOS GENERALES.

Estos reportes se pueden desplegar por pantalla, pero es preferible que se haga por impresora.

IV.1.4 MANUAL DE OPERACION DEL MODULO PEDXTEM

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

SUGERIR COMPRA DE ARTICULOS POR TEMPORADA

1. PANORAMA ESPECIFICO.

OBJETIVO: Sugerir compra de medicamentos por temporada.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL
Archivo de Pedidos, llamado PEDIDOS.FIL .

RESTRICCIONES: El programa consulta una tabla de estadísticas sobre las ventas, de los medicamentos, esta al principio no contendrá información, así que cuando elija esta opción por primera vez lo más probable es que no obtenga ningún resultado.

2. PROCEDIMIENTO:

Después que el programa le haya desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Transmita una letra C, y automáticamente quedará dentro del programa de Sugerir Pedidos por Temporada (PEDXTEM).

Si el Archivo Maestro aun no contiene información el programa le desplegará los siguientes mensajes:

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

**** TERMINA PROGRAMA DE PEDIDOS POR TEMPORADA ****

Pero si el Archivo Maestro contiene información el programa le desplegará lo siguiente:

TEMPORADA	PERIODO
A	DEL 21 DE MARZO AL 20 DE JUNIO
B	DEL 21 DE JUNIO AL 23 DE SEPTIEMBRE
C	DEL 24 DE SEPTIEMBRE AL 22 DE DICIEMBRE
D	DEL 23 DE DICIEMBRE AL 20 DE MARZO

DEME TEMPORADA QUE DESEA =>

Si transmite un caracter diferente de las letras indicadas, el programa le desplegará el siguiente mensaje de error:

*** ERROR ***

TRANSMITA UNICAMENTE DE LA <A> A LA <D>

DEME TEMPORADA QUE DESEA =>

Transmitida correctamente la letra de la temporada que desea consultar el programa le desplegará el siguiente mensaje:

DEME EL NUM DE UNIDADES QUE PARA USTED REFLEJEN
UNA BUENA VENTA => NN

Donde NN es un número entero, que representa la cantidad de unidades que debieron haberse vendido, para que fuera una buena venta, en la temporada que está consultando.

Transmitido NN, el programa empezará a listar aquellos medicamentos, que satisfagan el requisito de unidades vendidas en la temporada elegida.

La información que se le proporciona es la siguiente:

CLAVE	NOMBRE DEL MEDICAMENTO	UNIDADES VENDIDAS EN ESTA TEMPORADA	UNIDADES EN EXISTENCIA
1	XXXXXXXX	90	10
	CANTIDAD DE MEDICAMENTOS QUE SE RECOMIENDAN COMPRAR	ACTUALMENTE ESTE MEDICAMENTO ES SURTIDO POR LE PROVEEDOR	
	80	1	

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O =>

Si transmite un caracter diferente de S ó de N, el programa le desplegará los siguientes mensajes:

ERROR TRANSMITA UNICAMENTE <S> O <N>
FAVOR DE VOLVER A TRANSMITIR S)I O N)O =>

Si transmite una letra N el programa le desplegará la siguiente presunta:

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

Si transmite en esta presunta, un caracter diferente de S ó N, el programa le desplegará el mensaje de error explicado ya anteriormente.

Si transmite una letra N, el programa le desplegará los siguientes mensajes:

*** NO SE GRABO NINGUN PEDIDO ***

**** TERMINA PROGRAMA DE PEDIDOS POR TEMPORADA ****

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Ante esta presunta le sugerimos, consulte la descripción del manual del programa Principal donde viene explicado lo que se debe hacer ante esta situación. Resresando a la siguiente presunta:

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

Y como ya se ha explicado lo que ocurre, cuando se le transmite un caracter diferente de S ó de N, ó cuando se le transmite una letra N, ahora veamos que sucede cuando se le transmite una letra S.

Cuando se transmite una letra S, el programa le despliega la información del siguiente medicamento, el cual cumple con el requisito de unidades vendidas en la temporada que se está consultando. En caso que ya no haya más medicamentos que cumplan con la condición, el programa le desplegará los mensajes de terminación de programa, vistos ya anteriormente.

Redresando a la siguiente pregunta;

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO?, S)I O N)O =>

A partir de esta pregunta el proceso, que se sigue es idéntico, al que se lleva a cabo en la descripción del Manual de Suscribir Pedidos, por lo cual se le sugiere que consulte dicho manual, donde se le da una explicación detallada de como proporcionar los datos para la elaboración del pedido, después que le aparezca de nuevo la siguiente pregunta, redrese a este manual y continúe con el proceso del programa.

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

Como esta situación ya ha sido explicada anteriormente, solo nos resta decirle que el proceso se repetirá, hasta que ya no exista ningún medicamento que satisfaga la condición de unidades vendidas o que el archivo Maestro ya no tenga más información.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

La función esencial de este programa no es generar un reporte, sino suscribir algún pedido por temporada de medicamentos, claro que estos resultados también pueden ser obtenidos por impresora.

4. COMENTARIOS GENERALES:

Es muy importante que tome cuenta, que las tablas de las temporadas que se manejan, son el resultado de una estadística que se lleva sobre las ventas de los medicamentos, por temporada.

IV.1.5 MANUAL DE OPERACION DEL MODULO CADUCA

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS
A DARSE DE BAJA POR CADUCIDAD

1. PANORAMA ESPECIFICO.

OBJETIVO: Obtener un reporte de los medicamentos que están próximos a caducar, tomando como referencia la fecha del reporte.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL y Archivo de Nombres y Estadísticas, llamado NOMYEST.FIL .

RESTRICCIONES: Ninsuna.

2. PROCEDIMIENTO:

Una vez que el programa Principal, le ha desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

cuando esto suceda transmita la letra D, y quedará dentro del programa de Caducidad (CADUCA).

Si el archivo Maestro no contiene información, el programa le desplegará los siguientes mensajes:

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA OPCION
<F>, PARA DAR DE ALTA ALGUN MEDICAMENTO.

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Como está presunta corresponde al Programa Principal, le sugerimos consulte el manual de éste, donde se le explica lo que debe hacer, ante esta situación.

Si el archivo Maestro contiene información, el programa le solicitará la fecha que desea tenga el encabezado del reporte, de la siguiente forma:

ANOTE LA FECHA DE HOY ANO MES DIA :

ANO => AA

Donde AA es el número del año que desea tenga el reporte del pedido, habiendo transmitido el año el programa le solicita el mes de la misma forma:

MES => MM

Donde MM es el número del mes que desea tenga el reporte del pedido, habiendo transmitido el mes el programa le solicita el día de la misma forma:

DIA => DD

Donde DD es el número del día que desea tenga la fecha del reporte del pedido.

Habiendo transmitido la fecha de la forma en que lo solicitó el programa, este analiza los datos capturados y en caso que alguno de estos este equivocado, es decir que haya transmitido una fecha inexistente, el programa le desplegará el siguiente mensaje de error y le volverá a solicitar la fecha nuevamente.

***** E R R O R *****

ANOTE LA FECHA NUEVAMENTE

SI DESEA CONTINUAR OPRIMA LA TECLA <RETURN>
SE DESEA SALIRSE TRANSMITA LA LETRA <Q> =>

Si transmite la letra Q el programa le desplegará, la siguiente pregunta vista ya anteriormente;

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Como esta pregunta corresponde al manual del Programa Principal le sugerimos consulte dicho manual, donde se le explica lo que debe hacer ante esta situación.

Regresando a la pregunta anterior, el programa le vuelve a solicitar la fecha y la valida nuevamente.

ANO => AA

MES => MM

DIA => DD

Habiendo transmitido correctamente la fecha, el programa le despliega los siguientes mensajes;

ANOTE LA FECHA CORRECTAMENTE ANO AA MES MM DIA DD

RELACION DE MADICAMENTOS QUE CADUCARAN EN UN PERIODO NO MAYOR DE N DIA
ANOTE EL VALOR DE N ; ND

Donde ND es el Número de días, hasta el cual se desea conocer el número de medicamentos que van a caducar.

Transmitido el número de días y como el reporte que genera es para hojas de 132 caracteres, el listado que se le presenta aquí será dividido en dos partes, y este aparece de la siguiente manera;

Al inicio del reporte aparece el siguiente encabezado:

FARMACIA DE DIOS, S. A.

FECHA : AA / MM / DD

RELACION DE MEDICAMENTOS QUE CADUCARAN EN UN PERIODO NO MAYOR DE ND DIAS

Después le aparece el encabezado de los datos del reporte de la siguiente forma:

(Primera mitad de la línea del encabezado del reporte)

```
=====
CLAVE . NOMBRE DEL MEDICAMENTO . UNIDADES
                                     CLAVE DEL :UNIDADES EN :
                                     PROVEEDOR :EXISTENCIA :
=====
```

(Segunda mitad de la línea del encabezado del reporte)

```
=====
ACTUALES          UNIDADES          PEDIDAS
:FECHA DE :FALTAN : CLAVE DEL :UNIDADES :FECHA DE :FALTAN
:CADUCIDAD :X DIAS : PROVEEDOR :PEDIDAS :CADUCIDAD :X DIAS
=====
```

Al final del listado de los medicamentos que caducan, en el número de días transmitido, le aparece la siguiente línea donde se le da el número de medicamentos que caducarán:

EN ND DIAS, CADUCARAN NM MEDICAMENTOS.

Donde NM es el número de medicamentos que van a caducar, dentro de ese periodo.

Desplegando esta línea el programa termina y le vuelve a desplegar el siguiente mensaje visto ya anteriormente:

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Como este mensaje pertenece al Programa Principal, le sugerimos consulte dicho manual.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Este programa genera el reporte con los campos mencionados ya anteriormente en el Procedimiento, ahora pasaremos a la descripción de cada uno de ellos:

CLAVE - Número de clave del medicamento.
NOMBRE DEL MEDICAMENTO - Como su nombre lo dice es el nombre del medicamento.

En UNIDADES ACTUALES se tiene los siguientes campos:

CLAVE DEL PROVEEDOR - Clave del proveedor que surte este medicamento.
UNIDADES EN EXISTENCIA - Unidades que hay en existencia de este medicamento.
FECHA DE CADUCIDAD - Fecha en que caduca este medicamento.
FALTAN X DIAS - Días que faltan para que caduque el medicamento.

En UNIDADES PEDIDAS se tienen los siguientes campos:

CLAVE DEL PROVEEDOR - Clave del proveedor que surte la nueva remesa.

UNIDADES PEDIDAS - Número de unidades del pedido.

FECHA DE CADUCIDAD - Fecha de caducidad de las unidades pedidas.

FALTAN - Número de días que faltan para que caduque la nueva remesa de medicamentos.

4. COMENTARIOS GENERALES.

Si la fecha de caducidad de algún medicamento está puesta en cero, el programa no tomara en cuenta este medicamento, tampoco lo hará si las unidades en existencia y las unidades pedidas están en cero.

IV.1.6 MANUAL DE OPERACION DEL MODULO BAJA

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

DAR DE BAJA ALGUN ARTICULO

1. PANORAMA ESPECIFICO.

OBJETIVO: Dar de baja del Sistema algún medicamento que no sea costeable.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL , Archivo de Nombres y Estadísticas, llamado NOMYEST.FIL y Archivo de Claves Disponibles, llamado CVEDISP.FIL .

RESTRICCIONES: No puede volver a recuperar la información de algún medicamento que por equivocación haya dado de baja, sólo puede volver a cargar de nuevo esa información por medio del Programa de actualizaciones.

2. PROCEDIMIENTO:

Después que el programa le haya desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Cuando esto suceda transmita la letra E y quedará dentro del programa para dar de baja algún medicamento.

Si el archivo Maestro no contiene información, el programa le desplegará los siguientes mensajes;

ERROR: EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA
LA OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

*** TERMINA PROGRAMA DE PEDIDOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

En esta situación se le sugiere consulte, el Manual del Programa Principal, en la parte correspondiente a esta presunta.

En caso que el archivo Maestro contenga información, el programa le desplegará el siguiente mensaje;

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => CM

Donde CM es un número entero. Si transmite un número menor que 1, el programa le desplegará el siguiente mensaje de error y le volverá a solicitar de nuevo el dato;

ERROR LAS CLAVES INICIAN DESDE LA 00001
DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA =>

Si transmite un número mayor al de la máxima clave que contiene información, el programa le desplegará el siguiente mensaje de error;

ERROR SE EXCEDE DE LA CLAVE MAXIMA EXISTENTE
LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES : N

Donde N es el número de la clave máxima que contiene información.

Y le vuelve a solicitar el dato;

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA =>

Si transmite un número de clave, que ya anteriormente ha sido puesta disponible, el programa le desplega el siguiente mensaje y le vuelve a solicitar el dato;

CLAVE PUESTA DISPONIBLE YA ANTERIORMENTE
DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA =>

Transmitido el número de la clave que se desea dar de baja correctamente, el programa le desplegará el siguiente mensaje;

ES CORRECTA LA CLAVE DEL MEDICAMENTO; XXXXXXXXXXXXXXXX
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O =>

Esta presunta se le hace con el fin de que en un momento dado, pueda reconsiderar su determinación de poner disponible la clave transmitida.

Si transmite en esta presunta un caracter diferente de S, el programa no dara de baja la clave transmitida y le desplegará el siguiente mensaje:

DESEA DAR DE BAJA A OTRO MEDICAMENTO?, TRANSMITA S)I O N)O =>

Si transmite cualquier caracter diferente de S, el programa le desplegará el siguiente mensaje:

DESEA LISTAR LAS CLAVES QUE SE ENCUENTRAN
DISPONIBLES?, TRANSMITA S)I O N)O =>

Si transmite cualquier caracter diferente de S, el programa le desplegará los siguientes mensajes:

*** TERMINA PROGRAMA DE BAJA DE MEDICAMENTOS ***

DESEA LISTAR EL MENU, TRANSMITA S)I O N)O =>

En esta presunta le sugerimos consulte el Manual del programa Principal, en la parte correspondiente a esta situación.

Continuando con el proceso de este programa, regresemos a la siguiente presunta:

DESEA LISTAR LAS CLAVES QUE SE ENCUENTRAN
DISPONIBLES?, TRANSMITA S)I O N)O =>

Como anteriormente ya explicamos lo que sucede, cuando se le transmite un caracter diferente de S, ahora veamos que ocurre cuando se le transmite una S. Cuando se transmite una letra S, el programa le desplegará los números de las claves que han sido puestas disponibles, de la siguiente manera:

CLAVES DISPONIBLES

num-clave-1
num-clave-2
.
.
num-clave-n

En caso que no haya ninguna clave disponible, el programa le desplegará el siguiente mensaje:

NO HAY CLAVES DISPONIBLES

E inmediatamente después de este mensaje, el programa le desplegará el mensaje de terminación de programa, visto ya anteriormente.

Resresemos a la siguiente pregunta:

ES CORRECTA LA CLAVE DEL MEDICAMENTO: XXXXXXXXXXXXXXXX
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O =>

Como anteriormente ya explicamos lo que sucede, cuando se le transmite un caracter diferente de S, ahora veamos lo que ocurre, cuando se le transmite una letra S.

Quando se transmite una S, el programa pone como disponible, la clave del medicamento que ha sido dado de baja del Sistema, para cualquier otro medicamento que se desee dar de alta. Después el programa le vuelve a desplegar el siguiente mensaje:

DESEA DAR DE BAJA A OTRO MEDICAMENTO?, TRANSMITA S)I O N)O =>

A partir de aquí el proceso se repite, por lo cual se le sugiere consulte de nuevo a partir de esta pregunta.

Este proceso se repite hasta que, ya no desee dar de baja ningún otro medicamento.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Este programa no genera precisamente un reporte, pero puede ser listado el proceso si así se requiere.

4. COMENTARIOS GENERALES.

Es recomendable que lleve una lista de las claves que son puestas disponibles, ya que pueden ser utilizadas por otros medicamentos, que pueden convenir estén dentro del Sistema.

IV.1.7 MANUAL DE OPERACION DEL MODULO: ESCRIBE

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:
DAR DE ALTA ALGUN ARTICULO

1. PANORAMA ESPECIFICO.

OBJETIVO: Dar de alta nuevos medicamentos.

ARCHIVOS QUE USA: Archivo Maestro y Archivo de Nombres y Estadísticas.

RESTRICCIONES: Deben estar presentes los dos archivos para guardar la información.

Este programa puede correr en pantalla o impresora. Por rapidez y economía, se sugiere hacerlo en la pantalla.

2. PROCEDIMIENTO.

Para entrar a esta función elija la opción <F> del menú principal de funciones.

En cuanto entra a esta función, el sistema despliega el mensaje:

ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION :

CLAVE DEL MEDICAMENTO =>

NOMBRE DEL MEDICAMENTO =>

CLAVE DEL PROVEEDOR PRINCIPAL =>

PRECIO DE COMPRA DEL PROVEEDOR PRINCIPAL =>

DESCUENTO DE COMPRA DEL PROVEEDOR PRINCIPAL =>

CLAVE DEL SEGUNDO PROVEEDOR =>

PRECIO DE COMPRA DEL SEGUNDO PROVEEDOR =>

DESCUENTO DE COMPRA DEL SEGUNDO PROVEEDOR =>

CLAVE DEL TERCER PROVEEDOR =>

PRECIO DE COMPRA DEL TERCER PROVEEDOR =>

DESCUENTO DE COMPRA DEL TERCER PROVEEDOR =>

CLAVE DE MEDICINA SIMILAR 1 =>

CLAVE DE MEDICINA SIMILAR 2 =>

CLAVE DE MEDICINA SIMILAR 3 =>

EL PRECIO DE VENTA DEL MEDICAMENTO =>

EL NUMERO DE UNIDADES EN EXISTENCIA =>

FECHA DE CADUCIDAD DEL MEDICAMENTO

DAME AÑO =>

DAME MES =>

DAME DIA =>

EL IMPUESTO =>

EL PORCENTAJE DE UNIDADES MINIMAS
QUE DEBEN EXISTIR =>

TERMINO DE GRABAR EL MEDICAMENTO CLAVE é

SI DESEA SEGUIR METIENDO INFORMACION TRANSMITA RETURN
EN CASO CONTRARIO ANOTE 0 =>

El sistema le pide los datos uno a continuación del otro. El dato de la clave debe seguir un orden riguroso. La mayor parte de la información que introduce para dar de alta un nuevo artículo, es validada por el mismo sistema; en caso de error, lo detecta y especifica.

Cuando ya no quiera seguir grabando más información, simplemente anote la letra Q, cuando el sistema le haga el siguiente comentario:

SI DESEA SEGUIR METIENDO INFORMACION TRANSMITA RETURN
EN CASO CONTRARIO ANOTE Q =>

En ese momento, se da por terminada la sesión y tiene la posibilidad de listar el menú de funciones para elegir otra.

4. COMENTARIOS GENERALES.

Para dar de alta nuevos artículos, no es forzoso que se haga en una sola sesión. Usted puede hacerlo cuando guste.

IV.1.8 MANUAL DE OPERACION DEL MODULO LISTAR

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

LISTAR LA INFORMACION

1. PANORAMA ESPECIFICO.

OBJETIVO: Listar la información más relevante de los medicamentos que se encuentran dentro de los archivos Maestro y Nombres y Estadísticas.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL , Archivo de Nombres y Estadísticas, llamado NOMYEST.FIL .

RESTRICCIONES: No se permite listar información, en claves que aun no estén contenidas dentro del Sistema.

2. PROCEDIMIENTO:

Después que el programa le haya listado el menú de las funciones que puede usted ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Cuando esto suceda transmita la letra G y quedará dentro del programa de Listar información.

Si el archivo Maestro no contiene información, el programa le desplegará el siguiente mensaje de error:

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO.

Inmediatamente después de este mensaje, el programa le desplegará los siguientes mensajes:

*** TERMINA PROGRAMA DE LISTAR INFORMACION ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Ante esta pregunta, se le sugiere consulte el Manual del Programa Principal, donde se le da una explicación de lo que debe hacer.

Si el archivo Maestro contiene información, el programa le desplegará la siguiente pregunta:

COMO DESEA LISTAR SU INFORMACION?
DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO =>

Si transmite un caracter diferente de <U>, <T>, o <R>, el programa le desplegará el siguiente mensaje de error:

E R R O R
TRANSMITA UNICAMENTE U, T o R

Y de nuevo le vuelve a desplegar la siguiente pregunta:

COMO DESEA LISTAR SU INFORMACION?
DE U)NO EN UNO, T)ODA, O DE UN RANGO A OTRO =>

Si transmite una letra U, el programa le desplegará el siguiente mensaje:

TRANSMITA LA CLAVE DEL MEDICAMENTO, QUE DESEA CONSULTAR => N

Donde N es el número de la clave del medicamento, al cual desea listar su información.

Si N es menor que 0, o mayor al número de la clave máxima que contiene información, el programa le desplegará el siguiente mensaje de error y le volverá a solicitar, transmita nuevamente la clave del medicamento:

EL NUMERO DE LA CLAVE EXCEDE EL RANGO PERMITIDO
LA CLAVE MAXIMA QUE CONTIENE INFORMACION, ES: CM

Donde CM es el número de la clave máxima, que contiene información dentro del Sistema. Después de este mensaje, le vuelve a solicitar el dato;

FAVOR DE VOLVER A TRANSMITIR LA CLAVE =>

Transmitida la clave correctamente, el programa le despliega en línea recta la siguiente información;

CLAVE	MEDICAMENTOS SIMILARES	PRECIO X UNID.	NUM DE UNIDADES	FECHA DE CADUCIDAD	CLAVE DEL VENDEDOR
IMPUESTO %	PORCENTAJE MIN.DE UNI.	UNIDADES VENDIDAS	UNIDADES PEDIDAS	FECHA CAD/PED	

Después que le ha listado la información del medicamento deseado, el programa le desplegará la siguiente pregunta;

DESEA CONSULTAR LA INFORMACION DE OTRO MEDICAMENTO?
TRANSMITA S)I O N)O =>

Si transmite una S, el programa le volverá a desplegar la siguiente pregunta;

TRANSMITA LA CLAVE DEL MEDICAMENTO, QUE DESEA CONSULTAR =>

Pero si transmite, en la pregunta anterior una letra N, el programa le hará la siguiente pregunta;

DESEA CONTINUAR LISTANDO MAS INFORMACION?
TRANSMITA S)I O N)O =>

En esta pregunta, si transmite una letra N, el programa le desplegará el siguiente mensaje de terminación de programa, y le volverá a preguntar si desea listar el menú del programa Principal;

*** TERMINA PROGRAMA DE LISTAR INFORMACION ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Como esta pregunta corresponde al Programa Principal, le sugerimos consulte el manual del Programa Principal, donde se le explica lo que debe hacer, ante esta situación.

Resresando a la siguiente pregunta:

DESEA CONTINUAR LISTANDO MAS INFORMACION?
TRANSMITA S)I O N)O =>

Como en esta pregunta ya se explico anteriormente, lo que sucede cuando se le transmite una letra N, veamos ahora lo que ocurre cuando se le transmite una letra S.

Cuando se le transmite una letra S, el programa le vuelve a desplegar el siguiente mensaje:

COMO DESEA LISTAR SU INFORMACION?
DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO =>

Como en esta pregunta ya se explico lo que sucede, cuando se le transmite un caracter diferente de U, T o R, o cuando se le transmite una letra U, ahora veamos que ocurre cuando se le transmite una letra R.

Cuando se transmite una letra R, el programa le despliega el siguiente mensaje:

DEME CLAVE DE INICIO => CI

Donde CI es la clave del medicamento, desde el cual se desea empezar a listar la informacion de los medicamentos.

Si transmite un número de clave menor que 1, el programa le desplegará el siguiente mensaje de error y le volverá a solicitar la clave:

ERROR : LAS CLAVES INICIAN DESDE LA 00001

DEME CLAVE DE INICIO =>

Si transmite un número de clave, mayor que el número de la clave máxima que contiene información, el programa le desplegará el siguiente mensaje de error y le volverá a solicitar, nuevamente el dato:

ERROR : SE HA EXCEDIDO DEL RANGO PERMITIDO
EL NUMERO DE LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES : CM

Donde CM es la clave máxima que contiene información dentro del Sistema.

DEME CLAVE DE INICIO =>

Transmitida correctamente la clave de inicio, el programa le desplegará el siguiente mensaje:

DEME CLAVE FINAL => CF

Donde CF es la clave del medicamento, hasta el cual se desea listar la información.

Si transmite un número de clave menor al de la clave inicial, el programa le desplegará el siguiente mensaje de error, y nuevamente le volverá a solicitar la clave:

ERROR : TRANSMITA UNA CLAVE MAYOR O IGUAL A LA INICIAL

DEME CLAVE FINAL =>

Si transmite una clave mayor a la clave máxima, que contiene información, el programa le desplegará el siguiente mensaje de error y nuevamente le volverá a solicitar la clave:

ERROR : SE HA EXCEDIDO DEL RANGO PERMITIDO

EL NUMERO DE LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES : CM

Donde CM es la clave máxima, que contiene información dentro del Sistema.

DEME CLAVE FINAL =>

Transmitida correctamente la clave final, el programa le empezará a listar la información de aquellos medicamentos que se encuentren dentro del rango, comprendido entre la clave inicial hasta la clave final. La información que se lista es la descrita en la opción U.

Después que ha terminado de listar la información, el programa le volverá a preguntar lo siguiente:

DESEA CONTINUAR LISTANDO MAS INFORMACION?
TRANSMITA S)I O N)O =>

Como anteriormente, ya se ha explicado lo que sucede cuando se le transmite cualquier caracter, continuemos con el proceso de listar información y transmitamos una letra S. Cuando se le transmite una letra S, el programa le vuelve a desplegar la siguiente pregunta:

COMO DESEA LISTAR SU INFORMACION?
DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO =>

Como en esta pregunta, la única opción que nos falta por explicar es la T, veamos que sucede cuando se le transmite una letra T.

Cuando se transmite una letra T, el programa listará todas las claves de los medicamentos que existán en ese momento, en el archivo Maestro. La información que se lista es la descrita en la opción U.

Cuando termina de listar esta información, el programa vuelve a desplegar la siguiente presunta:

DESEA CONTINUAR LISTANDO MAS INFORMACION?
TRANSMITA S)I O N)O =>

A partir de esta presunta el proceso se repite cuantas veces lo desee, así que le sugerimos regresar a esta presunta más arriba y consulte lo que debe hacer ante esta situación.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

La descripción de la información que genera este programa, ya fue debidamente explicada en el proceso del manual.

4. COMENTARIOS GENERALES.

Con la opción para listar la información de las claves una a una, puede listar el registro cero que contiene información confidencial, para el uso del Sistema.

IV.1.9 MANUAL DE OPERACION DEL MODULO ACTUAL

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

ACTUALIZAR INFORMACION

1. PANORAMA ESPECIFICO.

OBJETIVO: Actualizar información de los medicamentos.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL y Archivo de Nombres y Estadísticas del medicamento, llamado NOMYEST.FIL .

RESTRICCIONES: Si no conoce los números de las claves, que le permiten entrar al programa de actualización, no podrá modificar información.

2. PROCEDIMIENTO:

Una vez que el programa Principal, le ha desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Quando esto suceda transmita la letra H, y quedará dentro del programa de Actualización de Información (ACTUAL).

Si el archivo Maestro no contiene información, el programa le desplegará los siguientes mensajes:

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA OPCION
<F>, PARA DAR DE ALTA ALGUN MEDICAMENTO.

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Y como esta pregunta corresponde al Programa Principal, le sugerimos consulte dicho manual, donde se le explica lo que debe hacer, ante esta situación.

Si el archivo Maestro contiene información, el programa le desplegará lo siguiente:

NOTE LA PRIMERA CONTRASEÑA : CT1

Donde CT1 es el primer número de la contraseña para permitir actualizar, información.

Transmitida la primera contraseña, el programa le solicita la segunda, contraseña de la siguiente manera:

NOTE LA SEGUNDA CONTRASEÑA : CT2

Donde CT2 es la segunda contraseña, para permitirle actualizar información.

Transmitida la segunda contraseña, el programa verifica si estas claves corresponden a las claves de autorización para entrar a actualizar. Si las claves no son las correctas el programa le desplegará los siguientes mensajes de error:

USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Como esta pregunta corresponde al Programa Principal le sugerimos, consulte dicho manual.

Prosiguiendo con nuestro proceso, si las claves fueron las correctas entonces el programa le desplegará el siguiente menú:

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES:

C A M P O
é N O M B R E

C A M P O
é N O M B R E

-
- | | |
|-------------------------------|---------------------------------------|
| 1) MEDICINA SIMILAR 1 | 14) NOMBRE DEL PRODUCTO |
| 2) MEDICINA SIMILAR 2 | 15) UNIDADES DEL RANGO 1 |
| 3) MEDICINA SIMILAR 3 | 16) UNIDADES DEL RANGO 2 |
| 4) PRECIO | 17) UNIDADES DEL RANGO 3 |
| 5) UNIDADES EN EXISTENCIA | 18) UNIDADES DEL RANGO 4 |
| 6) PRIMERA CADUCIDAD | 19) CLAVE DEL PROVEEDOR PRINCIPAL |
| ANO MES DIA | 20) PRECIO DEL PROVEEDOR PRINCIPAL |
| 7) PROVEEDOR PRINCIPAL | 21) DESCUENTO DEL PROVEEDOR PRINCIPAL |
| 8) IMPUESTO | 22) CLAVE DEL SEGUNDO PROVEEDOR |
| 9) PORCENTAJE MINIMO DE UNID. | 23) PRECIO DEL SEGUNDO PROVEEDOR |
| 10) UNIDADES VENDIDAS | 24) DESCUENTO DEL SEGUNDO PROVEEDOR |
| 11) UNIDADES PEDIDAS | 25) CLAVE DEL TERCER PROVEEDOR |
| 12) UNID. VEND. DEL PEDIDO | 26) PRECIO DEL TERCER PROVEEDOR |
| 13) SEGUNDA CADUCIDAD | 27) DESCUENTO DEL TERCER PROVEEDOR |
| ANO MES DIA | 28) FECHA DE MOVIMIENTO |

E inmediatamente le solicita el número del registro que desea modificar, de la siguiente forma:

CUAL REGISTRO QUIERE MODIFICAR? ANOTE EL NUMERO :

Si transmite un número no existente, el programa le despliega los siguientes mensajes:

NO EXISTE, EL REGISTRO N ESTA FUERA DEL RANGO (NM)

SI DESEA SEGUIR ACTUALIZANDO OPRIMA LA TECLA <RETURN>
EN CASO CONTRARIO TRANSMITA LA LETRA <Q> =>

Si transmite RETURN le volverá aparecer, el mismo menú visto anteriormente, y le volverá a solicitar el número del registro al cual desea hacer alguna modificación, de la siguiente manera:

CUAL REGISTRO QUIERE MODIFICAR? ANOTE NUMERO : NUM

Si NUM es un número que esta dentro de los que existen en el Sistema, entonces el programa, le despliega la siguiente pregunta:

CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : NC

Donde NC es un número que debe estar comprendido entre el 1 y el 28, ya que si transmite un número fuera de este rango el programa le desplegará, los siguientes mensajes:

NO EXISTE, EL CAMPO N ESTA FUERA DE RANGO

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO NUM ? ANOTE S)I O N)O :

Si transmite una S, el programa le vuelve a solicitar el número de campo que desea ser modificado, pero si transmite una N, el programa le despliega los siguientes mensajes vistos ya anteriormente:

SI DESEA SEGUIR ACTUALIZANDO OPRIMA LA TECLA <RETURN>
EN CASO CONTRARIO TRANSMITA LA LETRA <Q> =>

Ya se explico lo que sucede cuando se da RETURN, ahora veamos que ocurre cuando se le transmite una Q.

Si en esta situación se transmite una Q, el programa termina, y le despliega el siguiente mensaje:

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Y como esta presunta corresponde al manual del Programa Principal, le sugerimos consulte dicho manual, donde se le explica lo que debe hacer en una situación como esta.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Como el propósito de este programa es que se ejecute por pantalla, para que las actualizaciones sean más rápidas, entonces el programa no genera ningún reporte, claro a menos que desee obtener un listado y utilice como dispositivo de salida una impresora, entonces el proceso descrito en el procedimiento, será el que le aparezca como reporte.

4. COMENTARIOS GENERALES.

Es importante que las claves que utiliza para entrar a actualización solo las conozcan, sientas de mucha confianza, ya que con esto obtendrá, la seguridad que requiere para su información.

IV.1.10 MANUAL DE OPERACION DEL MODULO CARGPED

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

CARGA DE UNIDADES PEDIDAS

1. PANORAMA ESPECIFICO.

OBJETIVO: Cargar al campo de unidades pedidas del Archivo Maestro, la nueva cantidad de unidades pedidas.

ARCHIVOS QUE USA: Archivo Maestro, llamado MAESTRO.FIL .

RESTRICCIONES: No puede cargar unidades, a número de claves que no estén contenidas dentro del Sistema.

2. PROCEDIMIENTO:

Una vez que el programa Principal, le ha desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Cuando esto suceda transmita la letra I, y quedará dentro del programa de Carga de Unidades Pedidas (CARGPED).

Si el archivo Maestro no contiene información, el programa le desplegará los siguientes mensajes:

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA OPCION
<F>, PARA DAR DE ALTA ALGUN MEDICAMENTO.

*** TERMINA PROGRAMA DE CARGA DE UNIDADES PEDIDAS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Y como esta pregunta corresponde al Programa Principal, le sugerimos consulte el manual de éste, donde se le explica lo que debe hacer, ante esta pregunta.

Si el Archivo Maestro contiene información, el programa le desplegará el siguiente mensaje:

A CONTINUACION SE LE PIDEN LOS DATOS DE LOS MEDICAMENTOS,
A LOS CUALES SE LES DESEAN CARGAR UNIDADES PEDIDAS

Después de este mensaje, el programa le solicita, transmita la clave del medicamento al cual le desea cargar unidades pedidas, de la siguiente manera:

DEME CLAVE DEL MEDICAMENTO =>

Si la clave transmitida es menor que 0, el programa le desplegará el siguiente mensaje de error, y le volverá a solicitar de nuevo la clave:

ERROR : LA CLAVE O NO EXISTE

DEME CLAVE DEL MEDICAMENTO =>

Si transmite una clave mayor, a la clave máxima que contiene información dentro del Sistema, el programa le desplegará el siguiente mensaje de error y le volverá a solicitar de nuevo el dato:

ERROR : CLAVE DEL MEDICAMENTO EXCEDE EL RANGO PERMITIDO
LA CLAVE MAXIMA DEL MEDICAMENTO QUE CONTIENE INFORMACION ES : CM

Donde CM, es la clave máxima que contiene información dentro del Sistema.

Transmitida la clave del medicamento correctamente, el programa le despliega la siguiente pregunta:

CLAVE DEL MEDICAMENTO : XXXXXXXXXXXXXXXXXXXX
DESEA CARGARLE NUEVAS UNIDADES ? S)I O N)O =>

Esta pregunta se le hace con la finalidad, de que si se equivoca al transmitir la clave del medicamento al cual le desea cargar unidades pedidas, pueda corregir sin que suceda nada.

Por lo tanto si transmite una letra N, el programa le hará la siguiente pregunta:

DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O =>

Si transmite una N, el programa le desplegará los siguientes mensajes:

*** TERMINA PROGRAMA DE CARGA DE UNIDADES PEDIDAS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Como esta pregunta corresponde al Programa Principal, le sugerimos consulte el manual de éste, donde se le explica lo que debe hacer ante esta situación.

Resresemos a la siguiente pregunta:

DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O =>

Si transmite una letra S, el programa le volverá a desplegar el siguiente mensaje:

DEME CLAVE DEL MEDICAMENTO =>

Como esta pregunta ya ha sido explicada anteriormente, le sugerimos resrese a esta parte y consulte las opciones que puede transmitir.

Resresemos a la siguiente pregunta:

CLAVE DEL MEDICAMENTO : XXXXXXXXXXXXXXXX
DESEA CARGARLE NUEVAS UNIDADES ? S)I O N)O =>

Como ya anteriormente se ha explicado, lo que sucede cuando se le transmite una letra N, veamos ahora lo que ocurre, cuando se le transmite una letra S.

Cuando se transmite una letra S en esta pregunta, el programa le solicita el siguiente dato:

DEME LA CANTIDAD DE UNIDADES PEDIDAS =>

Es decir que le solicita las unidades que usted acaba de comprar de ese medicamento, para considerarlas cuando se le hayan agotado las que actualmente se encuentran en existencia. Después de haber transmitido las unidades pedidas el programa le solicitará el siguiente dato:

DEME EL PRECIO DE VENTA DEL MEDICAMENTO =>

Es decir, le solicita el precio del medicamento al cual lo va a vender.

Transmitido el precio del medicamento, el programa le solicita el siguiente dato:

DEME EL IMPUESTO DE VENTA DEL MEDICAMENTO => IM

Donde IM es el impuesto de venta del medicamento, y es un número comprendido entre el 1 y el 99, si es entero, pero si es real, el número debe ser mayor que cero o menor que uno, sino el programa le desplegará el siguiente mensaje de error y le volverá a solicitar el dato:

ERROR EL DATO ESTA EQUIVOCADO REBASA EL 100 %

DEME EL IMPUESTO DE VENTA DEL MEDICAMENTO =>

Transmitido correctamente el impuesto, el programa le solicitará lo siguiente:

DEME FECHA DE CADUCIDAD, EN ESTE ORDEN : AA MM DD
DONDE: AA ES EL AÑO, MM ES EL MES Y DD EL DIA => 85 12 31

Donde los pares de dígitos deben ir separados por un espacio, como se muestra en el ejemplo. Los datos transmitidos se analizarán primeramente por el año, el cual debe ser mayor o igual al de la fecha de las unidades en existencia actuales, si el año que transmitió no es el correcto, el programa le desplegará el siguiente mensaje de error:

ERROR : EL AÑO DEBE SER IGUAL O MAYOR, AL AÑO DE LA
FECHA DE CADUCIDAD DE LAS UNIDADES EN EXISTENCIA
EL AÑO DE LA FECHA DE CADUCIDAD DE LAS UNIDADES EN
EXISTENCIA ES : AUE

En donde AUE es el año de las unidades en existencia de la fecha de caducidad.

Luego se analiza el mes, el cual es un número que debe estar comprendido entre el 1 y el 12, sino el programa le desplegará el siguiente mensaje:

ERROR : ESTE MES MM NO EXISTE

Donde MM es el número del mes transmitido.

Y por último se analizará si el día del mes, es el correcto, si no desplegará el siguiente mensaje de error:

EL DIA DD DEL MES MM DEL AÑO AA NO EXISTE

Donde DD es el día que transmitió.

Si tuvo alguno de estos errores, en alguno de los datos de la fecha transmitida, se repetirá de nuevo este proceso y le volverá a solicitar la fecha de caducidad de las unidades que está cargando.

Transmitida la fecha correctamente, el programa le solicitará lo siguiente;

DEME CLAVE DEL PROVEEDOR PRINCIPAL DEL MEDICAMENTO =>

Solamente se le permitirá en este sistema tres proveedores que tendrán una clave que va desde la 1 hasta la número 3, y si transmite cualquier otro número el programa le desplegará el siguiente mensaje de error;

ERROR : CLAVE DE PROVEEDORES NO EXISTENTE
FAVOR DE VOLVER A DAR LA CLAVE

Transmitida la clave correctamente, el programa actualiza la información en el archivo Principal, del medicamento al cual se le cargo las unidades, e inmediatamente después el programa le vuelve a desplegar el siguiente mensaje;

DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O =>

Como esta presunta ya ha sido explicada anteriormente, le sugerimos consulte de nuevo, esta parte donde se le explica lo que debe hacer.

A partir de esta situación el proceso se repite, tantas veces como usted desee cargar más unidades a los medicamentos.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

El reporte que puede obtener si, así lo desea es simplemente el listado del proceso del programa.

4. COMENTARIOS GENERALES.

Es importante que transmita los datos correctamente, ya que estos serán utilizados, en el momento en el cual las unidades en existencia se hayan cero y será difícil poder detectar algún error en esta información.

IV.1.11 MANUAL DE OPERACION DEL MODULO LEERAN

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

RANGOS DE TEMPORADAS Y LISTA PROVEEDORES

1. PANORAMA ESPECIFICO.

OBJETIVO: Controlar el proceso para la elaboración de reportes de ventas por temporada y de datos estadísticos de precios y descuentos por proveedor.

ARCHIVOS QUE USA: Archivo de Nombres y Estadísticas, llamado NOMYEST.FIL .

RESTRICCIONES: No lista más que los dos reportes, indicados.

2. PROCEDIMIENTO:

Una vez que el programa Principal, le ha desplegado el menú de las funciones que puede ejecutar, dentro del Subsistema de Actualización y Consulta de Requerimientos, el programa le desplegará el siguiente mensaje:

DEME LA FUNCION QUE DESEA =>

Cuando esto suceda transmita la letra J, y quedará dentro del programa de Lista Rangos de Temporadas y Lista Proveedores (LEERAN).

Si el archivo Maestro no contiene información, el programa le desplegará los siguientes mensajes:

ERROR EL ARCHIVO DE NOMBRES Y ESTADISTICAS NO CONTIENE
INFORMACION, FAVOR DE VOLVER A LISTAR EL MENU Y
TRANSMITA LA OPCION <F>, PARA DAR DE ALTA ALGUN
MEDICAMENTO

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

Y como esta pregunta corresponde al Programa Principal, le sugerimos consulte el manual de éste, donde se le explica lo que debe hacer, ante esta situación.

Si el archivo de Nombres y Estadísticas contiene información, el programa le desplegará los siguientes mensajes:

CUAL LISTADO LISTADO DESEA OBTENER:

- 1.- NOMBRE Y RANGOS
- 2.- PROVEEDORES

ANOTE EL NUMERO DEL LISTADO QUE DESEA : NL

Donde NL es el número del listado que desea obtener, para este caso solo podrá transmitir el número 1 o el 2, en caso que transmita un número diferente a estos dos, el programa le desplegará los siguientes mensaje:

ANOTE UNICAMENTE 1 o 2

DESEA SACAR OTRO LISTADO S)I o N)O:

Si transmite una letra S, el programa le desplegará los mensajes ya vistos anteriormente, que son:

CUAL LISTADO DESEA OBTENER:

- 1.- NOMBRE Y RANGOS
- 2.- PROVEEDORES

ANOTE EL NUMERO DEL LISTADO QUE DESEA :

Si transmite el Número 1 el programa le genera el siguiente

reporte, que contiene los siguientes datos, que serán explicados en la parte correspondiente a la descripción del reporte.

```
=====
CLAVE . NOMBRE DEL MEDICAMENTO . RANGO 1 RANGO 2 RANGO 3 RANGO 4 FECHA DE
                                                                TRABAJO
=====
```

Al final de este reporte, el programa le despliega la siguiente pregunta:

DESEA SACAR OTRO LISTADO S)I o N)O:

Si vuelve a transmitir una letra S, el programa le desplegará los mensajes de los listados que puede obtener.

Si en la pregunta del listado que desea generar, le trasmite un 2, el programa le genera el siguiente reporte, que consta de los siguientes datos y que en la parte correspondiente a la descripción del reporte serán explicados detalladamente.

Como el listado generado es para hojas de 132 caracteres, se dividirá el encabezado de la información que se obtiene con este, de la siguiente manera:

(Primera parte del encabezado del reporte 2)

```
=====
CLAVE . NOMBRE DEL MEDICAMENTO . PROVEEDOR PRINCIPAL :
                                CLAVE PRECIO DESC.
=====
```

(Segunda parte del encabezado del reporte 2)

```
=====
: SEGUNDO PROVEEDOR : TERCER PROVEEDOR :
  CLAVE PRECIO DESC. CALVE PRECIO DESC.
=====
```

Y al final de cada uno de estos listados, el programa le despliega el siguientes mensaje visto ya anteriormente;

DESEA SACAR OTRO LISTADO S)I o N)O:

Como anteriormente ya analizamos lo que sucede cuando le transmitimos una letra S, ahora veamos que ocurre cuando se le transmite una letra N.

Cuando se transmite una letra N, en esta pregunta, el programa le despliega lo siguiente;

DESEA LISTAR EL MENU?, TRANSMITA S)I o N)O =>

Debido a que esta pregunta se sale del dominio de este manual, le sugerimos que consulte la parte correspondiente a esta pregunta que se localiza en el manual del Programa Principal.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Como este programa genera dos tipos de listados, describiremos cada uno de los campos que componen estos reportes;

Listado Núm. 1:

CLAVE	Clave del medicamento.
NOMBRE DEL MEDICAMENTO	Nombre del Medicamento.
RANGO 1	Número de unidades vendidas en la temporada 1.
RANGO 2	Número de unidades vendidas en la temporada 2.
RANGO 3	Número de unidades vendidas en la temporada 3.
RANGO 4	Número de unidades vendidas en la temporada 4.
FECHA DE TRABAJO	Fecha en la que se está consultando al Subsistema.

Listado Núm. 2:

CLAVE Clave del medicamento.

NOMBRE DEL MEDICAMENTO

Datos del Proveedor Principal:

CLAVE Clave del proveedor principal.

PRECIO Precio que da al medicamento.

DESC Descuento que nos hace del medicamento.

Datos del Segundo Proveedor:

CLAVE Clave del segundo proveedor.

PRECIO Precio que da al medicamento.

DESC Descuento que nos hace del medicamento.

Datos del Tercer Proveedor:

CLAVE Clave del tercer proveedor.

PRECIO Precio que da al medicamento.

DESC Descuento que nos hace del medicamento.

4. COMENTARIOS GENERALES.

El reporte de las ventas por temporada puede ser de gran utilidad, ya que le proporciona una muestra de como se han ido incrementando sus ventas, además el reporte de Proveedores le permite determinar que proveedor le vende más barato un medicamento.

[433]

IV.2 M A N U A L D E O P E R A C I O N
D E L S U B S I S T E M A D E
C O N T R O L D E V E N T A S

IV.2.1 MANUAL DE OPERACION DEL MODULO: PROY1

MODULO PRINCIPAL DEL SUBSISTEMA DE CONTROL DE VENTAS

1. PANORAMA ESPECIFICO.

OBJETIVO: Controlar todo el Subsistema de "CONTROL DE VENTAS".

ARCHIVOS QUE USA: Archivo Maestro, el cual contiene la información completa de todos los medicamentos; el Archivo de Nombres y Estadísticas que está ampliamente relacionado con el primero y el Archivo de Trabajo, el cual es una copia parcial del Archivo Maestro.

RESTRICCIONES: Deben estar presentes los tres archivos y contener información. El único que puede existir, aun vacío, es el de Trabajo.

Este programa puede correr en pantalla o impresora. Sin embargo, consulte la sección correspondiente para cada función para ver cuando es necesario estar trabajando en la impresora y cuando en la pantalla. Todo depende de la función que seleccione.

3. PROCEDIMIENTO.

Para entrar a este programa, simplemente coloque el disco del sistema en el drive A; en el drive B, inserte el disco que contiene los archivos.

Ahora escriba el nombre del programa PROY1 y transmita. Para transmitir, presione la tecla <RETURN>.

El sistema despliega el siguiente mensaje:

!!! BIENVENIDO A UNA NUEVA SESION DE TRABAJO !!!

EL SUBSISTEMA DE 'CONTROL DE VENTAS' REQUIERE QUE
EXISTAN EN EL DISCO B LOS SIGUIENTES ARCHIVOS:

- 1.- ARCHIVO MAESTRO
- 2.- ARCHIVO DE TRABAJO
- 3.- ARCHIVO DE NOMBRES Y ESTADISTICAS

EL SISTEMA TIENE N MEDICAMENTOS

Si no existe algún problema, como por ejemplo, que falte un archivo o que exista pero vacío, el sistema le despliega el menú principal de funciones.

Para desplegar el menú, le hace la pregunta:

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O

A una respuesta afirmativa S, aparece en la pantalla:

NOMBRE DE LA FARMACIA

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO
B	LEER EL ARCHIVO DE TRABAJO DEL DIA
C	FACTURACION PARA CLIENTES
D	CORTE DE CAJA DE LAS VENTAS DEL DIA
E	CORRECCIONES AL ARCHIVO MAESTRO
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA :

Si su respuesta fue negativa N, unicamente aparece:

FUNCION DESEADA :

Este es el momento para que Usted elija una de las funciones que aparecen en el menú, de acuerdo a su problema de manejo de información. Puede usar letras mayúsculas o minúsculas. Para entrar a cada función, el sistema valida ciertas condiciones que deben cumplirse para mantener la integridad de los datos y evitar algún problema posterior.

Sin embargo, no se preocupe, el sistema está protegido para evitar problemas con su información.

Para saber cuales son las condiciones que deben cumplirse para cada función, consulte la sección correspondiente de este manual.

Si por algún motivo llega a cometer algún error el sistema lo detecta y especifica. Cuando el error es muy grave, el sistema lo saca de sesión y podrá reiniciar hasta que cumpla con todos los requisitos para que funcione adecuadamente.

El listado de los errores es muy claro y además le indica qué hacer para corregir la anomalía.

Todos los mensajes de error están precedidos por el rótulo

* * * E R R O R * * *

A continuación aparecen los mensajes de error y como salir de ellos.

EL ARCHIVO MAESTRO NO EXISTE O ESTA VACIO

EL ARCHIVO DE TRABAJO NO EXISTE O ESTA VACIO

EL ARCHIVO DE NOM. Y EST. NO EXISTE O ESTA VACIO

En cualquier caso, el sistema da por terminada la sesión, ya que no se puede trabajar si falta algún archivo.

Usted no tiene que hacer nada para salir.

Revise el directorio del disco B; deben existir los tres archivos. De lo contrario, busque en sus discos auxiliares y coloque el adecuado.

Qué sucede si aparece el siguiente mensaje de error ?

EL ARCHIVO DE TRABAJO NO CONTIENE INFORMACION
FAVOR DE LISTAR EL MENU PRINCIPAL DE FUNCIONES
Y ELIJA LA FUNCION <A> PARA LLENARLO

Este error se presenta cuando quiere trabajar con el subsistema, pero el archivo de trabajo está vacío. En este caso, simplemente elija la opción <A> y llénelo.

El sistema solo acepta las funciones descritas en el menú. Qué sucede si por error, anota una letra que no corresponde a alguna función del menú ?

Por ejemplo, si Usted anota la letra M, la cual no está asociada a una función, el programa le responde así:

UNICAMENTE SE ACEPTA A, B, C, D, E, F o Q

Significa que la función anotada, no existe.

Qué sucede si pretende usar archivos Maestro y el Nombres y Estadísticas que contienen diferente número de medicamentos ?

El mensaje de error es el siguiente:

ARCHIVO MAESTRO TIENE (clave máxima) REGISTROS
ARCHIVO DE NOM. Y EST. TIENE (clave máxima) REGISTROS
ESTA USANDO ARCHIVOS EQUIVOCADOS

Revise en sus discos y coloque los archivos adecuados.

El siguiente mensaje de error es similar al anterior, solo que en este caso intervienen los archivos Maestro y el de Trabajo.

ARCHIVO MAESTRO TIENE (clave máxima) REGISTROS
ARCHIVO DE TRABAJO TIENE (clave máxima) REGISTROS
ESTA USANDO UN ARCHIVO DE TRABAJO EQUIVOCADO

Revise en sus discos y coloque los archivos adecuados.

Quando quiera dar por terminada la sesión, a la pregunta:

FUNCION DESEADA :

responda simplemente, escribiendo una letra Q y transmita.

Eso es todo, ahora saque el mayor provecho posible a este sistema. Pero recuerde, primero debe conocerlo, para poder usarlo.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

No es un reporte, más bien es un diálogo que se establece entre Usted y el sistema. Los desplegados consisten en el menú principal de funciones, listado de errores y alternativas de solución.

Por otra parte, en este programa, se verifican las lecturas y escrituras que se hacen a los archivos. Existen también mensajes en caso de que se presente algún error al momento de trabajar con los archivos.

Entre los mensajes de error que aparecen, están los siguientes:

LEYENDO DONDE NO HAY DATOS

Este error se presenta cuando se pretende leer un registro que no existe o que no contiene información.

ERROR DE CP/M

Cuando se presenta un error en el sistema operativo de su computadora.

LLEGO AL FINAL DEL DISCO

Todo tiene su límite, un disco también.

ERROR DE CODIGO DESCONOCIDO

Cuando se presenta un error no identificado por el programa. No se preocupe, el sistema está protegido para evitar este tipo de errores.

Sin embargo tenga cuidado y consulte el manual en caso de dudas.

4. COMENTARIOS GENERALES.

El sistema esta diseñado de tal manera que lo va guiando paso a paso, durante las sesiones de trabajo. Además, valida sus respuestas y en algunos casos, pide la ratificación de las mismas, por lo que resulta muy remoto que se equivoque al trabajar con su computadora.

Recuerde, primero aprenda a manejarlo, para poder obtener el máximo beneficio.

IV.2.2 MANUAL DE OPERACION DEL MODULO DUPLICAR.

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

DUPLICAR EL ARCHIVO MAESTRO
PARA OBTENER EL ARCHIVO DE TRABAJO.

1. PANORAMA ESPECIFICO.

OBJETIVO: Obtener una copia parcial del Archivo Maestro, llamado Archivo de Trabajo, grabado en el disco B con el nombre TRABAJO.FIL . Este nuevo archivo se usa en el Subsistema "CONTROL DE VENTAS", sin el cual, resulta imposible hacer facturas, obtener cortes de caja, etc. La copia de trabajo debe ser creada para cada nueva sesión de ventas. Esto es muy importante. Llamamos sesión de ventas al periodo de tiempo que Usted va a usar el Archivo de Trabajo para elaborar facturas. Por seguridad de su sistema, debe crear un archivo de trabajo para cada día, al iniciar el mismo; el sistema verifica que no exista un TRABAJO.FIL cargado con información. Por lo tanto, esta función es la PRIMERA que Usted debe usar al inicio de sus actividades diarias.

ARCHIVOS QUE USA: En el disco B deben existir los siguientes archivos: Archivo Maestro, llamado MAESTRO.FIL y debe estar "creado" el Archivo de Trabajo, con el nombre TRABAJO.FIL.

RESTRICCIONES: En caso de que falte algo de lo mencionado anteriormente, el Sistema lo detecta, sin permitir el acceso a cualquier función y da por terminada la sesión. Si desea reanudar su trabajo, debe proporcionar inmediatamente TODAS las condiciones necesarias para que funcione este programa.

Esta función puede usarse tanto en impresora como en pantalla de televideo. Se recomienda usar la pantalla.

2. PROCEDIMIENTO.

Con esta función se inician las labores de una nueva sesión de trabajo.

El sistema opera en base a preguntas que hace al usuario. Cuando anote una respuesta completa, transmita por medio de la tecla <RETURN>.

Para entrar a esta opción, debe elegir del MENU PRINCIPAL DE FUNCIONES, la función "DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO", por medio de la opción <A>, de la siguiente manera:

Después de listar el MENU, aparece el siguiente mensaje del Sistema:

FUNCION DESEADA:

Anote el caracter A.

En este momento entra Usted a la función y se despliega el mensaje:

EL SISTEMA INICIA EL PROCESO PARA OBTENER EL
ARCHIVO DE TRABAJO A PARTIR DEL ARCHIVO MAESTRO

El Sistema le pide la fecha así:

ANOTE LA FECHA DE HOY :

A N O =>

M E S =>

D I A =>

Usted debe escribir los datos en el orden requerido, transmitiendo (presionando la tecla <RETURN>) después de escribir el dato.

Cómo se anotan los datos de la fecha ?

Los datos de la fecha se anotan de la siguiente manera:

A N O => Anote las dos últimas cifras correspondientes al año: 85, 88, etc.

M E S => Escriba el número correspondiente a cada mes (ver TABLA DE MESES).

D I A => Tiene un valor de 1 a 31, según sea el mes.

TABLA DE MESES	
MES	EQUIVALENTE NUMERICO

ENERO	1
FEBRERO	2
MARZO	3
ABRIL	4
MAYO	5
JUNIO	6
JULIO	7
AGOSTO	8
SEPTIEMBRE	9
OCTUBRE	10
NOVIEMBRE	11
DICIEMBRE	12

El Sistema pregunta:

ESTA CORRECTA LA FECHA ?. ANOTE SI O NO :

En caso de que haya cometido un error, simplemente anote N y se pide nuevamente el dato de la fecha.

El Sistema saldrá de esta sección, cuando usted conteste afirmativamente a la pregunta ESTA CORRECTA LA FECHA ?.

Sin embargo, tomando en cuenta que la fecha es un dato muy importante, el sistema vuelve a preguntar:

REALMENTE ESTA CORRECTA LA FECHA ? ANOTE SI O NO

Esta es la última oportunidad que tiene para aceptar la validez de la fecha, aun cuando para el sistema ya sea válida, pero él no puede consultar el calendario como lo haría cualquiera de nosotros.

El Sistema tomará la fecha como válida, cuando usted conteste afirmativamente a la última pregunta.

Si su respuesta es negativa, se la pedirá nuevamente. Este ciclo se deja de ejecutar cuando Usted acepta una fecha como verdadera.

Usted es el UNICO responsable de la validez de la fecha. Recuerde que este dato, aparecerá en las facturas del día, en el corte de caja del mismo y en todos los documentos que maneja el Archivo de Trabajo.

Por lo tanto, lo invito para que tenga especial cuidado cuando anote el dato de la fecha.

En caso extremo de que haya aceptado una fecha como válida, sin serla, la UNICA forma de corregir este error garrafal, es eliminado la copia totalmente, volver a crearla y entrar nuevamente a esta función.

Para eliminar totalmente al Archivo de Trabajo, consulte la sección correspondiente.

Recuerde, NO HAY FORMA DE CORREGIR UNA FECHA EQUIVOCADA. Tenga mucho cuidado.

Si Usted está de acuerdo con la fecha anotada, es decir que la acepta como válida, entonces el Sistema continua con el proceso para obtener el Archivo de Trabajo, desplegando el siguiente mensaje:

SE ESTA DUPLICANDO EL ARCHIVO.

ESPERE LA SENAL PARA PODER CONTINUAR ...

Una vez que el Sistema terminó, su pantalla despliega:

EL SISTEMA TERMINO DE DUPLICAR, PUEDE CONTINUAR

Mientras no aparezca este mensaje, Usted NO PUEDE HACER NADA. Debe esperar forzosamente a que el Sistema termine de duplicar el archivo.

No intente hacer algo, porque lo único que logrará será que el Sistema no termine su trabajo y tenga que volver a empezar DESDE EL PRINCIPIO, aunque suene redundante; es por la seguridad del sistema.

El tiempo que tarda en hacer la copia es relativamente pequeño. Sea razonable, entre más grande sea el archivo, más tiempo tardará en obtener la copia.

No se desespere, dele tiempo al sistema, porque además de hacer la copia, la verifica, por lo que Usted estará plenamente seguro de que la información contenida en la copia es un duplicado fiel del original.

Qué sucede si Usted Pretende obtener un archivo que ya contiene información ?

Este caso se presenta cuando selecciona la función <A>, pero el Archivo de Trabajo ya contiene información. El sistema está diseñado para no permitir encimar información.

Si Usted pretende copiar datos a un archivo de trabajo que ya contiene información por medio de la función <A>, el sistema despliega el siguiente mensaje:

* * * E R R O R * * *

EL ARCHIVO DE TRABAJO YA CONTIENE INFORMACION.

POR LO TANTO NO PUEDE USAR ESTA FUNCION.

No tiene nada que hacer, el sistema lo detecta y no le permite entrar a esta opción, regresando el control al MENU PRINCIPAL DE FUNCIONES.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

El Sistema no produce un reporte de salida. Los únicos mensajes que aparecen son aquellos mediante los cuales, se establece el diálogo entre Usted y el Sistema.

4. COMENTARIOS GENERALES.

Recuerde que es necesario crear y llenar el archivo de trabajo para cada sesión de ventas, con el fin de tener siempre la información actualizada.

El sistema acepta letras mayúsculas o minúsculas al momento de establecer el diálogo con Usted.

IV.2.3 MANUAL DE OPERACION DEL MODULO LEECOPIA.

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

LEER EL ARCHIVO DE TRABAJO DEL DIA.

1. PANORAMA ESPECIFICO.

OBJETIVO: Leer cualquier registro del Archivo de Trabajo, durante la sesión del Subsistema "CONTROL DE VENTAS".

ARCHIVOS QUE USA: Es necesario que esté presente, con información, el Archivo de Trabajo TRABAJO.FIL.

RESTRICCIONES: La única restricción para que funcione el módulo LEECOPIA, es la presencia del Archivo de Trabajo y que contenga información. La lectura se puede hacer en la pantalla o con la impresora.

2. PROCEDIMIENTO.

El sistema trabaja en base a preguntas; por lo tanto cuando Usted anote su respuesta, transmita presionando la tecla <RETURN>.

Para entrar a este módulo, debe elegir del MENU PRINCIPAL DE FUNCIONES, la función "LEER EL ARCHIVO DE TRABAJO DEL DIA", por medio de la opción .

El Sistema inicia el trabajo, desplegando la fecha del archivo de la siguiente manera:

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA:

ANO : AA MES : MM DIA : DD

Donde AA es el valor del año, MM el valor del mes y DD el valor del día; cada uno en dos dígitos.

Inmediatamente después de que aparece la fecha del archivo de trabajo, el sistema le indica lo siguiente:

COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA

PARA CONTINUAR PRESIONE <RETURN>

Si está Usted trabajando en la pantalla, simplemente presione <RETURN> y aparecerá el encabezado del reporte, mostrado en la Fig. IV.2.3.1.

Por otra parte si está usando la impresora, corra el papel hasta el inicio de una nueva hoja, y presione la tecla <RETURN>. Hecho lo anterior, se inicia el reporte con el encabezado del mismo. Consulte la Fig. IV.2.3.1.

El sistema despliega el encabezado. Al terminar, inicia una nueva línea y escribe el siguiente mensaje:

?

Es la señal que da el sistema, para que Usted anote la clave del medicamento, cuya información desea que aparezca listada en el siguiente renglón.

Escriba la clave y transmita.

Los datos, aparecerán en el orden descrito en el encabezado, para que Usted haga las consultas necesarias.

Si anota uno o varios caracteres diferentes de número, por ejemplo letras, signos, etc., el sistema se queda esperando hasta que escriba Usted un número. Es decir únicamente reconoce números. No anote otra cosa que no sean números.

Al terminar de listar la información del medicamento solicitado por Usted, el cursor de la pantalla se posiciona al principio del siguiente renglón, esperando lo siguiente:

Si desea continuar leyendo otro medicamento, simplemente presione la tecla <RETURN>, sin hacer anotaciones. Si anota cualquier carácter el sistema no lo reconoce, a excepción de la letra <Q>.

Una vez hecho esto, el sistema entiende que Usted desea continuar y está listo para recibir la clave del medicamento que quiere consultar, desplegando en la pantalla:

?

esperando que Usted anote la clave de otro medicamento.

Qué sucede si Usted anota la clave de un medicamento, el cual no existe en el Archivo de Trabajo ? El sistema marca lo siguiente:

NO EXISTE, EL REGISTRO n ESTA FUERA DEL RANGO (CLAVE MAX)

Qué significa lo anterior ?

El Archivo de Trabajo está integrado por cierto número de registros. Existe un registro por cada medicamento que Usted haya grabado en el Sistema.

Por lo tanto, si en el Archivo Maestro se grabaron los datos de 5000 medicamentos, es decir que la clave máxima corresponde al medicamento 5000, el sistema solo aceptará como válidas las claves comprendidas en el rango de 1 a 5000.

Por lo anterior, si Usted, por error, solicita la información del medicamento 6000, anotando esta clave, entonces el sistema despliega el mensaje de error diciendo que **NO EXISTE**, ya que el **REGISTRO 6000 ESTA FUERA DEL RANGO permitido (de 1 a 5000, para este ejemplo)**.

Aun cuando el sistema realiza la verificación anterior, procure no cometer errores, ni trate de Jugar con su computadora.

Para salir de esta función, cuando aparezca el cursor al inicio de una nueva línea, anote el carácter <Q> y dará por concluidas las actividades de este módulo, regresando al **MENU PRINCIPAL DE FUNCIONES**.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Por medio de un diálogo entre el usuario y el Sistema, se realiza el intercambio de instrucciones para el funcionamiento del módulo.

3.1 SECCION DEL ENCABEZADO DE REPORTE.

```
(NOMBRE DE LA FARMACIA)          ARCHIVO DE TRABAJO          FECHA : AÑO/MES/DÍA

=====
CLAVE: P R E C I O : I M P :   U N I D A D E S
      : D E U E N T A : Z :   E N       : V E N D I D A S : P E D I D A S : V E N D I D A S
      : S I N I M P   C O N I M P :   : E X I S T E N C I A : ( A C T ) : ( U L T R E M ) : D E L P E D .
=====
```

FIG. IV.2.3.1. ENCABEZADO DE LA FUNCION "LEER EL ARCHIVO DE TRABAJO DEL DIA".

Primero describimos los datos de identificación de la empresa.

NOMBRE DE LA FARMACIA. Es el nombre de la empresa, contenido en 30 caracteres.

ARCHIVO DE TRABAJO. Identificación del reporte.

FECHA. Escribe el año, mes y día, separados por una diagonal.

A continuación aparece la descripción de cada una de las columnas que componen al encabezado:

CLAVE Es la clave de identificación de cada medicamento. Aparece en un campo de cinco dígitos.

PRECIO DE VENTA Contiene el precio de venta del medicamento. SIN IMP anota el precio sin impuesto. CON IMP escribe el precio incluyendo el impuesto. Tiene nueve dígitos.

IMP % Es el porcentaje de impuesto que debe pasar el cliente, al hacer la compra. Tiene tres dígitos.

UNIDADES EN EXISTENCIA Proporciona el número de unidades en existencia actuales. Maneja siete dígitos.

UNIDADES VENDIDAS (ACT) Proporciona la información relacionada con el número de unidades vendidas de las existencias actuales. Usa un campo de siete dígitos.

UNIDADES PEDIDAS (ULT REM) Indica cuantas unidades se pidieron al proveedor en la última remesa ya surtida. Tiene siete dígitos.

UNIDADES VENDIDAS DEL PED. En esta columna aparecen las unidades que ya se vendieron del pedido. Maneja siete dígitos.

El listado de la información puede hacerse en la pantalla o en la impresora. Para mayor claridad se sugiere que los listados los realicen en la impresora. Si lo que busca es rapidez, entonces use la pantalla.

Para su comodidad y con el fin de tener una referencia clara, el encabezado del reporte aparece después de listar 13 medicamentos. Los errores también cuentan.

4. COMENTARIOS GENERALES.

Esta función es útil para hacer consultas rápidas, por lo que debe hacerse en pantalla. Sin embargo, si la microcomputadora está conectada a la impresora también lo puede hacer.

Puede usar papel desde tamaño carta u otro más grande.

IV.2.4 MANUAL DE OPERACION DEL MODULO FACTURAR

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:
FACTURACION PARA CLIENTES.

1. PANORAMA ESPECIFICO.

OBJETIVO: Obtener la factura de cada cliente.

ARCHIVOS QUE USA: Archivo de Trabajo, llamado TRABAJO.FIL.

RESTRICCIONES: Es condición necesaria que exista el archivo TRABAJO.FIL con información cargada. Este módulo debe ser procesado usando la impresora , ya que por medio de ella, se elaboran las facturas por escrito. La impresora debe tener suficiente papel tipo STOCK (tamaño carta).

2. PROCEDIMIENTO.

Existen datos que aparecen impresos en la sección del usuario y/o en la sección del cliente. Para que Usted tenga una referencia precisa, en el presente capítulo del manual, es necesario que tenga a la mano un ejemplo de elaboración de facturas.

Para que el sistema tome en cuenta las respuestas o los datos que le demos, después de anotar cada uno de ellos, ya sea en forma de dato o conjunto de datos, debe PRESIONAR la tecla <RETURN> siempre, para transmitir.

Antes de entrar a esta función, cerciórese que la impresora tenga suficiente papel para elaborar las facturas de toda la sesión de ventas.

2.1 ACCESO A LA FUNCION FACTURAR.

La primera parte sirve para entrar al módulo FACTURAR. Presenta a Usted el Archivo de Trabajo que se usará durante el día. Usted es el UNICO responsable para aceptar o rechazar el archivo.

A continuación aparecen los diálogos que se establecen entre Usted y su Sistema, correspondientes a la Primera Parte.

Para tener acceso a este módulo, seleccionamos la opción <C> "FACTURACION PARA CLIENTES" que aparece en el MENU PRINCIPAL DE FUNCIONES.

Una vez hecho esto, el sistema nos despliega la fecha del Archivo de Trabajo, con el mensaje:

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :

ANO : AA, MES : MM, DIA: DD

Donde : AA puede tener un valor de 86 a 99

MM puede tener un valor de 1 a 12

DD puede tener un valor de 1 a 31

Despliega la siguiente pregunta:

LE SIRVE ? S)I o N)O :

Si el Archivo de Trabajo NO es el adecuado, ya sea porque es de fecha atrasada o simplemente porque no se ajusta a nuestras necesidades, tenemos la opción de salir del módulo FACTURAR anotando <N>.

Si escogimos esta opción <N>, debemos eliminar el Archivo TRABAJO.FIL y obtener uno nuevo. De este modo podremos regresar a FACTURAR, cuando sea conveniente.

Si por el contrario, el Archivo TRABAJO.FIL es útil para nosotros entonces, contestaremos con S, para poder continuar.

El sistema nos hace otra pregunta :

DESEA CONTINUAR ? S)I O N)O :

La respuesta depende únicamente de Usted. Solo puede responder con S (en caso afirmativo) o con N (en caso negativo).

El sistema continuará con el módulo FACTURAR si y solo si las dos respuestas a la preguntas anteriores fueron 'S' (afirmativas). En caso contrario, se sale del módulo y despliega el MENU PRINCIPAL DE FUNCIONES.

Una vez que fue aceptado el Archivo TRABAJO.FIL, el sistema le hace la siguiente indicación:

COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA

PRESIONE <RETURN>

Cuando el papel esté colocado en el lugar preciso, simplemente presione la tecla <RETURN> para poder continuar.

En este momento se inicia la elaboración de la factura.

CLAVE :

Si su respuesta fue afirmativa, entonces el sistema le pide la clave de la medicina similar que el cliente haya seleccionado.

ERROR XXXX NO ES SIMILAR
ANOTE NUEVAMENTE

Si por algún motivo anotó alguna clave que no corresponde a las similares listadas, entonces el sistema detecta el error, indicándole que la clave que usted dió no es la de una medicina similar, solicitándole nuevamente el dato de la clave.

NO LA TENEMOS
QUIERE OTRA SIMILAR? S)I o N)O :

Es remota la posibilidad, pero puede presentarse el caso que, de la medicina similar seleccionada no se tengan existencias. El cliente elige si desea otra similar o no. En caso de que su respuesta sea afirmativa entonces, el sistema le pide la CLAVE. Para una respuesta negativa N, sale de la sección de medicinas similares y continúa, solicitando nuevos datos.

CUANTAS :

Después que anotó la clave seleccionada y transmitió, el sistema le pide el número de unidades que desea de la medicina similar demandada. Anote la cantidad y transmita.

SOLO TENGO : NNNN
SE LAS LLEVA S)I o N)O :

Si el cliente demanda de la medicina 999 la cantidad de 12, pero únicamente se tiene en existencia 10 unidades, el sistema despliega el mensaje anterior y espera la respuesta.

En este punto el sistema despliega los siguientes aspectos, SIN anotar los encabezados:

\$ TOTAL ACUMULADO	CLAVE DE LA MEDICINA	\$ PRECIO UNITARIO (con imp)	CANTIDAD DEMANDADA	TOTAL POR MEDICINA :
--------------------	----------------------	------------------------------	--------------------	----------------------

\$ TOTAL ACUMULADO. Es el total en dinero que hasta ese momento lleva acumulado el cliente. Incluye los datos de la medicina que se está vendiendo. Este dato es muy importante, ya que sirve como referencia, para que el cliente pueda continuar solicitando otros medicamentos de acuerdo con el dinero que lleve para poder pasar. Maneja cifras de doce dígitos con dos decimales. Incluye impuesto. Este dato aparece en la sección del usuario.

CLAVE DE LA MEDICINA. Corresponde a la clave de la medicina, por medio de la cual la reconoce el sistema. Es el dato correspondiente a CLA o CLAVE.

\$ PRECIO UNITARIO. Es el precio unitario que tiene cada medicamento. Maneja cifras de diez dígitos con dos decimales. Incluye el impuesto.

CANTIDAD DEMANDADA. Corresponde a las unidades que el cliente solicitó por medio de la presunta CAN o CUANTAS.

\$ TOTAL POR MEDICINA. Es el resultado de multiplicar el precio unitario por la cantidad demandada. Este dato ya incluye el impuesto. Maneja cantidades de doce dígitos con dos decimales.

Al final de esta línea de datos desplegados por el sistema, aparecen dos puntos(:). Esta señal indica que es el momento de validar la venta que aparece listada. Qué significa esto?

En este instante, se puede cancelar la venta del medicamento cuyos datos aparecieron listados. Esto puede suceder debido a que el cliente se arrepienta de la compra por cualquier motivo. Tome en cuenta que aun cuando el cliente se haya arrepentido de la compra, en la factura aparecerán anotados los datos de esta medicina, pero serán ignorados al momento de obtener el TOTAL de la venta. Así que no se preocupe y continúe elaborando su factura.

Si el cliente acepta el producto entonces debe Usted anotar un asterisco < * > inmediatamente después de los dos puntos (:). De esta manera, el proceso de facturación continuará, tomando en cuenta la venta de este medicamento. En caso contrario, simplemente con que anote CUALQUIER caracter diferente del asterisco, el sistema ANULA este movimiento permitiéndole continuar.

NNNNNNNNNNNN.NN

Después de cada medicina vendida, el sistema le mostrará el TOTAL ACUMULADO que lleva hasta ese momento, para que el cliente pueda tener una referencia de lo gastado y poder continuar o decir que ya no desea otra cosa más.

T)OT O RET?

Inmediatamente después, incluso en el mismo renglón, el sistema le pregunta si desea continuar u obtener el total de la factura. Las opciones son T)OT O RET . Si desea dar por terminada la factura simplemente anote T. Por el contrario si desea agregar más productos a la factura simplemente presione la tecla <RETURN>.

Así, el sistema le pide CLA Y CAN (clave y cantidad). Tenga presente que esto lo hará para cada nueva medicina que el cliente vaya pidiendo.

Si el cliente ya no quiere otra medicina, entonces en cuanto el sistema le muestre T)OT O RET , elija la opción T. Tome en cuenta que si elija esta opción, ya no podrá regresar a facturar un nuevo medicamento contenido en el sistema (al menos en la factura presente o actual).

T)OT o A)DIC ?

Después de elegir T, se dá por terminada la facturación de medicinas contenidas en el sistema, pero la función le permite continuar haciendo la facturación de otros productos adicionales para el mismo cliente, en la misma factura. Es decir, el sistema le pregunta T)OT o A)DIC.

Si elige la opción A, podrá continuar facturando productos adicionales que están fuera del sistema. Estos productos pueden ser cigarrillos, dulces, perfumes o cualquier cosa que el cliente quiere y que la farmacia lo tenga.

PREC UNID IMP

Una vez elegida A (Producto Adicional), entonces el sistema le pide PREC (Precio), UNID (cantidad) e IMP (impuesto), todo en el mismo renglón. Usted deberá anotar los tres datos, separándolos con por lo menos un espacio y SIN OMITIR alguno. En caso de que un dato sea igual a cero, como el impuesto por ejemplo, de todas maneras deberá anotarlos. El dato del impuesto lo puede anotar como suste; en forma entera (20) o fraccionaria (0.20); el sistema lo acepta y procesa.

B)IEN O M)AL ?

Aquí también tiene la oportunidad de que, si cometió algún error, lo pueda corregir, ya que el sistema le pregunta si los datos que dió están correctos o no: B)IEN O M)AL. En caso de que estén mal, simplemente anote M y la operación será ignorada. Si la acepta anote B y el sistema continuará.

En este punto el sistema despliega los siguientes aspectos, SIN anotar los encabezados:

\$ TOTAL	'EX'	\$ PRECIO	CANTIDAD	TOTAL POR
ACUMULADO		UNITARIO	DEMANDADA	PRODUCTO :

\$ TOTAL ACUMULADO. Es el total en dinero que hasta ese momento lleva acumulado el cliente. Este dato es muy importante, ya que sirve como referencia para que el cliente pueda continuar solicitando productos o artículos adicionales de acuerdo con el dinero que lleve para pasar. Maneja cifras de doce dígitos con dos decimales. Incluye impuesto.

EX. Esta palabra significa que el producto que aparece listado corresponde a una venta adicional.

\$ PRECIO UNITARIO. Es el precio unitario que tiene el producto adicional. Corresponde a PREC, anotado por Usted. Maneja cifras de diez dígitos con dos decimales.

CANTIDAD DEMANDADA. Corresponde a las unidades que el cliente solicitó por medio de la presunta UNID.

\$ TOTAL POR PRODUCTO. Es el resultado de multiplicar el precio unitario por la cantidad demandada. Este dato ya incluye el impuesto.

El final de esta línea de datos desplegados por el sistema, aparecen dos puntos ({}). Esta señal indica que es el momento de validar la venta que aparece listada. Qué significa esto?

Significa lo mismo que en el punto anterior de las medicinas, solo que en este caso se aplica a los productos adicionales.

Recuerde, si acepta la compra el cliente, entonces solo anote un asterisco (*) después de los dos puntos ({}), y continúe. En caso contrario anote cualquier carácter diferente del asterisco.

A)DIC o T)OT ?

Después de procesar cada producto adicional, el sistema le hace la presunta : A)DIC o T)OTAL Si desea seguir facturando productos adicionales, anote el carácter A. Que quede claro, aquí solamente podrá seguir procesando productos adicionales en esta factura.

Para obtener el total general o global, tanto de las medicinas contenidas en el sistema, así como de los productos adicionales, simplemente anote T en cuanto presunte A)DIC o T)OT.

Aparece el desplegado siguiente:

\$ TOTAL
ACUMULADO

* T O T A L * *

\$ TOTAL
ACUMULADO

\$ TOTAL ACUMULADO. Corresponde a la venta total. Es la suma de la venta de medicamentos más la venta de productos adicionales. Maneja doce dígitos y dos decimales. Aparece en la sección del usuario.

* T O T A L * * . Es el mensaje que corresponde al total final.

\$ TOTAL ACUMULADO. Es igual al inciso del usuario.

DINERO

El sistema solicita el DINERO que el cliente pasa en billete o monedas por la venta realizada. Anote Usted la cantidad que pasa el cliente.

B)IEN O M)AL ?

Se confirma el dato con B o M. El sistema verifica que el dinero pasado por el cliente sea mayor o igual al TOTAL a pasar. Si no se cumple lo anterior, entonces el sistema marca error y solicita nuevamente el dato del DINERO. Recuerde, si al momento de dar el dato del DINERO comete algún error, no se preocupe, ya que el sistema además de pedir la confirmación de los datos, los verifica. Sin embargo, procure no cometer errores.

ERROR ANOTE NUEVAMENTE
DINERO

En caso de que haya aceptado una cantidad de dinero como correcta, pero si en realidad está equivocada, el sistema despliega el mensaje:

Con el dato del DINERO, se calcula el CAMBIO y aparece el siguiente desplegado:

Finalmente, aparece el cierre de la factura:

DINERO	"D I N E R O \$ "	NNNNNNNN.NN
TOTAL	"T O T A L \$ "	NNNNNNNN.NN
CAMBIO	"C A M B I O \$ "	NNNNNNNN.NN

"GRACIAS POR SU COMPRA"

En este momento se cierra la factura actual.

DESEA OTRA FACTURA ? El sistema le pregunta si desea obtener otra factura. Si su respuesta es afirmativa, el sistema procederá a elaborar una nueva, tal y como se explicó anteriormente.

Si Usted ya no desea elaborar otra factura y trata de salir de esta función entonces, cuando el sistema le presente la pregunta anterior, simplemente conteste N.

De este modo, termina con el módulo FACTURAR y el sistema transfiere el control a otra sección, para que Usted elija otra función del MENU PRINCIPAL DE FUNCIONES.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

Como se vió en el PROCEDIMIENTO, la primera parte consiste del diálogo entre el sistema y Usted, para aceptar el archivo de trabajo.

La segunda parte consiste de la elaboración de las facturas.

La factura está compuesta por dos secciones:

s e c c i ó n		s e c c i ó n
d e l		d e l
u s u a r i o		c l i e n t e

3.1 SECCION PARA EL USUARIO.

Esta sección es muy importante para usted, ya que tiene escritos todos los datos de las ventas. Esto le sirve para hacer una aclaración posterior y en el último de los casos, para hacer correcciones al archivo por medio de la función <E>.

Está formada por los siguientes aspectos:

- FACT. Número de la factura.
- CLA. Clave de la medicina.
- CAN. Cantidad a ser vendida.
- B)IEN o M)AL. Ratificación de los datos.
- Medicinas similares.
- Total acumulado de la venta.
- T)OT o RET. Total o <RETURN>.
- T)OT o A)DIC. Total o productos adicionales.
- Dinero que el cliente pasa, para obtener el "cambio".
- Cambio para el cliente.
- Termina o continua elaborando facturas.

3.2 SECCION PARA EL CLIENTE.

Está compuesta por los siguientes aspectos:

- FACTURA NUM. Número de la factura.
- AA/MM/DD. Fecha.
- Clave de la medicina.
- Precio unitario (incluye impuesto).
- Cantidad de medicinas.
- Subtotal (incluye el impuesto).
- Opción para aceptar o rechazar el producto contenido en el sistema.
- Opción para aceptar o rechazar el producto adicional o fuera del sistema. Sucede lo mismo que en el caso anterior.
- Total de la venta.
- Dinero pagado.
- Cambio.

4. COMENTARIOS GENERALES.

No es necesario que una sesión de ventas sea continua. Esto significa que, en cualquier momento puede salir de esta función, contestando N a la pregunta DESEA OTRA FACTURA ?. Esta suspensión puede ser por diversos motivos personales. En el momento que Usted suelte, puede volver a entrar a la función con el mismo archivo de trabajo. Recuerde que al iniciar esta opción se le muestra la fecha del archivo y se le pregunta si le sirve y si desea continuar. Contesté afirmativamente y reanude su sesión de ventas.

Quando ya no podrá usar un archivo de trabajo anterior ?

Quando ese archivo haya servido para resresar información al archivo maestro, es decir, cuando haya sido usado para la función <F>. En ese momento, ese archivo, queda inutilizado para hacer facturas, aun cuando con él se pueden seguir obteniendo Cortes de Caja de ventas del día.

Por lo tanto tenga mucho cuidado con sus archivos, y con lo que hace con ellos, ya que guardan información muy valiosa no solo para el sistema sino también para Usted.

Otro punto muy importante que no debe perder de vista es el hecho de que, en caso de cometer algún error durante el proceso de facturación, Usted tiene la posibilidad de hacer correcciones al archivo de trabajo.

Cómo se realiza esta actividad ?

El sistema está protegido contra errores humanos, pero recuerde que cuando Usted da algún dato, el sistema pide su confirmación y en algunos casos hasta la ratificación correspondiente. Suponiendo, que aun con los dos pasos anteriores, confirmación y ratificación de respuestas, Usted comete un error, puede hacer las correcciones por medio de la función <E> "CORRECCIONES AL ARCHIVO DE TRABAJO".

Qué información necesita para hacer las correcciones y de dónde la toma ?

La factura consta de dos secciones: una para el usuario y otra para el cliente. Si cometió algún error, anótelos en su sección y cuando haga las correcciones, simplemente consulte esa sección y proceda a corregir. Para mayor información consulte el manual del módulo de correcciones.

Qué hacer en caso de falla de energía eléctrica ?

Si no hay energía, el sistema no trabaja. Sin embargo, la farmacia debe seguir funcionando. Continúe haciendo sus facturas, anotando las ventas realizadas, para posteriormente, introducir las al sistema por medio de la función <E> "CORRECCIONES AL ARCHIVO DE TRABAJO".

Su cuadro de anotaciones finales, debe contener:

CLAVE DEL MEDICAMENTO UNIDADES VENDIDAS

IV.2.5 MANUAL DE OPERACION DEL MODULO CORTE.

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

CORTE DE CAJA DE LAS VENTAS DEL DIA.

1. PANORAMA ESPECIFICO.

OBJETIVO: Hacer el reporte del Corte de Caja de las ventas del día.

ARCHIVOS QUE USA: Requiere que exista el Archivo de Trabajo del día, TRABAJO.FIL en el disco B, cargado con la información de las ventas del día. Si Usted pretende obtener el corte de un archivo que no existe, el sistema marcará el error.

RESTRICCIONES: El corte de caja se hará con el Archivo de Trabajo que se tenga presente, es decir, que si deseamos obtener un corte del día X, debemos cargar el archivo del día X. Se sugiere que para mayor seguridad, una vez que haya terminado de hacer facturas, obtenga una copia de ese archivo. Consulte la sección correspondiente.

Para que esta función genere el reporte, es necesario que a la microcomputadora esté conectada la impresora y que tenga suficiente papel tamaño grande, de 132 caracteres por línea. Sin embargo, el reporte también se puede obtener por pantalla, pero con la limitación en el aspecto claridad.

2. PROCEDIMIENTO.

Para entrar a este módulo, debe elegir la opción <D>, "CORTE DE CAJA DE LAS VENTAS DEL DIA", del menú principal de funciones.

El sistema despliega el siguiente comentario:

COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA
PRESIONE <RETURN>

Siga las instrucciones al pie de la letra. Primero coloque el papel al inicio de una nueva hoja y presione <RETURN>. A partir de este momento Usted no tiene que hacer nada, el sistema lo hace todo, es decir se genera el reporte descrito en la siguiente sección.

Es automático el regreso al menú principal de funciones.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

SECCIONES DEL REPORTE.

(PRIMERA PARTE)

NOMBRE DE LA FARMACIA

=====

M E D I C I N A	UNIDADES	PRECIO	SUBTOTAL
CLAVE N O M B R E	VENDIDAS	DE	\$
		VENTA	

=====

(SEGUNDA PARTE)

FECHA:

=====

I V A	T O T A L	PRECIO	PREC.COMP	GANANCIA
\$	\$	DE	X UNID	N E T A
		COMPRA	VENDIDAS	\$

=====

FIG. IV.2.5.1 ENCABEZADO DEL REPORTE DEL CORTE DE CAJA.

El reporte está compuesto de 5 secciones.

Consulte la figura IV.2.5.1, en la cual se muestra un ejemplo de la forma del encabezado del reporte para el Corte de Caja del Día.

Una vez seleccionada la opción <D>, el sistema despliega las siguientes secciones:

3.1 SECCION DE IDENTIFICACION.

En esta sección aparece el nombre de la farmacia en un campo de treinta caracteres y la fecha de elaboración del corte de caja; todo en un solo renglón.

3.2 SECCION DE ENCABEZADO DE COLUMNAS.

El reporte incluye las siguientes columnas:

Todos los campos que corresponden a dinero, incluyen dos decimales. Este comentario es válido en todas las secciones.

M E D I C I N A. Se subdivide en dos aspectos: CLAVE Y NOMBRE.

CLAVE. Aparece la clave de la medicina como elemento de identificación. Únicamente aparecen aquellas medicinas en las cuales existió movimiento en cuanto a ventas se refiere. Por lo tanto, si el medicamento x tuvo ventas en las medicinas existentes o en las pedidas, entonces será tomada en cuenta para el reporte. Usa un campo de cinco dígitos.

NOMBRE. Es el nombre del medicamento. Usa 30 espacios.

UNIDADES VENDIDAS. En esta sección aparece anotado el número de unidades que se vendieron durante esa sesión de trabajo. Tiene un campo de seis dígitos.

PRECIO DE VENTA. Aquí aparece el precio de venta por unidad, sin impuesto. Maneja diez dígitos.

SUBTOTAL \$. El Subtotal es el producto de Unidades Vendidas por Precio de Venta. Maneja cifras de hasta diez dígito.

I V A \$. Este es el campo del impuesto global de cada medicamento. Da el dato en dinero. Usa cantidades de nueve cifras.

- T O T A L \$.** Es la suma de los campos SUBTOTAL más IVA. Maneja cantidades de diez dígitos.
- PRECIO DE COMPRA.** Es el último precio que la farmacia pasó por su adquisición. Corresponde al proveedor principal. Tiene un campo de ocho dígitos.
- PREC. COMP. X UNID. VENDIDAS.** Es el resultado de multiplicar PRECIO DE COMPRA por UNIDADES VENDIDAS. Maneja nueve dígitos.
- GANANCIA NETA \$.** Este resultado corresponde a la diferencia entre SUBTOTAL y PREC. COMP. X UNID. VENDIDAS. Tiene un campo de diez dígitos.

Estas dos secciones de encabezados aparecen cada 40 renglones, es decir, primero escribe los encabezados y a continuación anota los datos de 40 medicamentos, para que Usted tenga siempre una referencia clara en sus reportes.

3.3 SECCION DE TOTALES.

Corresponde a los totales de algunas columnas del reporte.

Esta sección está identificada por el mensaje:

VENTA TOTAL DEL DIA \$

El mensaje da inicio al renglón en el cual aparecen los totales de las siguientes columnas:

- SUBTOTAL. Campo de quince dígitos.
- IVA. Campo de trece dígitos.
- TOTAL. Campo de quince dígitos.
- PREC. COMP. X UNID. VENDIDAS. Campo de quince dígitos.
- GANANCIA NETA. Campo de catorce dígitos.

3.4 SECCION DE TOTALES DE ARTICULOS FUERA DE INVENTARIO.

En esta sección aparece la información del total de ventas de artículos que están fuera del inventario o del sistema, es decir, corresponden a los artículos cuya información (precio, unidades, etc.), no están en los archivos de trabajo y que Usted anotó en su oportunidad. Por considerarse información de menor importancia, únicamente incluye la venta total en dinero, sin dar el dato correspondiente al impuesto, ya que éste puede cambiar según la naturaleza del producto.

La sección está identificada por el mensaje:

VENTA TOTAL DEL DIA DE ARTICULOS FUERA DE INVENTARIO *

Inmediatamente después del signo de pesos aparece la cantidad en un campo de quince dígitos.

3.5 SECCION DE COMENTARIOS.

El comentario es una indicación que se hace al usuario y se enuncia de la siguiente manera:

NOTA: NO OLVIDE HACER LO SIGUIENTE!

1o. REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. USE LA FUNCION <F> DEL MENU. RECUERDE QUE LA FUNCION <F> SOLO PUEDE USARSE UNA VEZ CON CADA ARCHIVO DE TRABAJO. TENGA CUIDADO.

2o. PASE EL ARCHIVO DE TRABAJO AL DISCO ALMACENADOR.

Esto último es muy importante, ya que una vez que se ha corrido el módulo "CORTE DE CAJA DE LAS VENTAS DEL DIA", entonces debemos actualizar el Archivo Maestro con los nuevos datos, generados por las ventas del día, los cuales obviamente, serán diferentes a los datos que se tenían originalmente en el sistema.

Una vez que el programa termina de hacer el reporte, entonces aparece nuevamente el MENU PRINCIPAL.

4. COMENTARIOS GENERALES.

Es necesario aclarar que Usted puede guardar en sus discos, todos y cada uno de los archivos generados por cada día de trabajo, para que de esta manera tenga la posibilidad de correr la opción <D>, usando el archivo del día que convenga a sus intereses. Se susiere, por orden en el trabajo, que obtenga el "CORTE DE CAJA ..." del día en que se encuentra, es decir, si hoy es 6 de enero de 1986, debe usar el archivo de trabajo (de facturas) del día 6 de enero de 1986.

De esta manera su trabajo seguirá un orden preestablecido.

Otra actividad que debe hacer, antes de pasar los archivos de trabajo al disco almacenador, es cambiar el nombre del archivo de trabajo por uno nuevo, formado con la fecha. Así, el archivo de trabajo del día 6 de enero de 1986, se llamará ENE0686.FIL (MES:tres letras, DIA:dos dígitos y AÑO:dos dígitos y la extensión .FIL). Cuando vuelva a usar este archivo, para obtener otro reporte de Corte de Caja de Ventas del Día, simplemente vuelvalo a llamar TRABAJO.FIL, para que el sistema lo reconozca.

IV.2.6 MANUAL DE OPERACION DEL MODULO: REGRESAR Y TRANSFERIR.

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO
AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE
INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE
NOMBRES Y ESTADISTICAS.

1. PANORAMA ESPECIFICO.

OBJETIVO: Actualizar al Archivo Maestro con la información del Archivo de Trabajo. Transferir información del Archivo Maestro al de Nombres y Estadísticas con el fin de actualizar a los dos.

ARCHIVOS QUE USA: Es necesario que estén presentes los archivos MAESTRO.FIL, TRABAJO.FIL y NOMYEST.FIL en el disco B.

RESTRICCIONES: Para poder usar este módulo, deben existir los tres archivos mencionados. No se puede usar un Archivo TRABAJO.FIL en más de una ocasión para actualizar al Archivo Maestro MAESTRO.FIL, ya que esto generaría información errónea.

No se elabora reporte por escrito, por lo que, esta función se puede correr en pantalla o impresora. Por rapidez se sugiere realizarla en pantalla.

2. PROCEDIMIENTO.

Para tener acceso al módulo "REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS", se debe seleccionar la opción <F> del MENU PRINCIPAL DE FUNCIONES.

Una vez hecho esto, el sistema despliega el siguiente mensaje:

EL ARCHIVO DE TRABAJO QUE ACTUALIZARA
AL ARCHIVO MAESTRO ES DE FECHA :

DIA : DD MES : MM AÑO : AA

Donde DD, corresponde al día, MM al mes y AA a las dos últimas cifras del año.

El sistema verifica que el Archivo de Trabajo sea el adecuado para hacer la actualización. En el caso de que todo esté correcto, saca el siguiente mensaje:

EL ARCHIVO DE TRABAJO ES EL CORRECTO

SE ESTÁ ACTUALIZANDO EL ARCHIVO MAESTRO.
ESPERE LA SENAL PARA PODER CONTINUAR ...

Este mensaje significa que existen los dos archivos necesarios para que se ejecute el proceso.

Por lo tanto, a Usted solo le queda esperar a que el sistema realice la actualización. Una vez que termina despliega el siguiente mensaje:

EL SISTEMA TERMINO DE ACTUALIZAR AL MAESTRO

Significa que todo el proceso se realizó correctamente y por lo tanto termina la función del módulo y regresa al MENU PRINCIPAL DE FUNCIONES.

Pero si aparece el mensaje:

*** E R R O R ***

ESTE ARCHIVO DE TRABAJO YA FUE USADO UNA VEZ.

NO ESTA PERMITIDO ACTUALIZAR DOS VECES
CON EL MISMO ARCHIVO DE TRABAJO

Significa que Usted pretende actualizar con un archivo de trabajo que ya fue usado para tal fin. Por lo que, no está permitido usar en más de una ocasión al mismo archivo de trabajo para actualizar al archivo Maestro. Por lo tanto ya no puede volver a usarlo, ya que generaría información errónea en el Archivo Maestro. Tenga cuidado. Evite este error. El sistema está diseñado para evitar que Usted trate de actualizar con el mismo archivo más de una vez.

En caso de que se presente este problema, el sistema le despliega nuevamente el MENU PRINCIPAL DE FUNCIONES. En cuanto suceda esto, elija la opción <Q> para salir de sesión. Si por algún motivo en ninguna ocasión a usado el archivo de trabajo para actualizar, es posible que tenga en el disco B un archivo equivocado. Si se presenta este caso, el cual es muy raro pero puede suceder, inmediatamente busque entre sus Archivos TRABAJO.FIL aquel que le sirva y cópielo al disco B, que está usando actualmente, para que ya pueda trabajar correctamente.

Finalmente, el sistema le pide la fecha correspondiente al día que Usted está haciendo la actualización. Despliega el siguiente mensaje:

ANOTE LA FECHA DE HOY:

A N O =>

M E S =>

D I A =>

Escriba los datos correspondientes, uno por renglón, transmitiendo dato por dato. Haga uso de la tabla que aparece a continuación para anotar el mes:

TABLA DE MESES

NOMBRE	EQUIVALENTE NUMERICO
ENERO	1
FEBRERO	2
MARZO	3
ABRIL	4
MAYO	5
JUNIO	6
JULIO	7
AGOSTO	8
SEPTIEMBRE	9
OCTUBRE	10
NOVIEMBRE	11
DICIEMBRE	12

Para el año, anote únicamente las dos últimas cifras, es decir para 1986, solo escriba 86.

Si anotó la fecha sin errores, entonces el sistema desplega el mensaje:

ANOTO LA FECHA CORRECTAMENTE AND: aa, MES: mm, DIA: dd

Donde aa, mm y dd, corresponden a los datos que Usted anotó.

[481]

En caso contrario, se la pide nuevamente, de la siguiente manera:

**** E R R O R ****

ANOTE LA FECHA NUEVAMENTE

Y se repite el proceso para anotar la fecha.

Tome en cuenta lo siguiente: el sistema no sale de esta sección hasta que Usted le escriba una fecha CORRECTA. Por lo tanto, tenga cuidado y escriba la fecha sin errores.

Ahora el Programa le pide la ratificación de la fecha:

REALMENTE ESTA CORRECTA LA FECHA ? ANOTE S)I o N)O =>

Al salir de esta sección, el sistema despliega el siguiente mensaje:

EL SISTEMA ESTA TRANSFIRIENDO INFORMACION

ESPERE LA SENAL PARA PODER CONTINUAR

No haga nada, deje que el sistema trabaje por Usted.

Cuando el sistema termina, se lo indica con el comentario:

EL SISTEMA YA TERMINO

PUEDE USTED CONTINUAR.

Al terminar, el sistema sale de esta función y se da por terminada la sesión.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

No se obtiene reporte de salida; por lo tanto esta función puede correrse en pantalla o impresora.

4. COMENTARIOS GENERALES.

Por qué se genera información errónea ?

Se generaría información errónea si se permitiese que Usted, actualizase con un mismo archivo de trabajo en más de una ocasión.

Sin embargo como ya se explicó, el sistema no permite hacer lo anterior. Por lo que, su información siempre está protegida contra todo y contra todos.

IV.2.7 MANUAL DE OPERACION DEL MODULO ACTUALIZAR.

NOMBRE EN EL MENU PRINCIPAL DE FUNCIONES:

CORRECCIONES AL ARCHIVO DE TRABAJO.

1. PANORAMA ESPECIFICO.

OBJETIVO: Hacer correcciones al Archivo de Trabajo, que contiene las ventas del día.

ARCHIVOS QUE USA: Debe existir el Archivo de Trabajo en el disco B, llamado TRABAJO.FIL.

RESTRICCIONES: Es necesario que el archivo de trabajo contenga información. En caso de que falte el Archivo de Trabajo o que esté vacío, no se le permite entrar a esta función, ya que no es posible corregir algo que no existe.

2. PROCEDIMIENTO.

2.1 ACCESO A LA FUNCION.

Para entrar a esta función, debe elegir del MENU PRINCIPAL DE FUNCIONES, la función "CORRECCIONES AL ARCHIVO DE TRABAJO", por medio de la opción <E>.

Quando el sistema despliega el MENU PRINCIPAL DE FUNCIONES y le hace la Presunta:

FUNCION DESEADA:

Usted debe anotar <E>.

El sistema despliega la fecha del archivo como sigue:

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :
ANO : 86, MES : 1, DIA : 3

Y le hace una Presunta:

ESTE ES EL ARCHIVO DE TRABAJO QUE USTED DESEA ?
ANOTE S)I o N)O =>

Este mensaje es muy importante, ya que tomando como referencia la fecha, Usted decide si le es útil o no, por lo que debe aceptar o rechazar el archivo, simplemente anotando S o N.

Si Usted anota N, entonces se sale de la función sin actualizar nada, dándose por terminada la sesión.

Si su respuesta es afirmativa, es decir anotó S, entonces el sistema le pide las claves de seguridad, una a una, de la siguiente forma:

ANOTE LA CLAVE PRIMARIA DE SEGURIDAD =>

Usted debe anotar la clave correspondiente y transmitir.

El sistema pide:

ANOTE LA CLAVE SECUNDARIA DE SEGURIDAD =>

Anote el dato solicitado.

Si alguna de las claves que escribió Usted, no corresponden con las que guarda el sistema, simplemente no se le permite el acceso a la función y sale del sistema, desplegando el siguiente mensaje:

USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES.

Y se termina la sesión.

2.2 CORRECCION DE REGISTROS.

Únicamente en el caso de que haya dado las dos claves correctamente, entonces el sistema lo acepta y está listo para hacer las correcciones que Usted disponga.

Así, el sistema escribe:

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => (1)

Donde registro es equivalente a la CLAVE del MEDICAMENTO.

Tenga mucho cuidado, al escribir el número que se le pide, ya que el sistema solamente aceptará, aquellos que se encuentran dentro del rango permitido, es decir, únicamente podrá anotar el número que se encuentre entre 0 y el límite máximo de registros útiles contenidos en el Archivo de Trabajo, conocido también como LIMITE. Este dato se lo dá a conocer el sistema al iniciar cualquier sesión de trabajo.

Pero, no se preocupe tanto, aquí mismo usted puede saberlo, si es que ya se le olvidó.

Si Usted anota el número de un registro reconocido por el sistema, entonces pasa lo siguiente:

Suponga que quiere corregir el registro número 10, es decir el medicamento 10, entonces como respuesta a la última pregunta del sistema anote el número 10 y transmita.

Si el registro 10 se encuentra dentro del rango permitido, en su pantalla o terminal aparecen listados todos los campos que Usted tiene opción de corregir en este registro. El listado es así:

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES :

```
=====
C           A           M           P           O
é  N       O           M       B           R       E
=====
```

1	UNIDADES EN EXISTENCIA ACTUALES
2	UNIDADES VENDIDAS ACTUALES
3	UNIDADES PEDIDAS
4	UNIDADES VENDIDAS DEL PEDIDO
5	NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)
6	VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)

```
=====
```

Observe, que los dos últimos campos (5 y 6) son exclusivos del medicamento cero.

Y presenta la siguiente pregunta:

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO =>

La identificación de los campos, aparecerá únicamente cuando Usted anote la respuesta correcta a la pregunta (1). Si Usted se está iniciando en esto, es recomendable que tenga por escrito, la identificación de los campos.

Usted únicamente puede actualizar los campos que aparecen en la identificación de los mismos. Como respuesta anote el número del campo del 1 al 6.

Pero vea un detalle muy importante. Por ejemplo, si desea corregir o actualizar el registro 0, solamente puede usar los campos 5 y 6, ningún otro más. Para otros registros diferentes de 0, puede usar los campos del 1 al 4.

No Juegue, ni trate de probar si funciona o no el sistema, el cual está hecho para detectar todo tipo de errores.

Ya que insiste, si anota por ejemplo el número 20, (! hágalo!), el sistema despliega lo siguiente:

* * * E R R O R * * *

NO EXISTE, EL CAMPO 20 ESTA FUERA DEL RANGO

Si anotó un campo válido, continua con el proceso, apareciendo el siguiente mensaje:

DATO VIEJO: XXX , ANOTE EL DATO NUEVO :

Aquí, XXX corresponde al dato numérico contenido en el campo solicitado por Usted; el sistema le pide que anote el dato nuevo. El cursor se queda esperando el dato. Dato nuevo corresponde a lo que Usted quiera que se guarde en ese campo del registro 10 en lugar de lo que apareció en XXX.

Una vez que haya anotado el nuevo dato, entonces el sistema pide la ratificación de lo escrito por Usted, así:

ESTA CORRECTO ? ANOTE S)I O N)O =>

Tenga mucho cuidado con su respuesta. Si cometió algún error, simplemente anote N y puede volver a solicitar este campo. Si su respuesta es afirmativa, ese será el nuevo dato que se guarde en el campo presente.

Una vez hecho esto, es decir, que ya actualizó la información contenida en algún campo, entonces el sistema escribe:

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO NUM ?
ANOTE S)I O N)O =>

Donde NUM corresponde al registro que en ese momento se está modificando.

Para una respuesta afirmativa, el sistema vuelve a anotar :

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO =>

y se repite todo nuevamente.

En caso de una respuesta negativa, es decir, cuando ya no quiere hacer más correcciones a otro campo del mismo registro NUM, el sistema despliega el mensaje:

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN> (A)
EN CASO CONTRARIO ANOTE LA LETRA Q =>

Si su respuesta es afirmativa, simplemente presione la tecla <RETURN> y el sistema despliega la pregunta (1).

Y empezará a trabajar el registro cuyo número haya Usted anotado.

2.3 ERRORES QUE PUEDE COMETER Y COMO SALIR DE ELLOS.

Para la pregunta:

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO =>

Si Usted anota un número fuera del rango, entonces el sistema despliega el siguiente mensaje:

* * * E R R O R * * *

NO EXISTE EL REGISTRO NUM, ESTA FUERA DEL RANGO (LIMITE)

Donde LIMITE corresponde la clave máxima de medicamento contenida en el sistema.

Este mensaje significa que el registro NUM, que Usted anotó, no es reconocido por el sistema, ya que esta fuera del rango. Entre paréntesis, aparece anotado el máximo número de registros que reconoce el sistema.

No se preocupe, si comete este error, no tiene que hacer nada, el sistema lo hace por usted, y despliega el mensaje:

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN>
EN CASO CONTRARIO, ANOTE LA LETRA Q =>

y siga como si no hubiera pasado nada.

Para la pregunta:

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO =>

Si está en el registro 0, únicamente podrá modificar los campos 5 y 6. Pero si por error anota otro número de campo, digamos del 1 al 4, el sistema escribe:

[490]

UNICAMENTE PUEDE CORREGIR LOS CAMPOS 5 Y 6
DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 0 ?
ANOTE S)I O N)O =>

Usted decide.

Para la misma pregunta, si usted anota un número de campo menor que 1 o mayor que 6, digamos 567, el sistema despliega:

* * * E R R O R * * *

NO EXISTE, EL CAMPO 567 ESTA FUERA DEL RANGO

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 0 ?
ANOTE S)I O N)O =>

Para la misma pregunta, si usted está trabajando con un registro diferente de 0, digamos el 7, solamente puede modificar información de los campos del 1 al 4 y no 5 y 6, que son exclusivos del registro 0. Si insiste en corregir el campo 5, por ejemplo, el sistema despliega:

UNICAMENTE PUEDE CORREGIR LOS CAMPOS DEL 1 AL 4
DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 7 ?
ANOTE S)I O N)O =>

2.4 COMO SALIR DE LA FUNCION.

Para salir de esta función, cuando aparezca la pregunta (A), simplemente anote la letra Q y se da por terminada la sesión.

3. DESCRIPCION DEL REPORTE QUE GENERA ESTA FUNCION.

La función "CORRECCIONES AL ARCHIVO DE TRABAJO", no genera reporte por escrito. Está basada en preguntas y respuestas entre el sistema y Usted.

Esta función se puede correr, tanto en la terminal como en la impresora. Sin embargo se recomienda que se haga en la terminal, por las siguientes razones:

a) Rapidez. El trabajo se hace más rápido, ya que la impresora resulta lenta para este objetivo.

b) Economía. Evita usted el gasto de papel, ya que no tiene caso llevar un registro por escrito de las correcciones que se hagan.

c) Comodidad. Resulta más cómodo, trabajar en la terminal.

4. COMENTARIOS GENERALES.

Por seguridad del sistema, se recomienda NO usar esta función, ya que podría generar problemas, si no se tiene un estricto control en las correcciones que se hagan al archivo de trabajo.

Sin embargo, Usted se preguntará que uso práctico puede tener esta función.

La respuesta es la siguiente:

Use esta función cuando en verdad sea extremadamente necesario hacer correcciones al archivo, en los siguientes casos por ejemplo:

- Para corregir la VENTA TOTAL DE ARTICULOS FUERA DE INVENTARIO, por algún error generado al momento de hacer las facturas.

- Falla de energía eléctrica durante una sesión de trabajo.

Pero recuerde que para entrar a esta función, deben darse las claves de seguridad, las cuales son conocidas únicamente por la persona de mayor jerarquía en la farmacia.

[492]

C A P I T U L O V

L I S T A D O S F U E N T E S

D E L O S

P R O G R A M A S

D E L

S I S T E M A

[493]

V.1 LISTADOS FUENTES DE LOS PROGRAMAS
DEL SUBSISTEMA DE
ACTUALIZACION
Y
CONSULTA
DE
REQUERIMIENTOS

V.1.1 LISTADO FUENTE DEL PROGRAMA :

< S A B A D O >

```
*****  
*                                     *  
*          PROGRAMA          PRINCIPAL          SABADO          *  
*                                     *  
*****
```

PROGRAM NOCTURNO;

```
TYPE ARCHI=RECORD  
  CLAVE,MS1,MS2,MS3:INTEGER;  
  PRECIO:REAL;  
  UNID,ANN,MES,DIA:INTEGER;  
  CP:INTEGER;  
  IMP,FDU:REAL;  
  UVEN,UPED,UVP1:INTEGER;  
  AA,MM,DD:INTEGER;  
END;  
  ARCHII=RECORD  
  CLAVE:INTEGER;  
  MEDS1:INTEGER;  
  MEDS2:INTEGER;  
  MEDS3:INTEGER;  
  PRECIO:REAL;  
  UNID:INTEGER;  
  IMP:REAL;  
  UVEN:INTEGER;  
  UVPD:INTEGER;  
END;  
  ARCHIII=RECORD  
  CLAVE:INTEGER;  
  NOMBRE:STRING(306);  
  R1,R2,R3,R4:INTEGER;  
  PROV1:INTEGER;  
  PRE1,DP1:REAL;  
  PROV2:INTEGER;  
  PRE2,DP2:REAL;  
  PROV3:INTEGER;  
  PRE3,DP3:REAL;  
  DD,MM,AA:INTEGER;  
END;  
  ARCHIV=RECORD  
  CLAVE,UR,CP,AP:INTEGER;  
END;  
  ARCHV=RECORD  
  CDISP:INTEGER;  
END;
```

```

VAR BF:FILE OF ARCHI; (*ARCHIVO MAESTRO*)
    VA:FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)
    NM:FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS *)
    PD:FILE OF ARCHIV; (*ARCHIVO DE PEDIDOS*)
    NC:FILE OF ARCHV; (*ARCHIVO DE CLAVES DISPONIBLES*)

```

```

I,I1,LIM,II,X1,J:INTEGER;
ERROR:BOOLEAN;
CVMED,UNIDP,CVEPRO,SUMUNI,ANIO,MES,DIA:INTEGER;
PRECV,DESCV,SFP:REAL;
SN,BAND,BAND1,OPC1,OPC2:CHAR;
ACVE,ACVEA:ARRAY11..106 OF INTEGER;

```

```

EXTERNAL 116 PROCEDURE LPEDIDO;
EXTERNAL 126 PROCEDURE LISPED;
EXTERNAL 136 PROCEDURE LPED;
EXTERNAL 146 PROCEDURE CADUCA;
EXTERNAL 156 PROCEDURE BAJA;
EXTERNAL 166 PROCEDURE WRITERE;
EXTERNAL 176 PROCEDURE LISTAR;
EXTERNAL 186 PROCEDURE ACTUAL;
EXTERNAL 196 PROCEDURE DAINF;
EXTERNAL 1106 PROCEDURE FECHA;
EXTERNAL 1116 PROCEDURE LISTAP;
EXTERNAL 1126 PROCEDURE CARGPED;
EXTERNAL 1136 PROCEDURE LEERAN;

```

```

*****
* PROCEDURE ERRCHK *
*****

```

```

PROCEDURE ERRCHK;
BEGIN
  ERROR:=TRUE;
  CASE IORESULT OF
    0:ERROR:=FALSE;
    1:WRITELN('** ERROR EL DISCO NO CONTIENE EL ARCHIVO MAESTRO DE DATOS);
    2:WRITELN(' ** ERROR DE CP/M ** ');
    3:WRITELN('SEEKING TO UNWRITTEN EXTENT');
    4:WRITELN('** ERROR DE CP/M **');
    5:WRITELN('** LLEGO AL FINAL DEL DISCO **');
  ELSE
    WRITELN('UNRECOGNIZABLE ERROR CODE')
  END;
END;
END;

```

```
*****  
* PROCEDURE ESC *  
*****
```

```
PROCEDURE ESC;  
BEGIN  
  SEEKWRITE(PD,I1);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
END;
```

```
*****  
* PROCEDURE LEE *  
*****
```

```
PROCEDURE LEE;  
BEGIN  
  SEEKREAD(PD,I1);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
END;
```

```
*****  
* PROCEDURE ASIGNA *  
*****
```

```
PROCEDURE ASIGNA;  
BEGIN  
  PD .CLAVE:=PD .CLAVE;  
  PD .UR:=PD .UR;  
  PD .CP:=PD .CP;  
END;
```

```
*****  
* PROCEDURE DNUEVO *  
*****
```

```
PROCEDURE DNUEVO;  
BEGIN  
  WRITELN('ERROR TRANSMITA UNICAMENTE <S> O <N>, ');  
  WRITE('FAVOR DE VOLVER A TRANSMITIR S)I O N)D => ');  
  BAND:='*';  
END;
```

```
*****  
* PROCEDURE VALIDA *  
*****
```

```
PROCEDURE VALIDA;
```

```
BEGIN
```

```
IF LIM=0 THEN
```

```
  BEGIN
```

```
    I:=LIM+1;
```

```
    WRITELN('ERROR : EL ARCHIVO DE DATOS AUN NO CONTIENE INFORMACION ');
```

```
    WRITELN(' FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA ');
```

```
    WRITELN(' OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO ');
```

```
  END;
```

```
END;
```

```
*****  
* PROCEDURE GRAPED *  
*****
```

```
PROCEDURE GRAPED;
```

```
VAR IA:INTEGER;
```

```
BEGIN
```

```
  I1:=I1+1;
```

```
  PD .CLAVE:=CVEMED;
```

```
  PD .UR:=UNIDP;
```

```
  PD .CP:=CVEPRO;
```

```
  IF ACVE1CVEPRO6=0 THEN
```

```
    BEGIN
```

```
      ACVE1CVEPRO6:=I1;
```

```
      ACVEA1CVEPRO6:=I1;
```

```
      IA:=I1;
```

```
    END
```

```
  ELSE
```

```
    BEGIN
```

```
      ESC;
```

```
      IA:=I1;
```

```
      I1:=ACVE1CVEPRO6;
```

```
      LEE;
```

```
      ASIGNA;
```

```
      PD .AP:=IA;
```

```
      ACVE1CVEPRO6:=IA;
```

```
    END;
```

```
  ESC;
```

```
  I1:=IA;
```

```
END;
```

```
*****
* PROCEDURE GRAAPT *
*****
```

```
PROCEDURE GRAAPT;
VAR I9,IB,VACP:INTEGER;
BEGIN
  IB:=0;
  FOR I9:=1 TO 10 DO
    BEGIN
      IF ACVEA1I9 <> 0 THEN
        BEGIN
          IB:=IB+1;
          IF IB=1 THEN
            VACP:=ACVE1I9;
          IF IB>1 THEN
            BEGIN
              I1:=VACP;
              LEE;
              ASIGNA;
              PD .AP:=ACVEA1I9;
              ESC;
              VACP:=ACVE1I9;
            END;
          END;
        END;
      END;
      I1:=VACP;
      LEE;
      ASIGNA;
      PD .AP:=0;
      ESC;
      PD .AP:=ACVEA1I9;
      IF PDR.AP = 0 THEN
        PD .AP:=1;
        PD .CLAVE:=ANIO;
        PD .UR:=MES;
        PD .CP:=DIA;
        I1:=0;
      ESC;
    END;
  END;
```

```
*****
* PROCEDURE PIDE *
*****
```

```
PROCEDURE PIDE;
```

```
BEGIN
```

```
WRITE('DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O => ');
```

```
REPEAT
```

```
READ(SN);WRITELN;
```

```
IF SN IN 1'S','s'0 THEN
```

```
BEGIN
```

```
WRITELN('DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD ');
```

```
WRITE('SUGERIDA?, TRANSMITA S)I O N)O => ');
```

```
REPEAT
```

```
BAND:=' ';
```

```
READ(SN);WRITELN;
```

```
CASE SN OF
```

```
'S','s':WRITELN('GRABA PEDIDO');
```

```
'N','n':DAINF;
```

```
ELSE
```

```
DNUEVO;
```

```
END
```

```
UNTIL BAND = ' ';
```

```
GRAPED;
```

```
SPP:=PREC*UNIDP-PREC*UNIDP*DESCV+SPP;
```

```
WRITELN('DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA ');
```

```
WRITE('AHORA PEDIDO?, TRANSMITA S)I O N)O => ');
```

```
REPEAT
```

```
BAND:=' ';
```

```
READ(SN);WRITELN;
```

```
CASE SN OF
```

```
'S','s':WRITELN('HASTA AHORA LLEVA PEDIDO.- $',SPP:9:2);
```

```
'N','n':WRITELN;
```

```
ELSE
```

```
DNUEVO;
```

```
END
```

```
UNTIL BAND=' ';
```

```
END
```

```
ELSE
```

```
IF SN IN 1'N','n'0 THEN
```

```
BAND:=' ';
```

```
ELSE
```

```
DNUEVO;
```

```
UNTIL BAND=' ';
```

```
WRITE('DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA ');
```

```
WRITE('S)I O N)O => ');
```

```
REPEAT
```

```
BAND:=' ';
```

```
READ(SN);WRITELN;
```

```
CASE SN OF
```

```
'S','s':WRITELN;
```

```
'N','n':I:=LIN+1;
```

```
ELSE
```

```
DNUEVO;
```

```
END
```

```
UNTIL BAND=' ';
```

```
END; (* FIN DE PIDE *)
```

```
*****
* PROCEDURE SUGPED *
*****
```

```
PROCEDURE SUGPED;
VAR EZ:INTEGER;
    PMUE:REAL;
BEGIN
  ASSIGN(PD,'B:PEDIDOS.FIL');
  RESET(PD);
  I:=1;I1:=0;SPP:=0;
  FOR EZ:=1 TO 10 DO
  BEGIN
    ACVE1EZ0:=0;
    ACVEA1EZ0:=0;
  END;
  VALIDA;
  WHILE I<=LIM DO
  BEGIN
    SEEKREAD(BF,I);
    ERRCHK;
    IF ERROR THEN EXIT;
    SUMUNI:=BF .UNID+BF .UVEN;
    PMUE:=BF .PDU*SUMUNI;
    IF (PMUE>=BF .UNID) AND (BF .UPED=0) THEN
    BEGIN
      II:=I;
      SEEKREAD(NM,II);
      ERRCHK;
      IF ERROR THEN EXIT;
      LPEDIDO;
      WRITELN;WRITELN;
      LPED;
      PIDE;
    END;
    I:=I+1;
  END;
  IF LIM > 0 THEN
  IF I1>0 THEN
  GRAAPT
  ELSE
  BEGIN
    WRITELN;
    WRITELN(' *** NO GRABO NINGUN PEDIDO *** ');
    WRITELN;
  END;
  WRITELN;
  WRITELN(' *** TERMINA PROGRAMA DE SUGERIR PEDIDOS *** ');
  CLOSE(PD,I1);
END; (* FIN DE SUGPED *)
```

```
*****
* PROCEDURE PEDXTEM *
*****
```

```
PROCEDURE PEDXTEM;
VAR
IL,CUVXT, LONGNM, NECNM, E1, E2, SUMU, CUVXTE: INTEGER;
NOMMED: STRING(130);
BEGIN
ASSIGN(PD, 'B:PEDIDOS.FIL');
RESET(PD);
IF LIM > 0 THEN
BEGIN
LISTAP;
READ(CUVXT); WRITELN;
I:=1;
I1:=0; SPP:=0;
FOR IL:=1 TO 10 DO
BEGIN
ACVEIIL6:=0;
ACVEAII6:=0;
END
END;
VALIDA;
WHILE I <= LIM DO
BEGIN
II:=I;
SEEKREAD(NM, II);
ERRCHK;
IF ERROR THEN EXIT;
SEEKREAD(BF, I);
ERRCHK;
IF ERROR THEN EXIT;
CASE OPC2 OF
'A': CUVXTE:=NM .R1;
'B': CUVXTE:=NM .R2;
'C': CUVXTE:=NM .R3;
'D': CUVXTE:=NM .R4;
ELSE
WRITELN(' *** ERROR FATAL *** ');
END;
IF CUVXTE >= CUVXT THEN
BEGIN
NOMMED:=NM .NOMBRE;
LONGNM:=LENGTH(NOMMED);
NECNM:=30-LONGNM;
E1:=NECNM DIV 2;
E2:=NECNM MOD 2;
IF E2=0 THEN
E2:=E1
ELSE
E2:=E1+1;

```

```

WRITE(' :2,'CLAVE',' :8,'NOMBRE DEL MEDICAMENTO',' :3);
WRITELN('UNIDADES VENDIDAS',' :3,'UNIDADES EN');
WRITELN(' :40,'EN ESTA TEMPORADA',' :3,'EXISTENCIA');
WRITELN;
SUMU:=BF .UNID + BF .UPED;
WRITE(' :2,NM .CLAVE;5,' :3,' :E1,NOMMED,' :E2,' :5);
WRITELN(CUVXTE:6,' :11,SUMU:6);
WRITELN;
LPED;
PIDE;
END;
I:=I+1;
END;
IF LIM > 0 THEN
  IF I>0 THEN
    GRAAPT
  ELSE
    BEGIN
      WRITELN;
      WRITELN(' *** NO GRABO NINGUN PEDIDO *** ');
      WRITELN;
    END;
  CLOSE(PD,I);
  WRITELN;
  WRITELN('**** TERMINA PROGRAMA DE PEDIDOS POR TEMPORADA ****');
  WRITELN;
  END; (* FIN DE SUGPED *)

```

```

*****
* PROCEDURE LISTMENU *
*****

```

```
PROCEDURE LISTMENU;
```

```
BEGIN
```

```

WRITELN;WRITELN;
WRITELN(' :4,'FUNCION',' :7,'OPERACION');
WRITELN('-----');
WRITELN(' :17,'A',' :18,'SUGERIR PEDIDOS');
WRITELN(' :17,'B',' :18,'LISTAR PEDIDOS ');
WRITE(' :17,'C',' :18,'SUGERIR COMPRA DE ARTICULOS POR TEMPORADA');
WRITE(' :17,'D',' :18,'CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS');
WRITELN(' :16,'A DARSE DE BAJA POR CADUCIDAD');
WRITELN(' :17,'E',' :18,'DAR DE BAJA ALGUN ARTICULO');
WRITELN(' :17,'F',' :18,'DAR DE ALTA ALGUN ARTICULO');
WRITELN(' :17,'G',' :18,'LISTAR LA INFORMACION');
WRITELN(' :17,'H',' :18,'ACTUALIZAR INFORMACION');
WRITELN(' :17,'I',' :18,'CARGA DE UNIDADES PEDIDAS');
WRITELN(' :17,'J',' :18,'LISTA RANGOS TEMPORADA Y LISTA PROVEERS');
WRITELN(' :17,'K',' :18,'SALIRSE DEL SISTEMA');
WRITELN;
END; (* FIN LISTA MENU *)

```

```
*****
*          PROCEDURE PRINCIPAL          *
*****
```

```
BEGIN (*PROGRAMA PRINCIPAL*)
  ASSIGN(BF,'B:MAESTRO.FIL');
  RESET(BF);
  ASSIGN(VA,'B:TRABAJO.FIL');
  RESET(VA);
  ASSIGN(NM,'B:NOMYEST.FIL');
  RESET(NM);
  I:=0;
  SEEKREAD(BF,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  II:=0;
  SEEKREAD(NM,II);
  ERRCHK;
  IF ERROR THEN EXIT;
  FECHA;
  LIM:=BF .MSI;
  REPEAT
    WRITE('DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => ');
    READ(SN);
    IF SN = 'S' THEN
      BEGIN
        LISTMENU;
        REPEAT
          BAND1:=' ';
          WRITE(' :3, 'DEME LA FUNCION QUE DESEA => ');
          READ(OPC1);WRITELN;
          CASE OPC1 OF
            'A':SUGPED;
            'B':LISPED;
            'C':PEDXTEM;
            'D':CADUCA;
            'E':BAJA;
            'F':WRITERE;
            'G':LISTAR;
            'H':ACTUAL;
            'I':CARGPED;
            'J':LEERAN;
            'K':BAND:='*';
          ELSE
            BEGIN
              WRITELN(' :9, '*** E R R O R *** ');
              WRITELN('TRANSMITA UNICAMENTE DE LA <A> A LA <K> ');
              BAND1:='*';
            END
          END
        UNTIL BAND1=' ';
      END
      ELSE
        BAND:='*';
    UNTIL BAND='*';
  WRITELN(' *** TERMINA PROGRAMA DE ACTUALIZACION Y CONSULTA ***');
  WRITELN(' :14, '*** DE REQUERIMIENTOS ***');
  CLOSE(BF,I); CLOSE(VA,J); CLOSE(NM,II);
END.
```

V.1.2 LISTADO FUENTE DEL PROGRAMA:

< MOD 1 >

```
*****  
*                                     *  
*   PROGRAMA   MOD1                 *  
*                                     *  
*****
```

MODULE OVERLAY1:

TYPE ARCHI=RECORD

```
CLAVE,MS1,MS2,MS3:INTEGER;  
PRECIO:REAL;  
UNID,ANN,MES,DIA:INTEGER;  
CP:INTEGER;  
IMP,PDU:REAL;  
UVEN,UPED,UVP1:INTEGER;  
AA,MM,DD:INTEGER;  
END;
```

ARCHIII=RECORD

```
CLAVE:INTEGER;  
NOMBRE:STRING1306;  
R1,R2,R3,R4:INTEGER;  
PROV1:INTEGER;  
PRE1,DP1:REAL;  
PROV2:INTEGER;  
PRE2,DP2:REAL;  
PROV3:INTEGER;  
PRE3,DP3:REAL;  
DD,MM,AA:INTEGER;  
END;
```

VAR BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)

NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO NOMBRES Y ESTADISTICAS*)

ANIO,MES,DIA,SUNUNI:EXTERNAL INTEGER;

ERROR:EXTERNAL BOOLEAN;

NDV,E1,E2,E3:INTEGER;

PV1,PV2,PV3,DV1,DV2,DV3:REAL;

EXTERNAL PROCEDURE ERRCHK;

```
*****
* PROCEDURE NUMDIA *
*****
```

```
PROCEDURE NUMDIA;
VAR DIFA,DIFM,DIFD,TIPAR,TIPAC,TIPAV:INTEGER;
PROCEDURE MES1;
BEGIN
CASE NM .MM OF
1,3,5,7,8,10,12:NDV:=31-NM .DD;
2:IF TIPAR=0 THEN
NDV:=29-NM .DD
ELSE
NDV:=28-NM .DD;
4,6,9,11:NDV:=30-NM .DD;
END;
DIFA:=NM .MM+1;
END;
```

```
*****
* PROCEDURE MES2 *
*****
```

```
PROCEDURE MES2;
BEGIN
CASE DIFA OF
1,3,5,7,8,10,12:NDV:=NDV+31;
2:IF TIPAC=0 THEN
NDV:=NDV+29
ELSE
NDV:=NDV+28;
4,6,9,11:NDV:=NDV+30;
END;
DIFA:=DIFA+1;
END;
```

```
BEGIN
DIFA:=ANIO-NM .AA;
DIFM:=MES-NM .MM;
DIFD:=DIA-NM .DD;
TIPAR:=NM .AA MOD 4;
TIPAC:=ANIO MOD 4;
IF DIFA=0 THEN
IF DIFM=0 THEN
NDV:=DIFD
ELSE
BEGIN
MES1;
WHILE DIFA<MES DO
MES2;
NDV:=NDV+DIA;
END
ELSE
```

```

IF DIFA<4 THEN
BEGIN
MES1;
TIPAC:=TIPAR;
WHILE DIFA<=12 DO
MES2;
DIFA:=NM .AA+1;
WHILE DIFA<ANIO DO
BEGIN
TIPAV:=DIFA MOD 4;
IF TIPAV=0 THEN
NDV:=NDV+36
ELSE
NDV:=NDV+35;
DIFA:=DIFA+1;
END;
DIFA:=1;
TIPAC:=ANIO MOD 4;
WHILE DIFA<MES DO
MES2;
NDV:=NDV+DIA;
END
ELSE
BEGIN
WRITELN('EL ARTICULO ESTA OBSOLETO');
NDV:=9998;
END;
NDV:=NDV+1;
END;

```

```

*****
*           PROCEDURE LPEDIDO           *
*****

```

```

PROCEDURE LPEDIDO;
BEGIN

```

```

WRITE(' :2,'CLAVE', ' :4,'UNIDADES', ' :8,'UNIDADES');
WRITELN(' :10,'DIAS DE VENTA TRANSCURRIDOS');
WRITE(' :11,'EN EXIT.', ' :17,'QUE HABIA', ' :9);
WRITELN('DE ESTA CANTIDAD DE UNIDADES');
WRITE(' :10,'(,'DIA:2,'/','MES:2,'/','ANIO:2,')', ' :5,'(');
WRITELN(NM .DD:2,'/',NM .MM:2,'/',NM .AA:2,');
WRITELN;
NUMDIA;
WRITE(' :2,BF .CLAVE:5,' :5,BF .UNID:6,' :10,SUMUNI:6);
WRITELN(' :19,NDV:4,' :1,'DIAS');
WRITELN;WRITELN;
WRITELN(' :9,'PRECIO X UNIDAD', ' :19,'DESCUENTO DEL PROVEEDOR');
WRITE(' :1,'PROVEEDOR1', ' :2,'PROVEEDOR2', ' :12,'PROVEEDOR3');
WRITELN(' :10,NM .PROV:2,' :8,NM .PROV2:2,' :8,NM .PROV3:2);
WRITELN;
PV1:=NM .PRE1;
PV2:=NM .PRE2;

```

```

PV3:=NM .PRE3;
DV1:=NM .DP1*100.0;
DV2:=NM .DP2*100.0;
DV3:=NM .DP3*100.0;
IF DV1>9.0 THEN
  E1:=1
ELSE
  E1:=2;
IF DV2>9.0 THEN
  E2:=1
ELSE
  E2:=2;
IF DV3>9.0 THEN
  E3:=1
ELSE
  E3:=2;
WRITE(' ',PV1:6:2,' '13,' $ ',PV2:6:2,' '14,' $ ');
WRITE(PV3:6:2,' '19,' 'E1,DV1:2:1,' X ',' '2);
WRITELN(' 'E2,DV2:2:1,' X ',' '2,' 'E3,DV3:2:1,' X ');
END; (* FIN DE LPEDIDO *)

```

·MODEND.

V.1.3 LISTADO FUENTE DEL PROGRAMA :

< M O D 2 >

```
*****  
*                                     *  
*   PROGRAMA   MOD2                 *  
*                                     *  
*****
```

MODULE OVERLAY2;

```
TYPE ARCHIII=RECORD  
  CLAVE:INTEGER;  
  NOMBRE:STRING1306;  
  R1,R2,R3,R4:INTEGER;  
  PROV1:INTEGER;  
  PRE1,DP1:REAL;  
  PROV2:INTEGER;  
  PRE2,DP2:REAL;  
  PROV3:INTEGER;  
  PRE3,DP3:REAL;  
  DD,MM,AA:INTEGER;  
END;  
  ARCHIV=RECORD  
  CLAVE,UR,CP,AP:INTEGER;  
END;
```

VAR

```
NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)  
PD:EXTERNAL FILE OF ARCHIV; (*ARCHIVO DE PEDIDOS*)  
I1,I2,LIM:EXTERNAL INTEGER;  
PRECV,DESCV:EXTERNAL REAL;  
ERROR:EXTERNAL BOOLEAN;  
LIE,E1,E2,E3,E4,E5,AAP,MMP,DDP,UR:INTEGER;  
NOMMED,NOMP:STRING1306;  
MESP:STRING1106;  
PNXMED:REAL;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****
* PROCEDURE ASIGMES *
*****
```

```
PROCEDURE ASIGMES;
```

```
BEGIN
```

```
  CASE MMP OF
```

```
    1:MESP:= 'ENERO'   ;
    2:MESP:= 'FEBRERO';
    3:MESP:= 'MARZO'  ;
    4:MESP:= 'ABRIL'  ;
    5:MESP:= 'MAYO'   ;
    6:MESP:= 'JUNIO'  ;
    7:MESP:= 'JULIO'  ;
    8:MESP:= 'AGOSTO' ;
    9:MESP:= 'SEPTIEMBRE';
   10:MESP:= 'OCTUBRE';
   11:MESP:= 'NOVIEMBRE';
   12:MESP:= 'DICIEMBRE';
```

```
  ELSE
```

```
    BEGIN
```

```
      WRITELN;
```

```
      WRITELN(' *** ERROR FATAL ***');
```

```
      WRITELN;
```

```
      WRITELN(' EL MES ',MMP,2,' NO EXISTE');
```

```
      WRITELN;
```

```
    END
```

```
  END
```

```
END;
```

```
*****
* PROCEDURE LINEA *
*****
```

```
PROCEDURE LINEA;
```

```
BEGIN
```

```
  WRITE(' :12,II:5, ' :3, ' :E1,NOMMED, ' :E2, ' :7,UR:6, ' :17);
```

```
  WRITE('$ ', ' :E3,PRECU:6:2, ' :18, ' :E4,DESCU:2:1, ' %', ' :18, '$ ');
```

```
  WRITELN(' :E5,PNXMED:7:2);
```

```
END;
```

```
*****
* PROCEDURE ESPE3 *
*****
```

```
PROCEDURE ESPE3;
BEGIN
  IF (PRECV>0) AND (PRECV<=9.99) THEN
    E3:=5
  ELSE
    IF (PRECV>=10) AND (PRECV<=99.99) THEN
      E3:=4
    ELSE
      IF (PRECV>=100) AND (PRECV<=999.99) THEN
        E3:=3
      ELSE
        IF (PRECV>=1000) AND (PRECV<=9999.99) THEN
          E3:=2
        ELSE
          IF (PRECV>=10000) AND (PRECV<=99999.99) THEN
            E3:=1
          ELSE
            IF (PRECV>=100000) AND (PRECV<=999999.99) THEN
              E3:=0;
            END;
          END;
        END;
      END;
    END;
  END;
END;
```

```
*****
* PROCEDURE ESPES *
*****
```

```
PROCEDURE ESPES;
BEGIN
  IF (PNXMED>0) AND (PNXMED<=9.99) THEN
    E5:=6
  ELSE
    IF (PNXMED>=10) AND (PNXMED<=99.99) THEN
      E5:=5
    ELSE
      IF (PNXMED>=100) AND (PNXMED<=999.99) THEN
        E5:=3
      ELSE
        IF (PNXMED>=1000) AND (PNXMED<=9999.99) THEN
          E5:=3
        ELSE
          IF (PNXMED>=10000) AND (PNXMED<=99999.99) THEN
            E5:=2
          ELSE
            IF (PNXMED>=100000) AND (PNXMED<=999999.99) THEN
              E5:=1
            ELSE
              IF (PNXMED>=1000000) AND (PNXMED<=9999999.99) THEN
                E5:=0;
              END;
            END;
          END;
        END;
      END;
    END;
  END;
END;
```

```
*****  
* PROCEDURE LISENCAB *  
*****
```

```
PROCEDURE LISENCAB;  
BEGIN  
  WRITELN;  
  WRITELN(' ':42,'PROVEEDOR : ',NOMP);  
  LIC:=1;WRITELN;  
  WRITE(' ':12,'CLAVE',' ':19,'NOMBRE DEL ARTICULO',' ':19);  
  WRITE('NUM DE UNID.',' ':17,'PRECIO X',' ':6,'DESCUENTO');  
  WRITELN(' ':19,'PAGO NETO');  
  WRITELN;  
END;
```

```
*****  
* PROCEDURE FECHDREP *  
*****
```

```
PROCEDURE FECHDREP;  
BEGIN  
  WRITELN;  
  WRITE(' ':29,'REPORTE DE PEDIDOS DE MEDICAMENTOS ');  
  WRITELN('DEL ',DDP:2,' DE ',MESP,' DE 19',AAP:2);  
  WRITELN;WRITELN;  
END;
```

```
*****
* PROCEDURE LISPED *
*****
```

```
PROCEDURE LISPED;
VAR CP, LONGNM, NENM: INTEGER;
    SUMPNT: REAL;
    LISP: CHAR;
```

```
BEGIN
  ASSIGN(PD, 'B:PEDIDOS.FIL');
  RESET(PD);
  IF LIM>0 THEN
    BEGIN
```

```
      IF I1>0 THEN
        BEGIN
          LIE:=0; SUMPNT:=0; I1:=0;
          SEEKREAD(PD, I1);
          ERRCHK;
          IF ERROR THEN EXIT;
          I1:=PD .AP;
          AAP:=PD .CLAVE;
          MMP:=PD .UR;
          DDP:=PD .CP;
          ASIGMES;
          WRITELN; WRITELN;
```

```
        END
      ELSE
```

```
        BEGIN
          WRITELN;
          WRITELN(' *** EL ARCHIVO DE PEDIDOS AUN NO CONTIENE INFORMACION ***');
          WRITELN(' *** FAVOR DE EJECUTAR LA OPCION <A> PARA CARGAR PEDIDOS ***');
          WRITELN;
```

```
        END;
      IF I1<>0 THEN
        BEGIN
```

```
          FECHDREP;
          SEEKREAD(PD, I1);
          ERRCHK;
          IF ERROR THEN EXIT;
          I1:=PD .CLAVE;
          CP:=PD .CP;
          REPEAT
            LISP:=' ';
            SEEKREAD(NM, I1);
            NOMMED:=NM .NOMBRE;
            LONGNM:=LENGTH(NOMMED);
            NENM:=30-LONGNM;
            E1:=NENM DIV 2;
            E2:=NENM MOD 2;
            IF E2=0 THEN E2:=E1
            ELSE E2:=E1+1;
            PRECV:=NM .PRE1*1.0;
            DESCV:=NM .DP1;
```

```

ESPE3;
CASE CP OF
  1:NOMP:='JORGE GONZALEZ';
  2:NOMP:='BERARDO SANCHES';
  3:NOMP:='MAURICIO ESTRADA';
END;
PNXMED:=PD .UR*PRECV-(PRECV*PD .UR)*DESCV;
ESPE5;
SUMPNTP:=PNXMED+SUMPNTP;
DESCV:=100*DESCV;
IF DESCV<10.0 THEN E4:=2
ELSE E4:=1;
UR:=PD .UR;
I1:=PD ,AP;
IF I1<>0 THEN
  BEGIN
    SEEKREAD(PD,I1);
    ERRCHK;
    IF ERROR THEN EXIT;
  END
ELSE
  BEGIN
    CP:=0;
    LISP:='*';
  END;
IF CP=PD .CP THEN
  BEGIN
    IF LIE=0 THEN
      LISENCAB;
      LINEA;
    END
  ELSE
    BEGIN
      IF LIE=0 THEN
        LISENCAB;
        LINEA;
      WRITELN;
      WRITE(' :36,'TOTAL DEL PEDIDO A ESTE PROVEEDOR.- ');
      WRITELN(' :5,'$, '$ ',SUMPNTP:9:2);
      SUMPNTP:=0;LIE:=0;
      WRITELN;WRITELN;
      WRITELN;WRITELN;
      CP:=PD .CP;
    END;
    II:=PD .CLAVE;
  UNTIL LISP='*';
END;
END
ELSE
  BEGIN
    WRITELN('***ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION***');
    WRITELN('***FAVOR DE VOLVER A LISTAR EL MENU Y EJECUTE LA OPCION ***');
    WRITELN('*** <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO ***');
  END;
  CLOSE(PD,I1);
  WRITELN(' *** TERMINA PROGRAMA DE LISTAR PEDIDOS *** ');
END; (*FIN DE LISPED*)

MODEND.

```

V.1.4 LISTADO FUENTE DEL PROGRAMA:

< MOD 3 >

```
*****  
*  
*   PROGRAMA   MOD3   *  
*  
*****
```

MODULE OVERLAY3;

```
TYPE ARCHI=RECORD  
  CLAVE,MS1,MS2,MS3:INTEGER;  
  PRECIO:REAL;  
  UNID,ANN,MES,DIA:INTEGER;  
  CP:INTEGER;  
  IMP,PDU:REAL;  
  UVEN,UPED,UVP1:INTEGER;  
  AA,MM,DD:INTEGER;  
END;
```

```
  ARCHIII=RECORD  
    CLAVE:INTEGER;  
    NOMBRE:STRING(306);  
    R1,R2,R3,R4:INTEGER;  
    PROV1:INTEGER;  
    PRE1,DP1:REAL;  
    PROV2:INTEGER;  
    PRE2,DP2:REAL;  
    PROV3:INTEGER;  
    PRE3,DP3:REAL;  
    DD,MM,AA:INTEGER;  
  END;
```

```
VAR BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)  
    NM:EXTERNAL FILE OF ARCHIII;(*ARCHIVO DE NOMBRES Y ESTADISTICAS*)  
    CVEPRO,CVEMED,UNIDP:EXTERNAL INTEGER;  
    PRECV,DESCV:EXTERNAL REAL;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****  
* PROCEDURE LPED *  
*****
```

```
PROCEDURE LPED;
```

```
BEGIN
```

```
WRITE(' :4,'CANTIDAD DE MEDICAMENTOS',' :9,'ACTUALMENTE ESTE ');
```

```
WRITELN('ARTICULO');
```

```
WRITE(' :3,'QUE SE RECOMIENDAN COMPRAR',' :8,'ES SURTIDO POR ');
```

```
WRITELN('EL PROVEEDOR');
```

```
WRITELN(' :14,BF .UVEN:6,' :30,BF .CP:1);
```

```
CVPRO:=BF .CP;
```

```
CVEMED:=NM .CLAVE;
```

```
UNIDP:=BF .UVEN;
```

```
PRECV:=NM .PRE1;
```

```
DESCV:=NM .DP1;
```

```
END;
```

```
MODEND.
```

V.1.5 LISTADO FUENTE DEL PROGRAMA:

< M O D 4 >

```
*****  
*  
*   PROGRAMA MOD4   *  
*  
*****
```

MODULE OVERLAY4;

```
TYPE ARCHI=RECORD  
  CLAVE,MS1,MS2,MS3:INTEGER;  
  PRECIO:REAL;  
  UNID,ANN,MES,DIA:INTEGER;  
  CP:INTEGER;  
  IMP,PDU:REAL;  
  UVEN,UPED,UVP1:INTEGER;  
  AA,MM,DD:INTEGER;  
END;  
  ARCHIII=RECORD  
  CLAVE:INTEGER;  
  NOMBRE:STRING1306;  
  R1,R2,R3,R4:INTEGER;  
  PROV1:INTEGER;  
  PRE1,DP1:REAL;  
  PROV2:INTEGER;  
  PRE2,DP2:REAL;  
  PROV3:INTEGER;  
  PRE3,DP3:REAL;  
  DD,MM,AA:INTEGER;  
END;
```

```
VAR BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)  
    NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)  
  
ERROR:EXTERNAL BOOLEAN;  
LIM:EXTERNAL INTEGER;  
W1:INTEGER;  
NNO:STRING1306;  
EXTERNAL PROCEDURE ERRCHK;
```

```
*****  
* PROCEDURE CADUCA *  
*****
```

```
PROCEDURE CADUCA;
```

```
VAR ANO,MES,DIA,TOTANNEX,TOTMESEX,TOTDIAEX,TOTANNPE,TOTMESPE: INTEGER;  
TOTDIAPE,I,VER,NUMDIAS,MESESEX,TOTALDEX,LANGNOM,SANGRIA: INTEGER;  
MESESPE,TOTALDPE,CANTMED,TOTANOPE: INTEGER;  
SIG:CHAR;  
NOMMED:STRING(306);
```

```
*****  
* PROCEDURE LETRA *  
*****
```

```
PROCEDURE LETRA;
```

```
BEGIN
```

```
WRITELN;
```

```
WRITE(' ':14,NNO,' ':14,' ':14,' ':24,'FECHA : ',ANO:2);
```

```
WRITELN(' / ',MES:2,' / ',DIA:2);
```

```
WRITELN;
```

```
WRITE(' ':14,'RELACION DE MEDICAMENTOS QUE CADUCARAN ');
```

```
WRITELN(' EN UN PERIODO NO MAYOR DE ',NUMDIAS:3,' DIAS ');
```

```
WRITELN;
```

```
WRITE('=====');
```

```
WRITE('=====');
```

```
WRITELN('=====');
```

```
WRITELN;
```

```
WRITE(' ':41,' UNIDADES ACTUALES :', ' ':1);
```

```
WRITELN(' UNIDADES PEDIDAS');
```

```
WRITE(' ':3,'CLAVE', ' ':2,'. NOMBRE DEL MEDICAMENTO ', ' ':2);
```

```
WRITE('CLAVE DEL :UNIDADES EN :FECHA DE :FALTAN :', ' ':2);
```

```
WRITELN('CLAVE DEL :UNIDADES :FECHA DE :FALTAN');
```

```
WRITE(' ':42,'PROVEEDOR :EXISTENCIA :CADUCIDAD :X DIAS :', ' ':2);
```

```
WRITELN('PROVEEDOR :PEDIDAS :CADUCIDAD :X DIAS');
```

```
WRITELN;
```

```
WRITE('=====');
```

```
WRITELN('=====');
```

```
WRITELN;
```

```
END;
```

```
*****
* PROCEDURE FECHITA *
*****
```

```
PROCEDURE FECHITA;
```

```
PROCEDURE FECHAS;
```

```
VAR AN1,DI2:INTEGER;
```

```
BEGIN
```

```
  IF ((AND>=84)AND(AND<100)AND(MES>0)AND(DIA>0)) THEN
```

```
    BEGIN
```

```
      VER:=1;
```

```
      AN1:=AND MOD 4;
```

```
      CASE MES OF
```

```
        1,3,5,7,8,10,12:DI2:=31;
```

```
        4,6,9,11:DI2:=30;
```

```
        2: BEGIN IF AN1=0 THEN
```

```
            DI2:=29
```

```
          ELSE
```

```
            DI2:=28
```

```
        END
```

```
      ELSE
```

```
        VER:=0;
```

```
      END;
```

```
      IF VER=1 THEN
```

```
        BEGIN
```

```
          IF DIA>DI2 THEN
```

```
            VER:=0;
```

```
          END;
```

```
      END;
```

```
END;
```

```
BEGIN
```

```
SIG:=' ';
```

```
REPEAT
```

```
  AND:=0;MES:=0;DIA:=0;
```

```
  WRITE('AND => ');
```

```
  READLN(AND);
```

```
  WRITE('MES => ');
```

```
  READLN(MES);
```

```
  WRITE('DIA => ');
```

```
  READLN(DIA);
```

```
  VER:=0;
```

```
  FECHAS;
```

```
  IF VER=1 THEN
```

```
    BEGIN
```

```
      SIG:='Q';
```

```
      WRITE('ANOTE LA FECHA CORRECTAMENTE AND ',AND);
```

```
      WRITELN(' ',AND,' MES ',MES,' DIA ',DIA);
```

```
    END
```

```
  ELSE
```

```
    BEGIN
```

```
      WRITELN(' ***** E R R O R ***** ');
```

```
      SIG:=' ';
```

```
      WRITELN('ANOTE LA FECHA NUEVAMENTE ');
```

```
      WRITE('SE DEBEA SALIRSE ANOTE Q ! ');
```

```
      READLN(SIG);
```

```
    END
```

```
UNTIL SIG='Q';
```

```
END;
```

```
*****
* PROCEDURE ASIGNAR *
*****
```

```
PROCEDURE ASIGNAR;
VAR L, LONGN: INTEGER;
BEGIN
  ASSIGN(BF, 'B:MAESTRO.FIL');
  RESET(BF);
  ASSIGN(NM, 'B:NOMYEST.FIL');
  RESET(NM);
  L:=0;
  SEEKREAD(NM, L);
  ERRCHK;
  IF ERROR THEN EXIT;
  NNO:=NM .NOMBRE;
  LONGN:=LENGTH(NNO);
  W1:=30-LONGN;
  CANTMED:=0;
END;
```

```
*****
* PROCEDURE IMPRESO1 *
*****
```

```
PROCEDURE IMPRESO1;
BEGIN
WRITE(BF ,CP:6, ' ':6, BF .UNID:7, ' ':7, BF .ANN:2, '/', BF .MES:2, '/');
WRITE(BF .DIA:2, ' ':2, TOTALDEX:5, ' ':6);
END;
```

```
*****
* PROCEDURE IMPRESO2 *
*****
```

```
PROCEDURE IMPRESO2;
BEGIN
WRITE(NM .PROV:7, ' ':6, BF .UPED:7, ' ':3, BF .AA:2, '/', BF .MM:2, '/');
WRITE(BF .DD:2, ' ':2, TOTALDPE:5);
END;
```

```
*****
* PROCEDURE IMPRESION *
*****
```

```
PROCEDURE IMPRESION;
BEGIN
  SEEKREAD(NM,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  NOMMED:=NM .NOMBRE;
  LONGNOM:=LENGTH(NOMMED);
  SANGRIA:=30-LONGNOM;
  WRITE(' '2,I:5,' '3,NM .NOMBRE,' 'SANGRIA,' '1);
  IF ((BF .UNID>0)AND(TOTALDEX<=NUMDIAS)AND(BF .UPED>0)
    AND(TOTALDPE<=NUMDIAS)) THEN
    BEGIN
      IMPRESO1;
      IMPRESO2;
    END
  ELSE
    BEGIN
      IF ((BF .UNID>0)AND(TOTALDEX<=NUMDIAS)) THEN
        IMPRESO1
      ELSE
        BEGIN
          IF ((BF .UPED>0)AND(TOTALDPE<=NUMDIAS)) THEN
            BEGIN
              WRITE(' '47);
              IMPRESO2;
            END;
          END;
        END;
      END;
    END;
  WRITELN;
  CANTMED:=CANTMED+1;
END;
```

```
*****
* PROCEDURE CALCULOS *
*****
```

```
PROCEDURE CALCULOS;
VAR
  AUXIL1,AUXIL2: INTEGER;
BEGIN
  TOTALDEX:=0;
  TOTALDPE:=0;
  AUXIL1:=BF .ANN*BF .MES*BF .DIA;
```

```

IF AUXIL1>0 THEN
  BEGIN
    TOTANEX:=BF .ANN-ANO;
    TOTMESEX:=BF .MES-MES;
    TOTDIAEX:=BF .DIA-DIA;
    MESESEX:=12*(TOTANEX)+TOTMESEX;
    TOTALDEX:=MESESEX*30+TOTDIAEX
  END;
AUXIL2:=BF .AA*BF .MM*BF .DD;
IF AUXIL2>0 THEN
  BEGIN
    TOTANOPE:=BF .AA-ANO;
    TOTMESPE:=BF .MM-MES;
    TOTDIAPE:=BF .DD-DIA;
    MESESEPE:=12*(TOTANOPE)+TOTMESPE;
    TOTALDPE:=MESESEPE*30+TOTDIAPE
  END;
END;

*****
* PROCEDURE CADUCA1 *
*****

PROCEDURE CADUCA1;
VAR II:INTEGER;
BEGIN
  WRITE('RELACION DE MEDICAMENTOS QUE CADUCARAN EN UN PERIOD NO MAYR');
  WRITELN('DE N DIAS');
  WRITE('ANOTE EL VALOR DE N : ');
  READLN(NUMDIAS);
  LETRA;
  FOR II:=1 TO LIM DO
    BEGIN
      I:=II;
      SEEKREAD(BF,I);
      ERRCHK;
      IF ERROR THEN EXIT;
      IF ((BF .UNID>0)OR(BF .UPED>0)) THEN
        BEGIN
          CALCULOS;
          IF ((TOTALDEX<=NUMDIAS)OR(TOTALDPE<=NUMDIAS)) THEN
            IMPRESION;
          END;
        END;
    WRITELN;
    WRITELN;
    IF CANTMED>0 THEN
      WRITE('EN ',NUMDIAS:3,' DIAS',' CADUCARAN ',CANTMED:4);
      WRITELN(' MEDICAMENTOS. ');
    END;
  END;

```

```
*****  
*          PROCEDURE PRINCIPAL          *  
*****
```

```
BEGIN (*PRINCIPAL*)  
  IF LIM>0 THEN  
    BEGIN  
      ASIGNAR;  
      WRITE('ANOTE LA FECHA DE HOY   AÑO MES DIA   :  ');  
      FECHITA;  
      IF VER=1 THEN  
        CADUCA1;  
    END  
  ELSE  
    BEGIN  
      WRITELN('ERROR EL ARCHIVO MAESTRO NO CONTIENE INFORMACION');  
      WRITELN(' FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA');  
      WRITELN(' LA OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO');  
      WRITELN;  
    END;  
    I:=0;  
    WRITELN;  
    WRITELN;  
  END; (*FIN DE CADUCA*)  
  
MODEND.
```

V.1.6 LISTADO FUENTE DEL PROGRAMA :

< M O D S >

```
*****  
*  
*   PROGRAMA   MODS   *  
*  
*****
```

MODULE OVERLAYS;

TYPE ARCHI=RECORD

```
CLAVE,MS1,MS2,MS3:INTEGER;  
PRECIO:REAL;  
UNID,ANN,MES,DIA:INTEGER;  
CP:INTEGER;  
IMP,PDU:REAL;  
UVEN,UPED,UVP1:INTEGER;  
AA,MM,DD:INTEGER;  
END;
```

ARCHIII=RECORD

```
CLAVE:INTEGER;  
NOMBRE:STRING(306);  
R1,R2,R3,R4:INTEGER;  
PROV1:INTEGER;  
E1,DP1:REAL;  
PROV2:INTEGER;  
PRE2,DP2:REAL;  
PROV3:INTEGER;  
PRE3,DP3:REAL;  
DD,MM,AA:INTEGER;  
END;
```

ARCHV=RECORD

```
CDISP:INTEGER;  
END;
```

VAR BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)

NM:EXTERNAL FILE OF ARCHIII;(*ARCHIVO DE NOMBRES Y ESTADISTICAS*)

NC:EXTERNAL FILE OF ARCHV; (*ARCHIVO DE CLAVES DISPONIBLES*)

I,X1,II,J,LIM:EXTERNAL INTEGER;

ERROR:EXTERNAL BOOLEAN;

CMB:INTEGER;

EXTERNAL PROCEDURE ERRCHK;

```
*****  
* PROCEDURE LIMPBF *  
*****
```

```
PROCEDURE LIMPBF;  
BEGIN  
  I:=CMB;  
  SEEKREAD(BF,I);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
  BF .CLAVE:=CMB;  
  BF .MS1:=0;  
  BF .MS2:=0;  
  BF .MS3:=0;  
  BF .PRECIO:=0.0;  
  BF .UNID:=0;  
  BF .ANN:=0;  
  BF .MES:=0;  
  BF .DIA:=0;  
  BF .CP:=0;  
  BF .IMP:=0.0;  
  BF .PDU:=0.0;  
  BF .UVEN:=0;  
  BF .UPED:=0;  
  BF .UVF1:=0;  
  BF .AA:=0;  
  BF .MM:=0;  
  BF .DD:=0;  
  SEEKWRITE(BF,I);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
END; (* FIN DE CUE DISP *)
```

```
*****  
* PROCEDURE LIMPNM *  
*****
```

```
PROCEDURE LIMPNM;  
BEGIN  
  NM .CLAVE:=CMB;  
  NM .NOHRE:= '  
  NM .R1:=0;  
  NM .R2:=0;  
  NM .R3:=0;  
  NM .R4:=0;
```

```

NM .PROV1:=0;
NM .PRE1:=0.0;
NM .DP1:=0.0;
NM .PROV2:=0;
NM .PRE2:=0.0;
NM .DP2:=0.0;
NM .PROV3:=0;
NM .PRE3:=0.0;
NM .DP3:=0.0;
NM .DD:=0;
NM .MM:=0;
NM .AA:=0;
II:=CMB;
SEEKWRITE(NM,II);
ERRCHK;
IF ERROR THEN EXIT;
END;

```

```

*****
* PROCEDURE LISCLA *
*****

```

```

PROCEDURE LISCLA;
VAR
  NCA,ICD:INTEGER;
  RSN:CHAR;
BEGIN
  WRITELN;
  WRITELN('DESEA LISTAR LAS CLAVES QUE SE ENCUENTRAN ');
  WRITE('DISPONIBLES?, TRANSMITA S)I O N)O => ');
  READ(RSN);WRITELN;
  ASSIGN(NC,'B:CVEDISP.FIL');
  RESET(NC);
  IF RSN IN 'S','s' THEN
    BEGIN
      X1:=0;
      SEEKREAD(NC,X1);
      NCA:=NC .CDISP;
      WRITELN;
      IF NCA>0 THEN
        BEGIN
          WRITELN;WRITELN;
          WRITELN(' :5,'CLAVE DISPONIBLE ');
          FOR ICD:=1 TO NCA DO
            BEGIN
              X1:=ICD;
              SEEKREAD(NC,X1);
              WRITELN;
              WRITELN(' :11,NCN,CDISP:6);
            END;
          END
        ELSE
          WRITELN(' *** NO HAY CLAVES DISPONIBLES ***');
        END;
  END; (* FIN DE LISCLA *)

```

```
*****
* PROCEDURE BAJA *
*****
```

```
PROCEDURE BAJA;
VAR
  REP, TER, SNN: CHAR;
  NCD, J: INTEGER;
BEGIN
  ASSIGN(NC, 'B:CVEDISP.FIL');
  RESET(NC);
  X1:=0;
  IF LIM>0 THEN
    BEGIN
      SEEKREAD(NC, X1);
      ERRCHK;
      IF ERROR THEN EXIT;
      NCD:=NC .CDISP
      END;
  IF LIM>0 THEN
    BEGIN
      REPEAT
        REPEAT
          TER:=' ';
          WRITELN;
          WRITE('DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA =>');
          READLN(CMB);
          IF CMB > 0 THEN
            BEGIN
              IF CMB<=LIM THEN
                BEGIN
                  I:=CMB; II:=CMB; J:=CMB;
                  REP:=' ';
                  FOR J:=1 TO NCD DO
                    BEGIN
                      X1:=J;
                      SEEKREAD(NC, X1);
                      IF NC .CDISP=CMB THEN
                        REP:='*';
                    END;
                  IF REP=' ' THEN
                    BEGIN
                      WRITE('ES CORRECTA LA CLAVE DEL MEDICAMENTO : ');
                      WRITELN(NM, NOMBRE);
                      WRITE(' SE PUEDE PONER DISPONIBLE? ');
                      WRITE(' TRANSMITA S)I O N)O => ');
                      READ(SNN); WRITELN;
                    END;
                END;
            END;
        UNTIL CMB=0;
    END;
END;
```

```

        IF SNN IN 1'S','S'0 THEN
            BEGIN
                NCD:=NCD+1;
                LIMPBF;
                LIMPNN;
                X1:=NCD;
                NC ,CDISP:=CMB;
                SEEKWRITE(NC,X1);
                ERRCHK;
                IF ERROR THEN EXIT;
            END;
        ELSE
            BEGIN
                TER:='*';
                WRITELN;
                WRITELN('CLAVE PUESTA DISPONIBLE YA ANTERIORMENTE');
                WRITELN;
            END;
        ELSE
            BEGIN
                WRITELN;
                WRITELN('ERROR SE EXCEDE DE LA CLAVE MAXIMA EXISTENTE');
                WRITE('LA CLAVE MAXIMA QUE CONTIENE INFORMACION ');
                WRITELN(' ES LA:',LIM:5);
                WRITELN;
                TER:='*';
            END
        ELSE
            BEGIN
                WRITELN;
                WRITELN('ERROR LAS CLAVES INICIAN DESDE LA 0001');
                WRITELN;
                TER:='*';
            END
        UNTIL TER=' ';
        WRITELN;
        WRITE('DESEA DAR DE BAJA OTRO MEDICAMENTO?, ');
        WRITE(' TRANSMITA S:I O N)O => ');
        READLN(SNN);
        IF SNN IN 1'S','S'0 THEN
            TER:='*';
        ELSE
            TER:=' ';
        UNTIL TER=' ';
    END
ELSE
    BEGIN
        WRITELN;
        WRITELN('ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION');
        WRITELN(' FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA');
        WRITELN(' OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO ');
        WRITELN;
    END;

```

```
IF ERROR THEN EXIT;
X1:=0;
NC .CDISP:=NCD;
SEEKWRITE(NC,X1);
ERRCHK;
IF ERROR THEN EXIT;
CLOSE(NC,X1);
IF LIM <> 0 THEN
  LISCLA;
WRITELN;WRITELN;
WRITELN(' *** TERMINA PROGRAMA DE BAJA DE MEDICAMENTOS ***');
END;
```

MODEND.

V.1.7 LISTADO FUENTE DEL PROGRAMA:

< M O D 6 >

```
*****  
*  
*   PROGRAMA   MOD6   *  
*  
*****
```

MODULE OVERLAY6;

```
TYPE ARCHI=RECORD  
  CLAVE,MS1,MS2,MS3:INTEGER;  
  PRECIO:REAL;  
  UNID,ANN,MES,DIA:INTEGER;  
  CP:INTEGER;  
  IMP,PDU:REAL;  
  UVEN,UPED,UVF1:INTEGER;  
  AA,MM,DD:INTEGER;  
END;
```

```
  ARCHIII=RECORD  
    CLAVE:INTEGER;  
    NOMBRE:STRING(306);  
    R1,R2,R3,R4:INTEGER;  
    PROV1:INTEGER;  
    PRE1,DP1:REAL;  
    PROV2:INTEGER;  
    PRE2,DP2:REAL;  
    PROV3:INTEGER;  
    PRE3,DP3:REAL;  
    DD,MM,AA:INTEGER;  
  END;
```

```
VAR BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)  
    NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)  
    LIM,ANIO,MES,DIA:EXTERNAL INTEGER;  
    ERROR:EXTERNAL BOOLEAN;  
    EXTERNAL PROCEDURE ERRCHK;
```

```
*****
* PROCEDURE PORC *
*****
```

```
PROCEDURE PORC(VAR TTT:REAL)
VAR S:INTEGER;
BEGIN
  REPEAT
    S:=1;
    IF TTT>=1 THEN
      BEGIN
        IF ((TTT>=1)AND(TTT<100)) THEN
          TTT:=TTT/100
        ELSE
          BEGIN
            S:=0;
            WRITELN('** E R R O R **');
            TTT:=0;
            WRITE('NOTE EL DATO NUEVAMENTE : ');
            READLN(TTT);
          END;
        END
      UNTIL S=1;
    END;
  END;
```

```
*****
* PROCEDURE WRITERECS *
*****
```

```
PROCEDURE WRITERECS;
VAR I,II,C,M1,M2,M3,U,L,PRO1,PRO2,PRO3,VFE:INTEGER;
Y2,Y3,G1,T,N2,ANO,MES6,DIA6,AH1,MH1,DH1:INTEGER;
P,IM,Y1,PR1,PR2,PR3,D1,D2,D3:REAL;
SIG:CHAR;
NOHB:STRING130;
```

```
*****
* PROCEDURE FECHITA *
*****
```

```
PROCEDURE FECHITA;
VAR VER:INTEGER;
PROCEDURE FECHAS;
VAR AN1,DI2:INTEGER;
BEGIN
  IF ((ANO>=84)AND(ANO<100)AND(MES6>0)AND(DIA6>0)) THEN
    BEGIN
      VER:=1;
      AN1:=ANO MOD 4;
      CASE MES6 OF
        1,3,5,7,8,10,12:DI2:=31;
        4,6,9,11:DI2:=30;
        2: BEGIN IF AN1=0 THEN
              DI2:=29
            ELSE
              DI2:=28   END
        END;
      ELSE
        VER:=0;
      END;
      IF VER=1 THEN
        BEGIN
          IF DIA6>DI2 THEN
            VER:=0;
          END;
        END;
      END;
    END;
  BEGIN
    SIG:=' ';
    REPEAT
      ANO:=0;
      MES6:=0;
      DIA6:=0;
      IF VFE=0 THEN
        BEGIN
          WRITE('DAME ANO => ');
          READLN(ANO);
          WRITE('DAME MES => ');
          READLN(MES6);
          WRITE('DAME DIA => ');
          READLN(DIA6);
        END
      ELSE
        BEGIN
          ANO:=ANIO;
          MES6:=MES;
          DIA6:=DIA
        END;
      VER:=0;FECHAS;
      IF VER=1 THEN SIG:='Q'
      ELSE
        BEGIN
          WRITELN(' ***** E R R O R ***** ');
          SIG:=' ';
          WRITELN('ANOTE LA FECHA NUEVAMENTE : ');
        END
      UNTIL SIG='Q';
    END;
  END;
```

```
*****
* PROCEDURE WRITE2 *
*****
```

```
PROCEDURE WRITE2;
BEGIN
  Writeln;
  Write('CLAVE DE MEDICINA SIMILAR 1 => e ');
  Readln(M1);
  BF .MS1:=M1;
  Writeln;
  Write('CLAVE DE MEDICINA SIMILAR 2 => e ');
  Readln(M2);
  BF .MS2:=M2;
  Writeln;
  Write('CLAVE DE MEDICINA SIMILAR 3 => e ');
  Readln(M3);
  BF .MS3:=M3;
  Writeln;
  Write('EL PRECIO DE VENTA DEL MEDICAMENTO => $ ');
  Readln(P);
  BF .PRECIO:=P;
  Writeln;
  Write('EL NUMERO DE UNIDADES EN EXISTENCIA => ');
  Readln(U);
  BF .UNID:=U;
  Writeln;
  Writeln('FECHA DE CADUCIDAD DEL MEDICAMENTO');
  FECHITA;
  BF .ANN:=ANO;
  BF .MES:=MES6;
  BF .DIA:=DIA6;
  BF .CP:=PRO1;
  Writeln;
  Write('EL IMPUESTO => % ');
  Readln(IM);
  PORC(IM);
  BF .IMP:=IM;
  Writeln;
  Writeln('EL PORCENTAJE DE UNIDADES MINIMAS ');
  Write('QUE DEBEN EXISTIR           => % ');
  Readln(Y1);
  PORC(Y1);
  Writeln;
  BF .PDU:=Y1;
  BF .UVEN:=0;
  BF .UPED:=0;
  BF .UVP1:=0;
  BF .AA:=0;
  BF .MM:=0;
  BF .DD:=0;
  BF .CLAVE:=C;
  Writeln('TERMINO DE GRABAR EL MEDICAMENTO CLAVE e ',C);
  Writeln;
  I:=C;
  SEEKWRITE(BF,I);
  ERRCHK;
END; (* F I N 2*)
```

```
*****
* PROCEDURE FINAL *
*****
```

```
PROCEDURE FINAL;
BEGIN
  I:=0;
  SEEKREAD(BF,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  D1:=BF .IMP;
  D2:=BF .UVP;
  BF .MS1:=N2;
  BF .IMP:=D1;
  BF .PDU:=D2;
  SEEKWRITE(BF,I);
  ERRCHK;
  II:=0;
  SEEKREAD(NM,II);
  ERRCHK;
  IF ERROR THEN EXIT;
  NOMB:=NM .NOMBRE;
  NM .CLAVE:=N2;
  NM .NOMBRE:=NOMB;
  NM .DD:=DIA;
  NM .MM:=MES;
  NM .AA:=ANIO;
  SEEKWRITE(NM,II);
  ERRCHK;
END;
```

```
*****
* PROCEDURE WRITE1 *
*****
```

```
PROCEDURE WRITE1;
BEGIN
  NM .CLAVE:=C;
  WRITELN;
  WRITE('NOMBRE DEL MEDICAMENTO => ');
  READLN(NOMB);
  NM .NOMBRE:=NOMB;
  NM .R1:=0;
  NM .R2:=0;
  NM .R3:=0;
  NM .R4:=0;
  WRITELN;
  WRITE('CLAVE DEL PROVEEDOR PRINCIPAL => ');
  READLN(PRO1);
  NM .PROV1:=PRO1;
  WRITELN;
  WRITE('PRECIO DE COMPRA DEL PROVEEDOR PRINCIPAL => ');
  READLN(PRI);
  NM .PRE1:=PRI;
  WRITELN;
```

```
WRITE('DESCUENTO DE COMPRA DEL PROVEEDOR PRINCIPAL => % ');
READLN(D1);
PORC(D1);
NM .DP1:=D1;
WRITELN;
WRITE('CLAVE DEL SEGUNDO PROVEEDOR => € ');
READLN(PRO2);
NM .PROV2:=PRO2;
WRITELN;
WRITE('PRECIO DE COMPRA DEL SEGUNDO PROVEEDOR => $ ');
READLN(PR2);
NM .PRE2:=PR2;
WRITELN;
WRITE('DESCUENTO DE COMPRA DEL SEGUNDO PROVEEDOR => % ');
READLN(D2);
PORC(D2);
NM .DP2:=D2;
WRITELN;
WRITE('CLAVE DEL TERCER PROVEEDOR => € ');
READLN(PRO3);
NM .PROV3:=PRO3;
WRITELN;
WRITE('PRECIO DE COMPRA DEL TERCER PROVEEDOR => $ ');
READLN(PR3);
NM .PRE3:=PR3;
WRITELN;
WRITE('DESCUENTO DE COMPRA DEL TERCER PROVEEDOR => % ');
READLN(D3);
PORC(D3);
NM .DP3:=D3;
NM .DD:=DIA;
NM .MM:=MES;
NM .AA:=ANIO;
II:=C;
SEEKWRITE(NM,II);
ERRCHK;
END;
```

```
*****
*   PROCEDURE PRINCIPAL   *
*****
```

```
BEGIN
  ASSIGN(BF,'B:MAESTRO.FIL');
  RESET(BF);
  ASSIGN(NM,'B:NOMYEST.FIL');
  RESET(NM);
  VFE:=1;
  FECHITA;
  VFE:=0;
  N2:=LIM;
  SIG:=' ';
  REPEAT
    N2:=N2+1;
    WRITELN('ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION : ');
    WRITELN('-----');
    WRITE('CLAVE DEL MEDICAMENTO => & ');
    READLN(C);
    IF C=N2 THEN
      BEGIN
        WRITE1;
        WRITE2;
      END
    ELSE
      BEGIN
        WRITELN('*ERROR* EL SIGUIENTE REGISTRO DISPONIBLE ES EL ',N2:4);
        N2:=N2-1;
      END;
    WRITELN;
    WRITELN('SI DESEA SEGUIR METIENDO INFORMACION TRANSMITA RETURN');
    WRITE('EN CASO CONTRARIO ANOTE Q => ');
    READLN(SIG);
  UNTIL SIG='Q';
  FINAL;
  I:=N2;
  II:=N2;
  CLOSE(BF,I);
  CLOSE(NM,II);
  WRITELN;
END;

MODEND.
```

V.1.8 LISTADO FUENTE DEL PROGRAMA :

< MOD 7 >

```
*****  
* *  
* PROGRAMA MOD7 *  
* *  
*****
```

MODULE OVERLAY7;

```
TYPE ARCHI=RECORD  
  CLAVE,MS1,MS2,MS3:INTEGER;  
  PRECIO:REAL;  
  UNID,ANN,MES,DIA:INTEGER;  
  CP:INTEGER;  
  IMP,PDU:REAL;  
  UVEN,UPED,UVP1:INTEGER;  
  AA,MM,DD:INTEGER;  
END;
```

VAR BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)

```
I,LIM:EXTERNAL INTEGER;  
ERROR:EXTERNAL BOOLEAN;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****  
* PROCEDURE LISTUNO *  
*****
```

PROCEDURE LISTUNO;

```
BEGIN  
  WRITE(' :14,'FECHA DE', ' :13,'CLAVE DEL IMPUESTO', ' :12,'PORCENTAJE')  
  WRITELN(' :12,'UNIDADES', ' :12,'UNIDADES', ' :12,'FECHA')  
  WRITE(' :13,'SIMILARES', ' :17,'X UNIDAD', ' :12))  
  WRITE('UNIDADES', ' :13,'CADUCIDAD')  
  WRITE(' :12,'VENDEDOR', ' :15,'X', ' :15,'MIN DE UNI.', ' :12,'VENDIDAS')  
  WRITELN(' :12,'PEDIDAS', ' :13,'CAD/PED', ' :11))  
  WRITELN)  
END; (*FIN DE LISTUNO*)
```

```
*****
* PROCEDURE LISTDOS *
*****
```

```
PROCEDURE LISTDOS;
```

```
VAR E,E1:INTEGER;
    IMP,PMUE:REAL;
```

```
BEGIN
```

```
IMP:=BF .IMP*100;PMUE:=BF .PDU*100;
```

```
IF IMP<10.0 THEN
```

```
    E:=7
```

```
ELSE
```

```
    E:=6;
```

```
IF PMUE<10.0 THEN
```

```
    E1:=6
```

```
ELSE
```

```
    E1:=5;
```

```
WRITE(' '2,BF .CLAVE:5,' '1,BF .MS1:5,'/',BF .MS2:5,'/',BF .MS3:5)
```

```
WRITE(' '4,BF .PRECIO:7:2,' '3,BF .UNID:6,' '5);
```

```
WRITE(BF .DIA:2,'/',BF .MES:2);
```

```
WRITE('/',BF .ANN:2,' '6,BF .CP:2,' 'E,IMP:2:1);
```

```
WRITE(' X',' 'E1,PMUE:2:1,' X');
```

```
WRITE(' '4,BF .UVEN:6,' '14,BF .UPED:6,' '3,BF .DD:2);
```

```
WRITELN('/',BF .MM:2,'/',BF .AA:2,' '2);
```

```
END; (*FIN DE LISTDOS*)
```

```
*****
* PROCEDURE UNO *
*****
```

```
PROCEDURE UNO;
```

```
VAR SIG,BAND:CHAR;
```

```
BEGIN
```

```
SIG:=' ';
```

```
REPEAT
```

```
    WRITE('TRANSMITA LA CLAVE DEL MEDICAMENTO, QUE DESEA CONSULTAR => ')
```

```
    BAND:=' ';
```

```
    REPEAT
```

```
        READLN(I);
```

```
        IF (I>=0) AND (I<=LIM) THEN
```

```
            BEGIN
```

```
                SEEKREAD(BF,I);
```

```
                ERRCHK;
```

```
                IF ERROR THEN EXIT;
```

```
                BAND:='*';
```

```
            END
```

```
END
```

```

ELSE
  BEGIN
    WRITELN('EL NUMERO DE LA CLAVE EXCEDE EL RANGO PERMITIDO');
    WRITE('LA CLAVE MAXIMA QUE CONTIENE INFORMACION ');
    WRITELN('ES : ',LIM:5);
    WRITE('FAVOR DE VOLVER A TRANSMITIR LA CLAVE => ');
    END;
  UNTIL BAND='*';
  LISTUNO;
  LISTDOS;
  WRITELN;
  WRITELN('DESEA CONSULTAR LA INFORMACION DE OTRO MEDICAMENTO?, ');
  WRITE('TRANSMITA S)I O N)O => ');
  READLN(SIG);
  UNTIL SIG='N';

END;  (* FIN DE UNO *)

```

```

*****
* PROCEDURE RANGO *
*****

```

```
PROCEDURE RANGO;
```

```
VAR CVEI,CVEF,J:INTEGER;
    BAND1,BAND2:CHAR;
```

```
BEGIN
```

```
  BAND1:= ' ';
```

```
  WHILE (BAND1=' ') DO
```

```
    BEGIN
```

```
      WRITELN;
```

```
      WRITE('DEME CLAVE DE INICIO => ');
```

```
      READLN(CVEI);
```

```
      IF CVEI<=0 THEN
```

```
        WRITELN('ERROR: LAS CLAVES INICIAN DESDE LA 00001')
```

```
      ELSE
```

```
        IF CVEI>LIM THEN
```

```
          BEGIN
```

```
            WRITELN('ERROR : SE HA EXCEDIDO DEL RANGO PERMITIDO');
```

```
            WRITE(' EL NUMERO DE');
```

```
            WRITE('DE LA CLAVE MAXIMA QUE CONTIENE INFORMACION ');
```

```
            WRITELN('ES : ',LIM:5);
```

```
          END
```

```
        ELSE
```

```
          BAND1:= '*';
```

```
      END;
```

```
  BAND1:= ' ';
```

```

WHILE (BAND1=' ') DO
  BEGIN
    WRITE('DEME CLAVE FINAL => ');
    READLN(CVEF);
    IF CVEF>=CVEI THEN
      IF CVEF<=LIM THEN
        BAND1:='*'
      ELSE
        BEGIN
          WRITELN('ERROR : SE HA EXCEDIDO DEL RANGO PERMITIDO');
          WRITELN(', EL NUMERO DE');
          WRITE('LA CLAVE MAXIMA QUE CONTIENE INFORMACION ');
          WRITELN('ES : ',LIN:5);
        END
      ELSE
        WRITELN('ERROR: TRANSMITA UNA CLAVE MAYOR O IGUAL A LA INICIAL')
    END;
    I:=CVEI-1;
    REPEAT
      LISTUNO;
      J:=0;
      BAND2:=' ';
      REPEAT
        I:=I+1;
        J:=J+1;
        IF I<=CVEF THEN
          BEGIN
            SEEKREAD(BF,I);
            ERRCHK;
            IF ERROR THEN EXIT;
            LISTDOS;
            WRITELN;
          END
        ELSE
          BAND2:='*';
      UNTIL ((J=20) OR (BAND2='*'));
    UNTIL (BAND2='*');
  END; (* FIN DE RANGO *)

```

```
*****
* PROCEDURE TODO *
*****
```

```
PROCEDURE TODO;
```

```
VAR J:INTEGER;
    BAND3:CHAR;
BEGIN
  I:=0;
  REPEAT
    LISTUND;
    J:=0;
    BAND3:= ' ';
    REPEAT
      I:=I+1;
      J:=J+1;
      IF I<=LIM THEN
        BEGIN
          SEEKREAD(BF,I);
          ERRCHK;
          IF ERROR THEN EXIT;
          LISTDOS;
          WRITELN;
        END
      ELSE
        BAND3:='*';
    UNTIL (J=20) OR (BAND3='*');
  UNTIL (BAND3='*');
END; (*FIN DE TODO*)
```

```
*****
* PROCEDURE LISTAR *
*****
```

```
PROCEDURE LISTAR;
```

```
VAR TIPO:CHAR;
BEGIN
  IF LIM <> 0 THEN
    BEGIN
      REPEAT
        REPEAT
          WRITELN;
          WRITELN('COMO DESEA LISTAR SU INFORMACION ?');
          WRITELN;
          WRITE('DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO => ');
          READLN(TIPO);
          WRITELN;
        UNTIL TIPO IN ('N','T','R');
      UNTIL TIPO IN ('N','T','R');
    END;
  END;
END;
```

```

CASE TIPO OF
  'U','u':UNO;
  'T','t':TODO;
  'R','r':RANGO;
ELSE
  BEGIN
    TIPO:='*';
    WRITELN(' E R R O R ');
    WRITELN('TRANSMITA NUEVAMENTE U, T o R ');
    END
  END;
UNTIL TIPO <> '*';
TIPO:=' ';
WRITELN;WRITELN;
WRITELN('DESEA CONTINUAR LISTANDO INFORMACION ?');
WRITE('TRANSMITA S I O N O => ');
READLN(TIPO);
IF TIPO IN 'N','n' THEN
  TIPO:='*';
UNTIL TIPO='*';
END
ELSE
  BEGIN
    WRITELN('ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION ');
    WRITELN(' FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA ');
    WRITELN(' OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO ');
    END;
WRITELN;WRITELN;
WRITELN(' *** TERMINA PROGRAMA DE LISTAR INFORMACION *** ');
END; (*FIN DEL PROCEDURE LISTAR*)

MODEND.

```

V.1.9 LISTADO FUENTE DEL PROGRAMA:

< M O D B >

```
*****
*                                     *
*   PROGRAMA   MODB   *
*                                     *
*****
```

MODULE OVERLAYB:

```
TYPE ARCHI=RECORD
  CLAVE,MS1,MS2,MS3:INTEGER;
  PRECIO:REAL;
  UNID,ANN,MES,DIA:INTEGER;
  CP:INTEGER;
  IMP,PDU:REAL;
  UVEN,UPED,UVP1:INTEGER;
  AA,MM,DD:INTEGER;
  END;
  ARCHIII=RECORD
    CLAVE:INTEGER;
    NOMBRE:STRING1306;
    R1,R2,R3,R4:INTEGER;
    PROV1:INTEGER;
    PRE1,DP1:REAL;
    PROV2:INTEGER;
    PRE2,DP2:REAL;
    PROV3:INTEGER;
    PRE3,DP3:REAL;
    DD,MM,AA:INTEGER;
  END;
  ARCHV=RECORD
    CDISP:INTEGER;
  END;

VAR  BF:EXTERNAL FILE OF ARCHI;  (*ARCHIVO MAESTRO*)
      NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)
      NC:EXTERNAL FILE OF ARCHV;  (*ARCHIVO DE CLAVES DISPONIBLES*)

      ERROR:EXTERNAL BOOLEAN;
      LIM:EXTERNAL INTEGER;
      X,I2,M11,M22,M33,CV,XZ,CONT,Z1,CVE,CVE:INTEGER;
      SIGZ,SIGA,SS:CHAR;
      PP,CL1,CL2:REAL;
      NOAL:STRING1306;

EXTERNAL PROCEDURE ERRCHK;
```

 * PROCEDURE ACTUA1 *

```

PROCEDURE ACTUA1;
BEGIN
Writeln(' :5,'LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES:');
Writeln;
Writeln('   C   A   M   P   O',' :6,'   C   A   M   P   O');
Writeln(' é   N   O   M   B   R   E',' :9,' é   N   O   M   B   R   E');
Writeln('-----');
Writeln(' 1) MEDICINA SIMILAR 1 ',' :8,'14) NOMBRE DEL PRODUCTO');
Writeln(' 2) MEDICINA SIMILAR 2 ',' :8,'15) UNIDADES DEL RANGO 1');
Writeln(' 3) MEDICINA SIMILAR 3 ',' :8,'16) UNIDADES DEL RANGO 2');
Writeln(' 4) PRECIO                ',' :8,'17) UNIDADES DEL RANGO 3');
Write(' 5) UNIDADES EN EXISTENCIA',' :5);
Writeln('18) UNIDADES DEL RANGO 4');
Write(' 6) PRIMERA CADUCIDAD  ',' :8,'19) CLAVE DEL PROVEEDOR ');
Writeln(' PRINCIPAL');
Write('   AND MES   DIA',' :12,'20) PRECIO DEL PROVEEDOR ');
Writeln(' PRINCIPAL');
Write(' 7) PROVEEDOR PRINCIPAL',' :8);
Writeln('21) DESCUENTO DEL PROVEEDOR PRINCIPAL');
Write(' 8) IMPUESTO                ',' :8,'22) CLAVE DEL SEGUNDO ');
Writeln(' PROVEEDOR');
Write(' 9) PORCENTAJE MINIMO DE UNID',' :2);
Writeln('23) PRECIO DEL SEGUNDO PROVEEDOR');
Write('10) UNIDADES VENDIDAS ',' :9,'24) DESCUENTO DEL SEGUNDO ');
Writeln(' PROVEEDOR ');
Write('11) UNIDADES PEDIDAS      ',' :15,'25) CLAVE DEL TERCER ');
Writeln(' PROVEEDOR');
Write('12) UNID. VEND. DEL PEDIDO',' :15);
Writeln('26) PRECIO DEL TERCER PROVEEDOR');
Write('13) SEGUNDA CADUCIDAD  ',' :8,'27) DESCUENTO DEL TERCER ');
Writeln(' PROVEEDOR ');
Write('   AND MES   DIA',' :12,'28) FECHA DEL MOVIMIENTO');
END;
  
```

```
*****
* PROCEDURE CLAVBA *
*****
```

```
PROCEDURE CLAVBA;
VAR JJ:INTEGER;
BEGIN
  XZ:=0;
  SEEKREAD(NC,XZ);
  ERRCHK;
  IF ERROR THEN EXIT;
  CONT:=NC ,CDISP;
  SS:= ' ' ;
  Z1:=I2;
  FOR JJ:=1 TO CONT DO
    BEGIN
      XZ:=JJ;
      SEEKREAD(NC,XZ);
      ERRCHK;
      IF ERROR THEN EXIT;
      CVE:=NC ,CDISP;
      IF SS='*' THEN
        BEGIN
          XZ:=JJ-1;
          SEEKREAD(NC,XZ);
          ERRCHK;
          IF ERROR THEN EXIT;
          NC ,CDISP:=CVE;
          SEEKWRITE(NC,XZ);
          ERRCHK;
          IF ERROR THEN EXIT;
        END;
      IF NC ,CDISP=Z1 THEN
        SS:='*';
    END;
  IF SS='*' THEN
    CONT:=CONT-1;
  XZ:=0;
  SEEKREAD(NC,XZ);
  ERRCHK;
  IF ERROR THEN EXIT;
  NC ,CDISP:=CONT;
  SEEKWRITE(NC,XZ);
  ERRCHK;
  IF ERROR THEN EXIT;
END;
```

```

*****
* PROCEDURE CEROS *
*****

PROCEDURE CEROS;
BEGIN (*INICIA CEROS*)
  M11:=0;
  M22:=0;
  M33:=0;
  PP:=0;
END; (*FIN DE CEROS*)

```

```

*****
* PROCEDURE ACTUA4 *
*****

PROCEDURE ACTUA4;
BEGIN (*INICIA ACTUA4*)
  READLN(SIGA);
  IF SIGA='S' THEN
    BEGIN
      IF X<14 THEN
        BEGIN
          SEEKWRITE(BF,I2);
          ERRCHK;
        END
      ELSE
        BEGIN
          SEEKWRITE(NM,I2);
          ERRCHK;
        END;
      IF ERROR THEN EXIT;
    END;
  END;
END; (*FIN DE ACTUA4*)

```

```
*****  
* PROCEDURE ACTUA6 *  
*****
```

```
PROCEDURE ACTUA6;
```

```
  BEGIN
```

```
    CASE X OF
```

```
      21:BEGIN WRITE(NM ,DP1:6:2,' ANOTE EL NUEVO DATO : ');  
              READLN(PP);  
              NM ,DP1:=PP  END;
```

```
      22:BEGIN WRITE(NM ,PROV2:5,' ANOTE EL NUEVO DATO : ');  
              READLN(M11);
```

```
              NM ,PROV2:=M11 END;
```

```
      23:BEGIN WRITE(NMR,PRE2:10:2,' ANOTE EL NUEVO DATO : ');  
              READLN(PP);
```

```
              NM ,PRE2:=PP  END;
```

```
      24:BEGIN WRITE(NM ,DP2:6:2,' ANOTE EL NUEVO DATO : ');  
              READLN(PP);
```

```
              NM ,DP2:=PP  END;
```

```
      25:BEGIN WRITE(NM ,PROV3:5,' ANOTE EL NUEVO DATO : ');  
              READLN(M11);
```

```
              NM ,PROV3:=M11 END;
```

```
      26:BEGIN WRITE(NMR,PRE3:10:2,' ANOTE EL NUEVO DATO : ');  
              READLN(PP);
```

```
              NM ,PRE3:=PP  END;
```

```
      27:BEGIN WRITE(NM ,DP3:6:2,' ANOTE EL NUEVO DATO : ');  
              READLN(PP);
```

```
              NM ,DP3:=PP  END;
```

```
      28:BEGIN WRITE('ANO: ',NM ,AA:4,'MES: ',NM ,MM:4,'DIA: ',NM ,DD:4);  
              WRITE(' ANOTE EL NUEVO DATO : ');
```

```
              READLN(M11,M22,M33);
```

```
              NM ,AA:=M11;
```

```
              NM ,MM:=M22;
```

```
              NM ,DD:=M33 END;
```

```
  END;
```

```
END; (*FIN DE ACTUA6*)
```

```
*****
* PROCEDURE ACTUAS *
*****
```

```
PROCEDURE ACTUAS;
BEGIN (*INICIA ACTUAS*)
IF X<21 THEN
  BEGIN
  CASE X OF
    14:BEGIN WRITE(NM .NOMBRE:30,' ANOTE EL NUEVO DATO : ');
             READLN(NOAL);
             NM .NOMBRE:=NOAL END;
    15:BEGIN WRITE(NM .R1:5,' ANOTE EL NUEVO DATO : ');
             READLN(M11);
             NM .R1:=M11 END;
    16:BEGIN WRITE(NM .R2:5,' ANOTE EL NUEVO DATO : ');
             READLN(M11);
             NM .R2:=M11 END;
    17:BEGIN WRITE(NM .R3:5,' ANOTE EL NUEVO DATO : ');
             READLN(M11);
             NM .R3:=M11 END;
    18:BEGIN WRITE(NM .R4:5,' ANOTE EL NUEVO DATO : ');
             READLN(M11);
             NM .R4:=M11 END;
    19:BEGIN IF NM .PROV1=0 THEN
             COVE:=COVE+1;
             WRITE(NM .PROV1:5,' ANOTE EL NUEVO DATO : ');
             READLN(M11);
             NM .PROV1:=M11;
             BF .CP:=M11 END;
    20:BEGIN WRITE(NM .PRE1:10:2,' ANOTE EL NUEVO DATO : ');
             READLN(PP);
             NM .PRE1:=PP END;
  END;
  END
ELSE
  ACTUA6;
END) (*FIN DE ACTUAS*)
```

```
*****
* PROCEDURE ACTUA9 *
*****
```

```
PROCEDURE ACTUA9;
BEGIN
  IF X<14 THEN
    BEGIN
      CASE X OF
        7:BEGIN WRITE(BF .CP:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .CP:=M11;
              NM .PROV1:=M11 END;
        8:BEGIN WRITE(BF .IMP:3:2,' ANOTE EL NUEVO DATO : ');
              READLN(PP);
              BF .IMP:=PP END;
        9:BEGIN WRITE(BF .PDU:3:2,' ANOTE EL NUEVO DATO : ');
              READLN(PP);
              BF .PDU:=PP END;
        10:BEGIN WRITE(BF .UVEN:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .UVEN:=M11 END;
        11:BEGIN WRITE(BF .UPED:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .UPED:=M11 END;
        12:BEGIN WRITE(BF .UVP:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .UVP:=M11 END;
        13:BEGIN WRITE('ANO: ',BF .AA:4,' MES: ',BF .MM:4,' DIA: ');
              WRITELN(BF .DD:4);
              WRITE(' ANOTE EL NUEVO DATO ANO MES DIA : ');
              READLN(M11,M22,M33);
              BF .AA:=M11;
              BF .MM:=M22;
              BF .DD:=M33 END;
      END;
    END
  ELSE
    ACTUA5;
END;
```

```
*****
* PROCEDURE ACTUA3 *
*****
```

```
PROCEDURE ACTUA3;
BEGIN (*INICIA ACTUA3*)
  WRITE(' :8,'DATO VIEJO : ');
  IF X<7 THEN
    BEGIN
      CASE X OF
        1:BEGIN WRITE(BF .MS1:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .MS1:=M11 END;
        2:BEGIN WRITE(BF .MS2:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .MS2:=M11 END;
        3:BEGIN WRITE(BF .MS3:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .MS3:=M11 END;
        4:BEGIN IF BF .PRECIO=0 THEN
              COVE:=COVE+1;
              WRITE(BF .PRECIO:10:2,' ANOTE EL NUEVO DATO : ');
              READLN(PP);
              BF .PRECIO:=PP END;
        5:BEGIN WRITE(BF .UNID:5,' ANOTE EL NUEVO DATO : ');
              READLN(M11);
              BF .UNID:=M11 END;
        6:BEGIN WRITE('ANO: ',BF .ANN:4,' MES: ',BF .MES:4,' DIA: ');
              WRITELN(BF .DIA);
              WRITE(' ANOTE EL NUEVO DATO ANO MES DIA : ');
              READLN(M11,M22,M33);
              BF .ANN:=M11;
              BF .MES:=M22;
              BF .DIA:=M33 END;
      END;
    END
  ELSE
    ACTUA9;
  WRITELN;
  WRITE(' ESTA CORRECTO ? ANOTE S)I O N)O : ');
  ACTUA4;
END; (*DE ACTUA3*)
```

```
*****
*      PROCEDURE ACTUAL      *
*****
```

```
PROCEDURE ACTUAL;
BEGIN
  IF LIM>0 THEN
    BEGIN
      ASSIGN(BF,'B:MAESTRO.FIL');
      RESET(BF);
      ASSIGN(NM,'B:NOMYEST.FIL');
      RESET(NM);
      ASSIGN(NC,'B:CVEDISP.FIL');
      RESET(NC);
      WRITE('ANOTE LA PRIMERA CONTRASENA : ');
      READLN(CL1);
      WRITE('ANOTE LA SEGUNDA CONTRASENA : ');
      READLN(CL2);
      I2:=0;
      SEEKREAD(BF,I2);
      ERRCHK;
      IF ERROR THEN EXIT;
      IF ((CL1=BF .IMP)AND(CL2=BF .UVP1)) THEN
        BEGIN
          SIGZ:=' ';
          REPEAT
            ACTUA1;
            WRITELN;
            WRITE('CUAL REGISTRO QUIERE MODIFICAR? ANOTE EL NUMERO : ');
            READLN(I2);
            IF ((I2>0)AND(I2<=LIM)) THEN
              BEGIN
                COVE:=0;
                SEEKREAD(BF,I2);
                ERRCHK;
                SEEKREAD(NM,I2);
                ERRCHK;
                IF ERROR THEN EXIT;
                SIGA:='N';
                REPEAT
                  CEROS;
                  WRITE('CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : ');
                  READLN(X);
```

```

IF ((X>0)AND(X<=28)) THEN
  ACTUA3
ELSE
  BEGIN
    WRITE('NO EXISTE, EL CAMPO 'X,' ESTA FUERA DEL ');
    WRITELN(' RANGO');
    WRITELN;
    END;
  WRITE('DESEA CAMBIAR OTRO CAMPO DEL REGISTRO ',I2,');
  WRITE(' ? ANOTE S)I O N)O : ');
  READLN(SIGA)
  UNTIL SIGA='N';
  IF COVE=2 THEN
    CLAVBA;
  END
ELSE
  BEGIN
    WRITE('NO EXISTE, EL REGISTRO ',I2,' ESTA FUERA DEL ');
    WRITELN(' RANGO ('LIM,')');
    END;
  WRITELN('SI DESEA SEGUIR ACTUALIZANDO TRANSMITA RETURN');
  WRITE('EN CASO CONTRARIO ANOTE Q');
  READLN(SIGZ)
  UNTIL SIGZ='Q';
  I2:=LIM;
  CLOSE(BF,I2);
  CLOSE(NM,I2);
  I2:=CONT;
  CLOSE(NC,I2);
  I2:=0;
  END
ELSE
  WRITELN('USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES');
  END
ELSE
  BEGIN
    WRITELN(' ERROR EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
    WRITELN(' :11,'OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO')
    END;
  END;
MODEND.

```

V.1.10 LISTADO FUENTE DEL PROGRAMA :

< M O D 9 >

```
*****  
*  
*   PROGRAMA   MOD9   *  
*  
*****
```

MODULE OVERLAY9;

```
TYPE ARCHIII=RECORD  
  CLAVE:INTEGER;  
  NOMBRE:STRING130;  
  R1,R2,R3,R4:INTEGER;  
  PROV1:INTEGER;  
  PRE1,DP1:REAL;  
  PROV2:INTEGER;  
  PRE2,DP2:REAL;  
  PROV3:INTEGER;  
  PRE3,DP3:REAL;  
  DD,MM,AA:INTEGER;  
  END;
```

VAR NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)

```
  I,I1,II,LIM: EXTERNAL INTEGER;  
  ERROR:EXTERNAL BOOLEAN;  
  UNIDP,CVEPRO:EXTERNAL INTEGER;  
  PRECV,DESCV:EXTERNAL REAL;  
  BAND3,RSN:CHAR;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****  
* PROCEDURE DNUEV *  
*****
```

```
PROCEDURE DNUEV;  
BEGIN  
  WRITELN('ERROR TRANSMITA UNICAMENTE <8> O <N> ');  
  WRITE('FAVOR DE VOLVER A TRANSMITIR S)I O N)O => ');  
  BAND3:='*';  
END;
```

```
*****
* PROCEDURE DAMEU *
*****
```

```
PROCEDURE DAMEU;
BEGIN
WRITE('CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO => ');
REPEAT
  BAND3:=' ';
  READ(UNIDP);WRITELN;
  IF UNIDP>32000 THEN
    BEGIN
      WRITELN('CANTIDAD MAXIMA A PEDIR 32000 UNIDADES ');
      WRITE('FAVOR DE VOLVER A TRANSMITIR LA CANTIDAD => ');
      BAND3:='*';
    END
  UNTIL BAND3=' ';
WRITELN;
END;
```

```
*****
* PROCEDURE DAMED *
*****
```

```
PROCEDURE DAMED;
BEGIN
WRITE('DEME EL DESCUENTO QUE LE HACE EL PROVEEDOR => ');
REPEAT
  BAND3:=' ';
  READ(DESCV);WRITELN;
  IF DESCV>1.0 THEN
    DESCV:=DESCV/100.0;
  IF DESCV>100.0 THEN
    BEGIN
      WRITELN('ERROR EL DATO ESTA EQUIVOCADO REBASA EL 100 % ');
      WRITE('FAVOR DE DAR EL DESCUENTO => ');
      BAND3:='*';
    END
  UNTIL BAND3=' ';
END;
```

```
*****
* PROCEDURE DATODO *
*****
```

```
PROCEDURE DATODO;
BEGIN
DAMEU;
WRITE('DEME PRECIO POR UNIDAD => ');
READ(PRECV);WRITELN;
DAMED;
NM .PRE1:=PRECV;
NMX.DP1:=DESCV;
SEEKWRITE(NM,1);
ERRCHK;
IF ERROR THEN EXIT;
END;
```

```
*****
* PROCEDURE DAUUA *
*****
```

```
PROCEDURE DAUUA;
BEGIN
  WRITELN('DESEA DAR EL NUMERO DE MEDICAMENTOS A COMPRAR ');
  WRITE('TRANSMITA, S)I O N)O => ');
  REPEAT
    BAND3:= ' ';
    READ(RSN);WRITELN;
    CASE RSN OF
      'S','s':DAMEU;
      'N','n':WRITELN('SE TOMA LA CANTIDAD SUGERIDA');
    ELSE
      DNUEV;
    END
  UNTIL BAND3=' ';
  WRITELN;
  WRITE('DESEA DAR EL PRECIO X MEDICAMENTO, TRANSMITA S)I O N)O => ');
  REPEAT
    BAND3:= ' ';
    READ(RSN);WRITELN;
    CASE RSN OF
      'S','s':BEGIN
        WRITE('DEME PRECIO POR UNIDAD => ');
        READ(PRECV);WRITELN;
        NM .PRE1:=PRECV;
      END;
      'N','n':WRITELN('SE TOMA EL PRECIO DEL PROVEEDOR ESCOGIDO');
    ELSE
      DNUEV;
    END
  UNTIL BAND3=' ';
  WRITELN;
  WRITE('DESEA DAR EL DESCUENTO DEL PROVEEDOR,TRANSMITA S)I O N)O => ');
  REPEAT
    BAND3:= ' ';
    READ(RSN);WRITELN;
    CASE RSN OF
      'S','s':BEGIN
        DAMED;
        NM .DP1:=DESCV;
      END;
      'N','n':WRITELN('SE TOMA EL DESC. DEL PROVEEDOR ESCOGIDO');
    ELSE
      DNUEV;
    END
  UNTIL BAND3=' ';
  SEEKWRITE(NM,II);
  ERRCHK;
  IF ERROR THEN EXIT;
END;
```

```
*****
*      PROCEDURE DAINF      *
*****
```

```
PROCEDURE DAINF;
BEGIN
  WRITELN('A QUE PROVEEDOR DESEA HACER EL PEDIDO? ');
  WRITE('DEME NUMERO DE PROVEEDOR => ');
  REPEAT
    BAND3:= ' ';
    READ(CVEPRO);WRITELN;
    CASE CVEPRO OF
      1,2,3;BAND3:=' ';
    ELSE
      BEGIN
        WRITELN('SOLO NOS PROVEE EL 1,2 Y 3 TRANSMITA ');
        WRITE('DE NUEVO EL NUMERO DEL PROVEEDOR => ');
        BAND3:='*';
      END;
    END
  UNTIL BAND3=' ';
  WRITELN;
  WRITE('DESEA DAR TODOS LOS DATOS DEL PEDIDO,TRANSMITA S)I O N)O =>');
  REPEAT
    BAND3:= ' ';
    READ(RSN);WRITELN;
    CASE RSN OF
      'S','s';DATODO;
      'N','n';DAUAU;
    ELSE
      DNUEV;
    END
  UNTIL BAND3=' ';
END;

MODEND.
```

V.1.11 LISTADO FUENTE DEL PROGRAMA :

< M O D A >

```
*****  
*  
*   PROGRAMA   MODA   *  
*  
*****
```

MODULE OVERLAY10;

```
TYPE ARCHIII=RECORD  
  CLAVE:INTEGER;  
  NOMBRE:STRING(306);  
  R1,R2,R3,R4:INTEGER;  
  PROV1:INTEGER;  
  PRE1,DP1:REAL;  
  PROV2:INTEGER;  
  PRE2,DP2:REAL;  
  PROV3:INTEGER;  
  PRE3,DP3:REAL;  
  DD,MM,AA:INTEGER;  
  END;
```

```
VAR NM:EXTERNAL FILE OF ARCHIII>(*ARCHIVO DE NOMBRES Y ESTADISTICAS*);  
  II:EXTERNAL INTEGER;  
  ERROR:EXTERNAL BOOLEAN;  
  ANIO,MES,DIA:EXTERNAL INTEGER;  
  DIFAA,DIFMM,DIFDD,DIFTA:INTEGER;  
  BANDZ:CHAR;  
  EXTERNAL PROCEDURE ERRCHK;
```

```
*****
* PROCEDURE VAAN *
*****
```

```
PROCEDURE VAAN;
```

```
BEGIN
  WRITELN('      *** BUENAS NOCHES      ');
  WRITELN;
  WRITELN;
  WRITELN('** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE**');
  WRITELN;
  WRITELN('      **  ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **');
  WRITELN;
  WRITELN;
  WRITE('  ** FECHA DEL ULTIMO ACCESO, DIA:',NM ,DD:2,' MES:');
  WRITELN(NM .MM:2,' AÑO:',NM .AA:2,' ** ');
  WRITELN;
  WRITELN('  ** A CONTINUACION SE LE PIDE LA FECHA DE HOY **');
  WRITELN;
  REPEAT
    BANDZ:=' ';
    WRITE('DEME AÑO => ');
    READ(ANIO);WRITELN;
    DIFAA:=ANIO-NM .AA;
    IF (DIFAA<0) OR (DIFAA>1) THEN
      BEGIN
        WRITE('ERROR : EL AÑO DEBE SER MAYOR O IGUAL ');
        WRITELN('AL DE LA FECHA DE LA');
        WRITE(' :8,'ULTIMA CONSULTA AL SUBSISTEMA ');
        WRITELN(' Y NO MAYOR A UN AÑO ');
        WRITE('DE DIFERENCIA, FAVOR DE VOLVER A ');
        WRITELN('TRANSMITIR EL AÑO ');
        WRITELN;
        BANDZ:='*';
      END
    ELSE
      IF DIFAA=1 THEN
        IF NM .MM < 12 THEN
          BEGIN
            WRITE('ERROR : FAVOR DE DAR EL AÑO IGUAL AL ');
            WRITELN(' DE LA ULTIMA CONSULTA ');
            WRITE(' :8,'DEBIDO A QUE LA DIFERENCIA DE MESES ');
            WRITELN(' SERA MAYOR A DOS ');
            WRITELN;
            BANDZ:='*';
          END;
        UNTIL BANDZ=' ';
      END;
  END;
```

```
*****
* PROCEDURE VMES *
*****
```

```
PROCEDURE VMES;
BEGIN
```

```
  REPEAT
```

```
    BANDZ:=' ';
```

```
    WRITE('DEME MES => ');
```

```
    READ(MES);WRITELN;
```

```
    IF (MES>0) AND (MES<=12) THEN
```

```
      BEGIN
```

```
        DIFMM:=MES-NM .MM;
```

```
        IF DIFAA=0 THEN
```

```
          IF (DIFMM<0) OR (DIFMM>2) THEN
```

```
            BEGIN
```

```
              WRITE('ERROR : EL MES DEBE SER MAYOR O ');
```

```
              WRITELN(' IGUAL AL DE LA FECHA DE LA');
```

```
              WRITE(' :8,'ULTIMA CONSULTA AL SUBSISTEMA Y NO ');
```

```
              WRITELN(' MAYOR A DOS ME-');
```

```
              WRITE(' :8,'SES DE DIFERENCIA, ');
```

```
              WRITELN(' FAVOR DE VOLVER A DAR EL MES');
```

```
              WRITELN;
```

```
              BANDZ:='*';
```

```
            END
```

```
          ELSE
```

```
            WRITELN
```

```
        ELSE
```

```
          IF DIFMM>=-9 THEN
```

```
            BEGIN
```

```
              WRITE('ERROR : EL MES DEBE SER EL PRIMERO ');
```

```
              WRITELN(' O MAXIMO EL SEGUNDO AÑO');
```

```
              WRITELN(' :8,'QUE ACABA DE TRANSMITIR ');
```

```
              WRITELN;
```

```
              BANDZ:='*';
```

```
            END
```

```
          END
```

```
        ELSE
```

```
          BEGIN
```

```
            WRITE('ERROR: EL MES ',MES:2,' NO EXISTE,');
```

```
            WRITELN;
```

```
            WRITELN(' FAVOR DE VOLVER A TRANSMITIR, ' :8,'EL MES ');
```

```
            WRITELN;
```

```
            BANDZ:='*';
```

```
          END
```

```
        UNTIL BANDZ=' ';
```

```
      END; (* FIN VMES *)
```

```
*****
* PROCEDURE VDIA *
*****
```

```
PROCEDURE VDIA;
```

```
BEGIN
```

```
  REPEAT
```

```
    REPEAT
```

```
      BANDZ:=' ';
```

```
      WRITE('DEME DIA => ');
```

```
      READ(DIA);WRITELN;
```

```
      DIFTA:=ANIO MOD 4;
```

```
      IF DIA<1 THEN
```

```
        BANDZ:='*'
```

```
      ELSE
```

```
        BEGIN
```

```
          CASE MES OF
```

```
            1,3,5,7,8,10,12:IF DIA>31 THEN BANDZ:='*';
```

```
            2:IF DIFTA=0 THEN
```

```
              IF DIA>29 THEN
```

```
                BANDZ:='*'
```

```
              ELSE
```

```
                WRITELN
```

```
              ELSE
```

```
                IF DIA>28 THEN
```

```
                  BANDZ:='*';
```

```
            4,6,9,11:IF DIA>30 THEN BANDZ:='*';
```

```
          END;
```

```
        END;
```

```
      IF BANDZ='*' THEN
```

```
        BEGIN
```

```
          WRITE('ERROR : EL DIA ',DIA:2,' DEL MES ',MES:2,' NO EXISTE ');
```

```
          WRITELN('FAVOR DE VOLVER A ');
```

```
          WRITELN(' :8,'TRANSMITIR EL DIA ');
```

```
          WRITELN;
```

```
        END;
```

```
      UNTIL BANDZ=' ';
```

```
      DIFDD:=DIA-NM.N.DD;
```

```
      IF DIFAA=0 THEN
```

```
        IF DIFMM=0 THEN
```

```
          IF DIFDD<0 THEN :
```

```
            BEGIN
```

```
              WRITE('ERROR : EL DIA DEBE SER MAYOR O IGUAL ');
```

```
              WRITELN(' AL DE LA FECHA DE LA');
```

```
              WRITE(' :8,'ULTIMA CONSULTA AL SUBSISTEMA, ');
```

```
              WRITELN(' FAVOR DE VOLVER A');
```

```
              WRITELN(' :8,'TRANSMITIR EL DIA');
```

```
              WRITELN;
```

```
              BANDZ:='*';
```

```
            END;
```

```
      UNTIL BANDZ=' ';
```

```
      NM .DD:=DIA;
```

```
      NM .MM:=MES;
```

```
      NM .AA:=ANIO;
```

```
      SEEKWRITE(NM,II);
```

```
      ERRCHK;
```

```
      IF ERROR THEN EXIT;
```

```
END; (* FIN VDIA *)
```

```
*****  
*          PROCEDURE FECHA          *  
*****
```

```
PROCEDURE FECHA;  
BEGIN  
  VAAN;  
  VMES;  
  VDIA;  
END;  
  
MODEND.
```

V.1.12 LISTADO FUENTE DEL PROGRAMA:

< M O D B >

```

*****
*                                     *
*   PROGRAMA   MODB                 *
*                                     *
*****

```

MODULE OVERLAY11;

VAR OPC2:EXTERNAL CHAR;

```

*****
*   PROCEDURE LISTAP                 *
*****

```

PROCEDURE LISTAP;

VAR BAND:CHAR;

BEGIN

REPEAT

WRITELN(' :3,'TEMPORADA',' ' :16,'PERIODO');

WRITELN;

WRITELN(' :6,'A',' ' :11,'DEL 21 DE MARZO AL 20 DE JUNIO');

WRITELN;

WRITELN(' :6,'B',' ' :11,'DEL 21 DE JUNIO AL 23 DE SEPTIEMBRE');

WRITELN;

WRITELN(' :6,'C',' ' :11,'DEL 24 DE SEPTIEMBRE AL 22 DE DICIEMBRE');

WRITELN;

WRITELN(' :6,'D',' ' :11,'DEL 23 DE DICIEMBRE AL 20 DE MARZO');

WRITELN;WRITELN;

WRITE(' :4,'DEME TEMPORADA QUE DESEA => ');

READLN(OPC2);

CASE OPC2 OF

'A','B','C','D':BAND:= ' ';

ELSE

BEGIN

WRITELN('*** E R R O R ***');

WRITELN('TRANSMITA UNICAMENTE DE LA <A> A LA <D>');

BAND:='*';

END

END

UNTIL BAND=' ';

WRITELN(' :4,'DEME EL NUM DE UNIDADES QUE PARA USTED REFLEJEN ');

WRITE(' :4,'UNA BUENA VENTA => ');

END; (* FIN DE LISTAP *)

MODEND.

V.1.13 LISTADO FUENTE DEL PROGRAMA:

< M O D C >

```
*****  
*  
*   PROGRAMA   MODC   *  
*  
*****
```

MODULE OVERLAYC;

TYPE ARCHI=RECORD

```
CLAVE,MS1,MS2,MS3:INTEGER;  
PRECIO:REAL;  
UNID,ANN,MES,DIA:INTEGER;  
CP:INTEGER;  
IMP,PDU:REAL;  
UVEN,UPED,UVP1:INTEGER;  
AA,MM,DD:INTEGER;  
END;
```

ARCHIII=RECORD

```
CLAVE:INTEGER;  
NOMBRE:STRING(306);  
R1,R2,R3,R4:INTEGER;  
PROV1:INTEGER;  
PRE1,DP1:REAL;  
PROV2:INTEGER;  
PRE2,DP2:REAL;  
PROV3:INTEGER;  
PRE3,DP3:REAL;  
DD,MM,AA:INTEGER;  
END;
```

```
VAR BF:EXTERNAL FILE OF ARCHI; (* ARCHIVO MAESTRO *)  
NM:EXTERNAL FILE OF ARCHIII;(*ARCHIVO DE NOMBRES Y ESTADISTICAS*)  
LIM,I,II:EXTERNAL INTEGER;  
ERROR: EXTERNAL BOOLEAN;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****
* PROCEDURE LINERR *
*****
```

```
PROCEDURE LINERR;
BEGIN
  WRITELN;
  WRITELN('ERROR TRANSMITA UNICAMENTE <S> O <N> ');
  WRITE('FAVOR DE VOLVER A TRANSMITIR S)I O N)O => ');
  END;
```

```
*****
* PROCEDURE VERFECH *
*****
```

```
PROCEDURE VERFECH;
VAR
  A,M,D:INTEGER;
  BAND2:CHAR;
  TIPA:REAL;
BEGIN
  WRITELN;
  WRITELN('DEME FECHA DE CADUCIDAD, EN ESTE ORDEN: AA MM DD ');
  WRITE('DONDE: AA ES EL AÑO,MM ES EL MES Y DD EL DIA => ');
  WRITELN;
  BAND2:= ' ';
  REPEAT
    WRITELN;
    IF BAND2='*' THEN
      WRITE('FAVOR DE VOLVER A DAR LA FECHA DE CADUCIDAD => ');
    BAND2:= ' ';
    READLN(A,M,D);
    IF A<BF .ANN THEN
      BEGIN
        WRITELN;
        WRITELN('ERROR: EL AÑO DEBE SER IGUAL O MAYOR,AL AÑO DE LA');
        WRITELN('          FECHA DE CADUCIDAD DE LAS UNIDADES EN EXISTENC. ');
        WRITELN('EL AÑO DE LA FECHA DE CADUCIDAD DE LAS UNIDADES EN ');
        WRITELN('EXISTENCIA ES : ',BF .ANN:2);
        WRITELN;
        BAND2:= '*';
      END;
    END;
```

```

IF (M<1) OR (M>12) THEN
  BEGIN
  WRITELN;
  WRITELN('ERROR: ESTE MES ',M:2,' NO EXISTE ');
  WRITELN;
  BAND2:='*';
  END;
TIPA:=A MOD 4;
CASE M OF
  1,3,5,7,8,10,12:IF (D<1) OR (D>31) THEN
    BEGIN
    WRITELN;
    WRITE('EL DIA ',D:2,' DEL MES ',M:2);
    WRITELN(' NO EXISTE');
    WRITELN;
    BAND2:='*';
    END;
  2:IF TIPA=0 THEN
    IF (D<0) OR (D>29) THEN
      BEGIN
      WRITELN;
      WRITE('EL DIA ',D:2,' DEL MES ',M:2,' DEL AÑO ');
      WRITELN(A:2,' NO EXISTE');
      WRITELN;
      BAND2:='*';
      END
    ELSE
      WRITELN
    ELSE
      IF (D<0) OR (D>28) THEN
        BEGIN
        WRITELN;
        WRITE('EL DIA ',D:2,' DEL MES ',M:2,' DEL AÑO ');
        WRITELN(A:2,' NO EXISTE');
        WRITELN;
        BAND2:='*';
        END;
  4,6,9,11:IF (D<1) OR (D>30) THEN
    BEGIN
    WRITELN;
    WRITELN('EL DIA ',D:2,' DEL MES ',M:2,' NO EXISTE');
    WRITELN;
    BAND2:='*';
    END;
ELSE
  WRITELN('MES ',M:2,' NO EXISTE ');
END
UNTIL BAND2 = '*';
BF .AA:=A;
BF .MM:=M;
BF .DD:=D;
END; (* FIN VERIF. DE FECHA *)

```

```
*****
*  PROCEDURE PIDE DAT  *
*****
```

```
PROCEDURE PIDE DAT;
VAR
  CUP,CVEP:INTEGER;
  NPRECH,NIMPM:REAL;
  BAND1:CHAR;
BEGIN
  CVEP:=I;
  SEEKREAD(NM,II);
  ERRCHK;
  IF ERROR THEN EXIT;
  SEEKREAD(BF,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  II:=CVEP;
  I:=II;
  WRITELN;
  WRITE('DEME LA CANTIDAD DE UNIDADES PEDIDAS => ');
  READLN(CUP);
  BF .UPED:=CUP;
  WRITELN;
  WRITELN;
  WRITE('DEME EL PRECIO DE VENTA DEL MEDICAMENTO => ');
  READLN(NPRECH);
  BF .PRECIO:=NPRECH;
  REPEAT
    BAND1:= ' ';
    WRITELN;
    WRITE('DEME EL IMPUESTO DE VENTA DEL MEDICAMENTO => ');
    READLN(NIMPM);
    IF NIMPM>1.0 THEN
      NIMPM:=NIMPM/100.0;
    IF NIMPM>100 THEN
      BEGIN
        WRITELN;
        WRITELN('*** ERROR EL DATO ESTA EQUIVOCADO REBASA EL 100 % ***');
        WRITELN;
        BAND1:='*';
      END
  UNTIL BAND1 = ' ';
  BF .IMP:=NIMPM;
  VERFECH;
  REPEAT
    BAND1:= ' ';
    WRITELN;
    WRITE('DEME CLAVE DEL PROVEEDOR PRINCIPAL DEL MEDICAMENTO => ');
    READLN(CVEP);
```

```

IF (CVEP<1) OR (CVEP>3) THEN
  BEGIN
  WRITELN;
  WRITELN('ERROR; CLAVE DE PROVEEDOR NO EXISTENTE ');
  WRITELN('      FAVOR DE VOLVER A DAR LA CLAVE ');
  WRITELN;
  BAND1:='*';
  END
UNTIL BAND1 = ' ';
NM ,PROV1:=CVEP;
SEEKWRITE(NM,II);
ERRCHK;
IF ERROR THEN EXIT;
BF ,CP:=CVEP;
SEEKWRITE(BF,I);
ERRCHK;
IF ERROR THEN EXIT;
END; (* FIN DE PIEDAT *)

```

```

*****
*      PROCEDURE CARGPED      *
*****

```

```

PROCEDURE CARGPED;
VAR
  CVEM;INTEGER;
  BAND,SN:CHAR;
BEGIN
  SN:=' ';
  IF LIM <> 0 THEN
    BEGIN
    WRITELN;WRITELN;
    WRITE('A CONTINUACION SE LE PIDEN LOS DATOS ');
    WRITELN(' DE LOS MEDICAMENTOS, A');
    WRITELN('LOS CUALES SE LES DESEA CARGAR UNIDADES PEDIDAS');
    WRITELN;WRITELN;
    REPEAT
      BAND:=' ';
      WRITELN;
      WRITE('DEME CLAVE DEL MEDICAMENTO => ');
      READLN(CVEM);
      IF CVEM>=1 THEN
        IF CVEM<=LIM THEN
          BEGIN
            WRITELN;
            WRITELN('ES LA CLAVE DEL MEDICAMENTO : ',NM,NOMBRE);
            WRITE('AL CUAL DESEA CARGAR UNIDS PEDIDAS? S)I O N)O => ');
          END
        END
      END
    UNTIL BAND = ' ';
  END

```

```

REPEAT
  READ(SN);WRITELN#
  IF SN IN 1'S','s'ó THEN
    BEGIN
      I:=CVEM#
      II:=I#
      PIEDAT#
      END
    ELSE
      IF SN IN 1'N','n'ó THEN
        SN:='N'
      ELSE
        LINERR
      UNTIL SN IN 1'N','n'ó
      END
  ELSE
    BEGIN
      WRITELN#
      WRITELN('ERROR: CLAVE DE MEDICAMENTO EXCEDE EL ');
      WRITE(' RANGO PERMITIDO LA CLAVE MAXIMA DEL MEDICAMENTO ');
      WRITELN(' QUE CONTIENE INFORMACION');
      WRITELN('ES LA : ',LIM:5);
      WRITELN#
      BAND:='*'#
      END
  ELSE
    BEGIN
      WRITELN#
      WRITELN('ERROR: LA CLAVE ',CVEM:5,' NO EXISTE. ');
      WRITELN#
      BAND:='*'#
      END;
  IF BAND<>'*' THEN
    BEGIN
      WRITELN#
      WRITE('DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O => ');
      REPEAT
        READ(SN);WRITELN#
        IF SN IN 1'S','s'ó THEN
          SN:='S'
        ELSE
          IF SN IN 1'N','n'ó THEN
            SN:='N'
          ELSE
            LINERR
          UNTIL SN IN 1'S','s','N','n'ó
          END
      UNTIL SN IN 1'N','n'ó
      END
  ELSE
    BEGIN
      WRITELN('ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION');
      WRITELN(' FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA');
      WRITELN(' OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO');
      END;
  WRITELN('*** TERMINA PROGRAMA DE CARGA DE UNIDADES PEDIDAS ***');
  END;
MODEND.

```

V.1.14 LISTADO FUENTE DEL PROGRAMA:

< M O D D >

```
*****
*                                     *
*   PROGRAMA   MODD                 *
*                                     *
*****
```

MODULE OVERLAYD;

```
TYPE  ARCHIII=RECORD
      CLAVE:INTEGER;
      NOMBRE:STRING1306;
      R1,R2,R3,R4:INTEGER;
      PROV1:INTEGER;
      PRE1,DP1:REAL;
      PROV2:INTEGER;
      PRE2,DP2:REAL;
      PROV3:INTEGER;
      PRE3,DP3:REAL;
      DD,MM,AA:INTEGER;
      END;
```

```
VAR  NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)
      LIM:EXTERNAL INTEGER;
      ERROR:EXTERNAL BOOLEAN;
      EXTERNAL PROCEDURE ERRCHK;
```

```
*****
* PROCEDURE LEERAN *
*****
```

```
PROCEDURE LEERAN;
VAR I,A5,A6:INTEGER;
    SIG:CHAR;
```

```
PROCEDURE NOMBRE;
VAR A4:INTEGER;
    A3:STRING1306;
```

```
BEGIN
A3:=NM .NOMBRE;
A4:=LENGTH(A3);
A5:=30-A4;
END;
```

```
*****
* PROCEDURE LEERAN1 *
*****
```

```
PROCEDURE LEERAN1;
VAR A1:INTEGER;
BEGIN
Writeln;
Writeln;
WRITE('=====');
Writeln('=====');
Writeln;
WRITE(' CLAVE . NOMBRE DEL MEDICAMENTO . RANGO 1 RANGO 2 ');
Writeln('RANGO 3 RANGO 4 FECHA DE TRABAJO');
Writeln;
WRITE('=====');
Writeln('=====');
Writeln;
FOR A1:=1 TO LIM DO
BEGIN
I:=A1;
SEEKREAD(NM,I);
ERRCHK;
IF ERROR THEN EXIT;
NOMBRE;
WRITE(' :2,NM .CLAVE:4,' :4,NM .NOMBRE,' :A5,' :2,NM ,R1:8);
WRITE(' :2,NM .R2:8,' :2,NM .R3:8,' :2,NM .R4:8,' :7);
Writeln(NM .AA:2,'/',NM .MM:2,'/',NM .DD:2);
END;
Writeln; Writeln; Writeln;
END; (*FIN DE LEERAN1*)
```

```
*****
* PROCEDURE LEERAN2 *
*****
```

```
PROCEDURE LEERAN2;
VAR AI:INTEGER;
BEGIN
Writeln;
Writeln;
WRITE('=====');
Writeln('=====');
Writeln;
WRITE(' CLAVE . NOMBRE DEL MEDICAMENTO . PROVEEDOR PRINCIPAL ');
Writeln('SEGUNDO PROVEEDOR ; TERCER PROVEEDOR ');
WRITE(' :42,'CLAVE PRECIO DESC. CLAVE PRECIO DESC. CLAVE');
Writeln('PRECIO DESC. ');
WRITE('=====');
Writeln('=====');
```

```

WRITELN;
FOR AI:=1 TO LIM DO
  BEGIN
    I:=AI;
    SEEKREAD(NM,I);
    ERRCHK;
    IF ERROR THEN EXIT;
    NOMBRE;
    WRITE(NM .CLAVE:6,' ':4,NM .NOMBRE,' ':A5,NM .PROV1:6,' ':4);
    WRITE(NM .PRE1:8:2,' ':3,NM .DP1:4:2,' ':4,NM .PROV2:5,' ':4);
    WRITE(NM .PRE2:8:2,' ':3,NM .DP2:4:2,' ':4,NM .PROV3:5,' ':4);
    WRITELN(NM .PRE3:8:2,' ':3,NM .DP3:4:2);
  END;
WRITELN; WRITELN; WRITELN;
END; (*FIN DE LEERAN2*)

```

```

*****
*   PROCEDURE   PRINCIPAL DEL PROGRAMA DE LEERAN *
*****

```

```

BEGIN (*PRINCIPAL*)
  IF LIM>0 THEN
    BEGIN
      SIG:=' ';
      REPEAT
        ASSIGN(NM,'B:NOMYEST.FIL');
        RESET(NM);
        WRITELN;
        WRITELN('CUAL LISTADO DESEA OBTENER: ');
        WRITELN;
        WRITELN(' 1.- NOMBRE Y RANGOS');
        WRITELN(' 2.- PROVEDORES');
        WRITELN;
        WRITE(' ANOTE EL NUMERO DEL LISTADO QUE DESEA : ');
        READLN(A6);
        CASE A6 OF
          1:LEERAN1;
          2:LEERAN2
        ELSE
          WRITELN(' ANOTE UNICAMENTE 1 o 2 ');
        END;
        WRITE(' DESEA SACAR OTRO LISTADO S)I o N)O: ');
        READLN(SIG);
        UNTIL SIG='N';
        I:=0;
        WRITELN; WRITELN; WRITELN;
      END
    ELSE
      BEGIN
        WRITELN('ERROR EL ARCHIVO DE NOMBRES Y ESTADISTICAS NO CONTIENE');
        WRITELN(' INFORMACION. FAVOR DE VOLVER A LISTAR EL MENU Y ');
        WRITELN(' TRANSMITA LA OPCION <F>, PARA DAR DE ALTA ALGUN');
        WRITELN(' MEDICAMENTO');
      END;
    END;
  END;
MODEND.

```

V.2 LISTADOS FUENTES DE LOS PROGRAMAS
DEL SUBSISTEMA DE
CONTROL DE VENTAS

V.2.1 LISTADO FUENTE DEL PROGRAMA :
< P R O Y 1 >

```
*****  
*  
*          PROGRAMA          PRINCIPAL          PROY1          *  
*  
*****
```

PROGRAM PRUEBA;

TYPE

ARCHI=RECORD

```
CLAVE,MS1,MS2,MS3:INTEGER;  
PRECIO:REAL;  
UNID,ANN,MES,DIA:INTEGER;  
CP:INTEGER;  
IMP,FDU:REAL;  
UVEN,UPED,UVP1:INTEGER;  
AA,MM,DD:INTEGER;  
END;
```

ARCHII=RECORD

```
CLAVE:INTEGER;  
MEDS1:INTEGER;  
MEDS2:INTEGER;  
MEDS3:INTEGER;  
PRECIO:REAL;  
UNID:INTEGER;  
IMP:REAL;  
UVEN:INTEGER;  
UPE1:INTEGER;  
UVPD:INTEGER;  
END;
```

ARCHIII=RECORD

```
CLAVE:INTEGER;  
NOMBRE:STRING(306);  
R1,R2,R3,R4:INTEGER;  
PROV1:INTEGER;  
PRE1,DP1:REAL;  
PROV2:INTEGER;  
PRE2,DP2:REAL;  
PROV3:INTEGER;  
PRE3,DP3:REAL;  
DD,MM,AA:INTEGER;  
END;
```

```

VAR
  BF:FILE OF ARCHI; (*ARCHIVO MAESTRO*)
  VA:FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)
  NH:FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)
  SIONO,NOMBREMP:STRING1306;
  I,LIMMAST,BLANCNOM:INTEGER;
  EXISTVA,LIMIWA,LIMINM,ANOEX,MESEX,DIAEX,VER:INTEGER;
  ERROR:BOOLEAN;
  FUNDESEA:CHAR;

```

```

EXTERNAL PROCEDURE $HLT;
EXTERNAL 116 PROCEDURE DUPLICAR;
EXTERNAL 126 PROCEDURE LECOPIA;
EXTERNAL 136 PROCEDURE FACT;
EXTERNAL 146 PROCEDURE CORTE;
EXTERNAL 156 PROCEDURE REGRESA;
EXTERNAL 166 PROCEDURE TRANSFERIR;
EXTERNAL 176 PROCEDURE ACTUALI7;

```

```

*****
* PROCEDURE ERRCHK *
*****

```

```

PROCEDURE ERRCHK;
BEGIN
  ERROR:=TRUE;
  CASE IORESULT OF
    0:ERROR:=FALSE;
    1:WRITELN('LEYENDO DONDE NO HAY DATOS');
    2:WRITELN('ERROR DE CP/M');
    3:WRITELN('BUSCANDO UN LIMITE EN BLANCO');
    4:WRITELN('ERROR DE CP/M');
    5:WRITELN('LLEGO AL FINAL DEL DISCO');
  ELSE
    WRITELN('ERROR DE CODIGO DESCONOCIDO');
  END;
END;

```

```
*****  
* PROCEDURE FECHAUX *  
*****
```

```
PROCEDURE FECHAUX;  
VAR  
    AN1,DI2:INTEGER;  
BEGIN  
IF ((ANOEX>=84)AND(ANOEX<100)AND(MESEX>0)AND(DIAEX>0)) THEN  
    BEGIN  
    VER:=1;  
    AN1:=ANOEX MOD 4;  
    CASE MESEX OF  
        1,3,5,7,8,10,12:DI2:=31;  
        4,6,9,11:DI2:=30;  
        2: BEGIN IF AN1=0 THEN  
                DI2:=29  
                ELSE  
                DI2:=28    END  
            END;  
    ELSE  
    VER:=0;  
    END; (*DEL CASE*)  
    IF VER=1 THEN  
    BEGIN  
    IF DIAEX>DI2 THEN  
        VER:=0;  
    END;  
    END;  
END;  
END;
```

```
*****  
* PROCEDURE FECHAEX *  
*****
```

```
PROCEDURE FECHAEX;  
VAR  
    SIG:CHAR;  
BEGIN  
    WRITELN('ANOTE LA FECHA DE HOY : ');  
    SIG:= ' ';
```

```

REPEAT
  ANOEX:=0;
  MESEX:=0;
  DIAEX:=0;
  WRITE('A N O => ');
  READLN(ANOEX);
  WRITELN;
  WRITE('M E S => ');
  READLN(MESEX);
  WRITELN;
  WRITE('D I A => ');
  READLN(DIAEX);
  WRITELN;
  FECHAUX;
  IF VER=1 THEN
    BEGIN
      SIG:='Q';
      WRITE('ANOTE LA FECHA CORRECTAMENTE ANO: ');
      WRITELN(ANOEX,' , MES: ',MESEX,' , DIA: ',DIAEX);
      END
    ELSE
      BEGIN
        WRITELN(' ***** E R R O R ***** ');
        SIG:=' ';
        WRITELN('ANOTE LA FECHA NUEVAMENTE ');
        WRITE('SE DESEA SALIRSE ANOTE Q, EN CASO CONTRARIO ');
        WRITE('TRANSMITA <RETURN> : ');
        READLN(SIG);
        END
  UNTIL SIG='Q';
END;

```

```

*****
* PROCEDURE CABEZA *
*****

```

```

PROCEDURE CABEZA;
VAR
  I:INTEGER;
BEGIN
  I:=0;
  SEEKREAD(NM,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  NOMBREM:=NM ,NOMBRE;
  BLANCONH:=30-LENGTH(NOMBREM);
END;

```

```
*****  
* PROCEDURE LETRA *  
*****
```

```
PROCEDURE LETRA#  
BEGIN  
  WRITELN#  
  WRITELN(' :30,' ':BLANCOS,NOMBREMP#)  
  WRITELN#  
  WRITELN('FUNCION', ' :10,' DESCRIPCION DE LA FUNCION')#  
  WRITE('-----')#  
  WRITELN(' :3,' 'A', ' :13,' 'DUPLICAR EL ARCHIVO MAESTRO')#  
  WRITELN(' :17,' 'PARA OBTENER EL ARCHIVO DE TRABAJO')#  
  WRITELN#  
  WRITELN(' :3,' 'B', ' :13,' 'LEER EL ARCHIVO DE TRABAJO DEL DIA')#  
  WRITELN#  
  WRITELN(' :3,' 'C', ' :13,' 'FACTURACION PARA CLIENTES')#  
  WRITELN#  
  WRITELN(' :3,' 'D', ' :13,' 'CORTE DE CAJA DE LAS VENTAS DEL DIA')#  
  WRITELN#  
  WRITELN(' :3,' 'E', ' :13,' 'CORRECCIONES AL ARCHIVO DE TRABAJO')#  
  WRITELN#  
  WRITE(' :3,' 'F', ' :13,' 'REGRESAR INFORMACION DEL ARCHIVO DE')#  
  WRITELN(' TRABAJO')#  
  WRITELN(' :17,' 'AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE')#  
  WRITELN(' :17,' 'INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE')#  
  WRITELN(' :17,' 'NOMBRES Y ESTADISTICAS')#  
  WRITELN#  
  WRITELN(' :3,' 'Q', ' :13,' 'SALIRSE')#  
  WRITE('-----')#  
  WRITELN#  
END#
```

```
*****
* PROCEDURE MENSAJE *
*****
```

```
PROCEDURE MENSAJE;
BEGIN
  WRITELN;WRITELN;
  WRITELN('!!! BIENVENIDO A UNA NUEVA SESION DE TRABAJO !!!');
  WRITELN;
  WRITELN('EL SUBSISTEMA "CONTROL DE VENTAS", REQUIERE QUE');
  WRITELN('EXISTAN EN EL DISCO B LOS SIGUIENTES ARCHIVOS:');
  WRITELN;
  WRITELN('1.- ARCHIVO MAESTRO');
  WRITELN('2.- ARCHIVO DE TRABAJO');
  WRITELN('3.- ARCHIVO DE NOMBRES Y ESTADISTICAS');
  WRITELN;WRITELN;
END;
```

```
*****
* PROCEDURE ASIGNAR *
*****
```

```
PROCEDURE ASIGNAR;
BEGIN
  ASSIGN(BF,'B:MAESTRO.FIL');
  RESET(BF);
  ASSIGN(VA,'B:TRABAJO.FIL');
  RESET(VA);
  ASSIGN(NM,'B:NOMYEST.FIL');
  RESET(NM);
  CABEZA;
  I:=0;
  LIMMAST:=0;
  LIMIVA:=0;
  LIMINM:=0;
  EXISTVA:=0;
  SEEKREAD(BF,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  LIMMAST:=BF .MS1;
  SEEKREAD(VA,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  LIMIVA:=VA .UNID;
  EXISTVA:=VA .UVPD;
  SEEKREAD(NM,I);
```

```

ERRCHK)
  IF ERROR THEN EXIT)
  LIMINH:=NM ,CLAVE)
  I:=0)
  CLOSE(BF,I))
  CLOSE(VA,I))
  CLOSE(NM,I))
  ASSIGN(BF,'B:MAESTRO.FIL')
  RESET(BF)
  ASSIGN(VA,'B:TRABAJO.FIL')
  RESET(VA)
  ASSIGN(NM,'B:NOMYEST.FIL')
  RESET(NM)
END)

```

```

*****
* PROCEDURE LISTAERR *
*****

```

```

PROCEDURE LISTAERR(NUMERR:INTEGER)
BEGIN
  WRITELN)
  WRITELN(' :12, ' * * *  E R R O R  * * * '))
  CASE NUMERR OF
    1:WRITELN(' EL ARCHIVO MAESTRO NO EXISTE O ESTA VACIO'))
    2:WRITELN(' EL ARCHIVO DE TRABAJO NO EXISTE O ESTA VACIO'))
    3:WRITELN(' EL ARCHIVO DE NOM. Y EST. NO EXISTE O ESTA VACIO'))
    4:BEGIN
      WRITELN(' EL ARCHIVO DE TRABAJO NO CONTIENE INFORMACION.))
      WRITELN(' FAVOR DE LISTAR EL MENU PRINCIPAL DE FUNCIONES))
      WRITELN(' Y ELIJA LA FUNCION <A> PARA LLENARLO ')
      END)
    5:WRITELN('UNICAMENTE SE ACEPTA A, B, C, D, E, F o Q')
    6:BEGIN
      WRITELN('ARCHIVO MAESTRO TIENE ',LIMMAST,' REGISTROS'))
      WRITELN('ARCHIVO DE NOMBRES Y EST. TIENE ',LIMINH,' REGISTROS'))
      WRITELN('ESTA USANDO ARCHIVOS EQUIVOCADOS'))
      END)
    7:BEGIN
      WRITELN('ARCHIVO MAESTRO TIENE ',LIMMAST,' REGISTROS'))
      WRITELN('ARCHIVO DE TRABAJO TIENE ',LIMIVA,' REGISTROS'))
      WRITELN('ESTA USANDO UN ARCHIVO DE TRABAJO EQUIVOCADO'))
      END)
    END)
  WRITELN)
END)

```

```
*****
* PROCEDURE BUSCAERR *
*****
```

```
PROCEDURE BUSCAERR;
BEGIN
  IF LIMMAST=0 THEN
    LISTAERR(1);
  IF EXISTVA=0 THEN
    LISTAERR(2);
  IF LIMINH=0 THEN
    LISTAERR(3);
END;
```

```
*****
* PROCEDURE BUSCAFUN *
*****
```

```
PROCEDURE BUSCAFUN;
BEGIN
  CASE FUNDESEA OF
    'A','a':BEGIN
      DUPLICAR;
      LIMIVA:=LIMMAST
      END;
    'Q','q':$HLT;
  ELSE
    BEGIN
      IF LIMIVA>0 THEN
        BEGIN
          CASE FUNDESEA OF
            'B','b':LECOPIA;
            'C','c':FACT;
            'D','d':CORTE;
            'E','e':BEGIN
              ACTUALI7;
              $HLT
              END;
            'F','f':BEGIN
              IF LIMIVA=LIMMAST THEN
                BEGIN
                  REGRESA;
                  IF LIMINH=LIMMAST THEN
                    BEGIN
                      TRANSFERIR
                    ELSE
                      LISTAERR(6)
                    END
                  ELSE
                    ELSE
```

```

LISTAERR(7)
END;

      $HLT
      END;
    ELSE
      LISTAERR(5);
    END;
  ELSE
    LISTAERR(4);
  END;
END;
END;

```

```

*****
*                PROGRAMA    PRINCIPAL                *
*****

```

```

BEGIN
  MENSAJE;
  ASIGNAR;
  WRITELN;
  WRITELN('EL SISTEMA TIENE ',LIMMAST:5,' MEDICAMENTOS. ');
  IF ((LIMMAST>0)AND(EXISTVA>0)AND(LIMINM>0)) THEN
    BEGIN
      REPEAT
        WRITELN;
        WRITE('DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ');
        WRITE(' ANOTE S)I : O N)O : ');
        READLN(SIONO);
        IF SIONO110 IN I'S','s'0 THEN
          LETRA;
        WRITE('FUNCION DESEADA : ');
        READLN(FUNDESEA);
        BUSCAFUN;
        UNTIL FALSE;
      END
    ELSE
      BUSCAERR;
      I:=LIMMAST;
      CLOSE(BF,I);
      CLOSE(VA,I);
      CLOSE(NM,I);
    END.

```

V.2.2 LISTADO FUENTE DEL PROGRAMA:
< M O D U L O >

```
*****  
*  
*          PROGRAMA          MODUL          *  
*  
*****
```

MODULE MODU1;

TYPE

ARCHI=RECORD

```
CLAVE,MS1,MS2,MS3:INTEGER;  
PRECIO:REAL;  
UNID,ANN,MES,DIA:INTEGER;  
CP:INTEGER;  
IMP,PDU:REAL;  
UVEN,UPED,UVP1:INTEGER;  
AA,MM,DD:INTEGER;  
END;
```

ARCHII=RECORD

```
CLAVE:INTEGER;  
MEDS1:INTEGER;  
MEDS2:INTEGER;  
MEDS3:INTEGER;  
PRECIO:REAL;  
UNID:INTEGER;  
IMP:REAL;  
UVEN:INTEGER;  
UPE1:INTEGER;  
UVPD:INTEGER;  
END;
```

VAR

```
BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)  
VA:EXTERNAL FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)  
ANOEX,MESEX,DIAEX:EXTERNAL INTEGER;  
ERROR:EXTERNAL BOOLEAN;  
LIMMAST,LIMIVA:EXTERNAL INTEGER;
```

EXTERNAL PROCEDURE ERRCHK;
EXTERNAL PROCEDURE FECHAEX;

```
*****
* PROCEDURE DUPLICAR *
*****
```

```
PROCEDURE DUPLICAR;
VAR
  J:INTEGER;
  VERIFI:CHAR;
```

```
*****
* PROCEDURE ASIGNAR *
*****
```

```
PROCEDURE ASIGNAR1;
BEGIN
  ASSIGN(BF,'B:MAESTRO.FIL');
  RESET(BF);
  ASSIGN(VA,'B:TRABAJO.FIL');
  RESET(VA);
  VERIFI:=' ';
END;
```

```
*****
* PROCEDURE FECHA1 *
*****
```

```
PROCEDURE FECHA1;
BEGIN
  REPEAT
    FECHAEX;
    WRITE(' '8,'REALMENTE ESTA CORRECTA LA FECHA ?. ANOTE S I O N O ');
    READLN(VERIFI)
  UNTIL ((VERIFI='S')OR(VERIFI='s'));
  VA .CLAVE:=0;
  VA .MEDS1:=ANDEX;
  VA .MEDS2:=MESEX;
  VA .MEDS3:=DIAEX;
  VA .PRECIO:=0;
  VA .UNID:=BF .MS1;
  VA .IMP:=BF .IMP;
  VA .UVEN:=1;
  VA .UPE1:=BF .UVP1;
  VA .UVPD:=1;
END;
```

```
*****
* PROCEDURE DUPLICA1 *
*****
```

```
PROCEDURE DUPLICA1#
VAR
  INDICE:INTEGER#
BEGIN
  WRITELN(' :B,'EL SISTEMA INICIA EL PROCESO PARA OBTENER EL')
  WRITELN(' :B,'ARCHIVO DE TRABAJO A PARTIR DEL ARCHIVO MAESTRO')
  WRITELN#WRITELN#
  WRITELN(' :B,'SE ESTA DUPLICANDO EL ARCHIVO.')
  WRITELN#
  WRITELN(' :B,'ESPERE LA SENAL PARA PODER CONTINUAR ...')
  FOR INDICE:=0 TO LIMMAST DO
    BEGIN
      J:=INDICE#
      SEEKREAD(BF,J)#
      ERRCHK#
      IF ERROR THEN EXIT#
      IF INDICE<>0 THEN
        BEGIN
          VA .CLAVE:=BF .CLAVE#
          VA .MEDS1:=BF .MS1#
          VA .MEDS2:=BF .MS2#
          VA .MEDS3:=BF .MS3#
          VA .PRECIO:=BF .PRECIO#
          VA .UNID:=BF .UNID#
          VA .IMP:=BF .IMP#
          VA .UVEN:=0#
          VA .UPE1:=BF .UPED#
          VA .UVPD:=BF .UVP1#
        END
      ELSE
        FECHA1#
        SEEKWRITE(VA,J)#
      ERRCHK#
    END#
  CLOSE(VA,J)#
  J:=0#
  WRITELN(' :B,'EL SISTEMA TERMINO DE DUPLICAR, PUEDE CONTINUAR ')
  WRITELN#WRITELN#WRITELN#
END#
```

```
*****  
*          PROGRAMA      PRINCIPAL          *  
*****
```

```
BEGIN  
  IF LIMIVA=0 THEN  
    BEGIN  
      ASIGNAR1;  
      DUPLICA1;  
    END  
  ELSE  
    BEGIN  
      WRITELN(' :12,'* * *   E R R O R   * * *');  
      WRITELN;  
      WRITELN(' :18,'EL ARCHIVO DE TRABAJO YA CONTIENE INFORMACION.');      WRITELN;  
      WRITELN(' :18,'POR LO TANTO  NO PUEDE USAR ESTA FUNCION.');      WRITELN;  
      END;  
    END;  
  MODEND.
```

V.2.3 LISTADO FUENTE DEL PROGRAMA:

< M O D U 2 >

```
*****  
*                                     *  
*           PROGRAMA      MODU2      *  
*                                     *  
*****
```

MODULE MODU2)

TYPE

```
ARCHII=RECORD  
  CLAVE:INTEGER;  
  MEDS1:INTEGER;  
  MEDS2:INTEGER;  
  MEDS3:INTEGER;  
  PRECIO:REAL;  
  UNID:INTEGER;  
  IMP:REAL;  
  UVEN:INTEGER;  
  UPE1:INTEGER;  
  UVPD:INTEGER;  
  END;
```

VAR

```
VA:EXTERNAL FILE OF ARCHII; (* ARCHIVO DE TRABAJO*)  
ERROR:EXTERNAL BOOLEAN;  
BLANCON:EXTERNAL INTEGER;  
NOMBREMP:EXTERNAL STRING(306);
```

EXTERNAL PROCEDURE ERRCHK;

```
*****  
*  PROCEDURE  LECOPIA  *  
*****
```

PROCEDURE LECOPIA;

VAR

```
  J,LIMITE:INTEGER;  
  SIG:CHAR;
```

```
*****  
* PROCEDURE REvisa2 *  
*****
```

```
PROCEDURE REvisa2(VAR LIMITER:INTEGER);  
BEGIN  
  J:=0;  
  SEEKREAD(VA,J);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
  WRITELN;  
  WRITELN(' :8,'ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :');  
  WRITELN;  
  WRITE(' :8,'ANO : ',VA .MEDS1:2,' MES : ',VA .MEDS2:2);  
  WRITELN(' , DIA : ',VA .MEDS3:2);  
  LIMITER:=VA .UNID  
END;
```

```
*****  
* PROCEDURE CABEZA2 *  
*****
```

```
PROCEDURE CABEZA2;  
BEGIN  
  J:=0;  
  SEEKREAD(VA,J);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
  WRITELN;  
  WRITELN;  
  WRITE(' :BLANCNOM,NOMBREMP,' :16,'ARCHIVO DE TRABAJO',' :16);  
  WRITELN('FECHA : ',VA .MEDS1:2,' / ',VA .MEDS2:2,' / ',VA .MEDS3:2);  
  WRITELN;  
  WRITE('=====');  
  WRITELN('=====');  
  WRITE('CLAVE: P R E C I O : IMP : U N I D ');  
  WRITELN(' A D E S');  
  WRITE(' :5,': DE VENTA : % : EN : VENDIDAS :);  
  WRITE(' :5,': SIN IMP CON IMP : : EXISTENCIA : ( ACT ) :);  
  WRITELN(' (ULT REM) : DEL PED. ');  
  WRITE('=====');  
  WRITELN('=====');  
  WRITELN;  
END;
```

```
*****
*   PROCEDURE   BUSCA2   *
*****
```

```
PROCEDURE BUSCA2;
VAR
  RENGLON:INTEGER;
  PRENET:REAL;
BEGIN
  SIG:=' ';
  RENGLON:=1;
  REPEAT
    IF RENGLON=14 THEN
      BEGIN
        CABEZA2)
        RENGLON:=1;
        END;
    WRITE(' ? @ ');
    READLN(J);
    IF ((J>0)AND(J<=LIMITE)) THEN
      BEGIN
        SEEKREAD(VA,J);
        ERRCHK;
        IF ERROR THEN EXIT;
        WRITELN;
        WRITE(VA .CLAVE:5,' '14,VA .PRECIO:10:2,' '14);
        WRITELN((VA .IMP*100):4:0,' '16,VA .UNID:7,' '8);
        WRITELN(VA .UVEN:7,' '15,VA .UPE1:7,' '14,VA .UUPB:7);
        END
      ELSE
        BEGIN
          WRITELN;
          WRITE('NO EXISTE, EL REGISTRO ',J:4,' ESTA FUERA DEL RANGO ');
          WRITELN(' ( ',LIMITE:5,' )');
          END;
    RENGLON:=RENGLON+1;
    READ(SIG);
  UNTIL ((SIG='Q')OR(SIG='@'));
  WRITELN;WRITELN;WRITELN;
END;
```

```
*****  
*                PROGRAMA    PRINCIPAL                *  
*****
```

```
BEGIN  
  ASSIGN(VA,'B:TRABAJO.FIL');  
  RESET(VA);  
  REVISAS2(LIMITE);  
  WRITELN;WRITELN;  
  WRITELN(' ':8,'COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA');  
  WRITELN;  
  WRITELN(' ':8,'PARA CONTINUAR PRESIONE <RETURN>');  
  READLN(SIG);  
  CABEZA2;  
  BUSCA2;  
  J:=0;  
END;  
MODEND.
```

V.2.4 LISTADO FUENTE DEL PROGRAMA :
< M O D U 3 >

```
*****  
*                                     *  
*                PROGRAMA            MODU3                *  
*                                     *  
*****
```

MODULE MODU3;

TYPE

```
ARCHII=RECORD  
  CLAVE:INTEGER;  
  MEDS1:INTEGER;  
  MEDS2:INTEGER;  
  MEDS3:INTEGER;  
  PRECIO:REAL;  
  UNID:INTEGER;  
  IMP:REAL;  
  UVEN:INTEGER;  
  UPE1:INTEGER;  
  UVPD:INTEGER;  
END;
```

VAR

```
VA:EXTERNAL FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)  
ERROR:EXTERNAL BOOLEAN;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****  
*  PROCEDURE FACTURAR  *  
*****
```

PROCEDURE FACTURAR;

VAR

```
SIGUE,CONTINUA,ACEPTA:CHAR;  
TOTACUM,TOTXMED,PRECMED,IMPVED,PREXUN,PRXUNXIN:REAL;  
TOADXFA,SUB,TOADXSE,TOADGRA,SUBTPRO:REAL;  
UNIDACT,UNIVEN,UNIRES,UNISOLI,J,AH,MH,DH,USD:INTEGER;  
UNIDEPE,UNVENPED,SALTE,NUMFAC,AXMXD,LIMIT3:INTEGER;
```

```
*****
* PROCEDURE CHECA3 *
*****
```

```
PROCEDURE CHECA3;
BEGIN
  SEEKREAD(VA,J);
  ERRCHK;
  IF ERROR THEN EXIT;
END;
```

```
*****
* PROCEDURE ENTRADA3 *
*****
```

```
PROCEDURE ENTRADA3;
BEGIN
  ASSIGN(VA,'B:TRABAJO.FIL');
  RESET(VA);
  J:=0;
  CHECA3;
  AH:=VA .MEDS1;
  MH:=VA .MEDS2;
  DH:=VA .MEDS3;
  AXMXD:=AH*MH*DH;
  LIMIT3:=VA .UNID;
  NUMFAC:=VA .UVEN;
  USO:=VA .UVPD;
  WRITELN('ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA : ');
  WRITELN('ANO : ',AH:2,' MES : ',MH:2,' DIA : ',DH:2);
  CLOSE(VA,J);
  ASSIGN(VA,'B:TRABAJO.FIL');
  RESET(VA);
END;
```

```
*****
* PROCEDURE DATOS3 *
*****
```

```
PROCEDURE DATOS3;
BEGIN
  UNIDACT:=VA .UNID;
  PRECHED:=VA .PRECIO;
  IMPMED:=VA .IMP;
  UNIVEN:=VA .UVEN;
  UNIDEPE:=VA .UPE1;
  UNVENPED:=VA .UVPD;
  SIGUE:='S';
END;
```

```
*****
*   PROCEDURE SIMILAR3   *
*****
```

```
PROCEDURE SIMILAR3(VAR SALTEP:INTEGER);
VAR
  SIMIL1,SIMIL2,SIMIL3,CONDIC:INTEGER;
BEGIN
  CONDIC:=0;
  SALTEP:=1;
  SIMIL1:=VA .MEDS1;
  SIMIL2:=VA .MEDS2;
  SIMIL3:=VA .MEDS3;
  WRITELN(' 1: 2: 3:');
  WRITELN(SIMIL1:6,SIMIL2:6,SIMIL3:6);
  WRITE('QUIERE ALGUNA S)I o N)O: ');
  READLN(SIGUE);
  IF ((SIGUE='S')OR(SIGUE='s')) THEN
    BEGIN
      REPEAT
        REPEAT
          WRITE('CLAVE : ');
          READLN(J);
          IF ((J=SIMIL1)OR(J=SIMIL2)OR(J=SIMIL3)) THEN
            BEGIN
              CHECA3;
              DATOS3;
              CONDIC:=3;
            END
          ELSE
            BEGIN
              WRITELN('ERROR ',J:4,' NO ES SIMILAR');
              WRITE('ANOTE ');
            END
        UNTIL CONDIC=3;
        IF ((UNIDACT<=0)AND(UNIDEPE<=0)) THEN
          BEGIN
            WRITELN('NO LA TENEMOS');
            WRITE('QUIERE OTRA SIMILAR? S)I o N)O :');
            READLN(SIGUE);
            IF ((SIGUE='N')OR(SIGUE='n')) THEN SALTEP:=9;
          END
        ELSE
          BEGIN
            SALTEP:=9;
            SIGUE:='S';
            WRITE('CUANTAS : ');
            READLN(UNISOLI);
          END;
        UNTIL SALTEP=9;
      END;
    END;
END;
```

```
*****
*   PROCEDURE FACTMED3   *
*****
```

```
PROCEDURE FACTMED3#
```

```
BEGIN
```

```
  SALTE:=0#
```

```
  CHECA3#
```

```
  DATOS3#
```

```
  IF ((UNIDACT<=0)AND(UNIDEPE<=0)) THEN
    SIMILAR3(SALTE#)
```

```
  IF ((UNISOLI>(UNIDACT+UNIDEPE))AND((UNIDACT+UNIDEPE)>0)) THEN
    BEGIN
```

```
    WRITELN('SOLO TENGO : ',(UNIDACT+UNIDEPE):4#)
```

```
    WRITE('SE LAS LLEVA S)I O N)O : ');
```

```
    READLN(SIGUE#)
```

```
    IF ((SIGUE='S')OR(SIGUE='s')) THEN
```

```
      UNISOLI:=UNIDACT+UNIDEPE#
```

```
      SALTE:=3#
```

```
    END#
```

```
  IF SALTE<>0 THEN
```

```
    BEGIN
```

```
      WRITE('B)IEN O M)AL? ');
```

```
      READLN(CONTINUA#)
```

```
    END#
```

```
  IF (((CONTINUA='B')OR(CONTINUA='b'))AND((SIGUE='S')OR(SIGUE='s')))
```

```
    THEN
```

```
    BEGIN
```

```
      PREXUN:=PRECMED*UNISOLI#
```

```
      PRXUNXIM:=PREXUN*IMPMD#
```

```
      TOTXMED:=PREXUN+PRXUNXIM#
```

```
      PRECMED:=TOTXMED/UNISOLI#
```

```
      SUBTPRO:=SUBTPRO+TOTXMED#
```

```
      WRITE(SUBTPRO:12:2,' '16,J:5,' '3,PRECMED:10:2,' '13#)
```

```
      WRITE(UNISOLI:5,TOTXMED:12:2,' '3,'');
```

```
      READLN(ACEPTA#)
```

```
      IF ACEPTA='*' THEN
```

```
        BEGIN
```

```
          SUB:=TOTXMED#
```

```
          UNIRES:=UNIDACT-UNISOLI#
```

```
          IF UNIRES>=0 THEN
```

```
            BEGIN
```

```
              VA .UVEN:=UNIVEN+UNISOLI#
```

```
              VA .UNID:=UNIRES#
```

```
            END
```

```
          ELSE
```

```
            BEGIN
```

```
              VA .UNID:=0#
```

```
              VA .UVEN:=UNIVEN+UNIDACT#
```

```
              VA .UPE1:=UNIDEPE+UNIRES#
```

```
            - VA .UVPD:=UNVENPED-UNIRES#
```

```
          END#
```

```
SEEKWRITE(VA,J);  
ERRCHK;  
IF ERROR THEN  
EXIT  
END  
ELSE  
SUBTPRO:=SUBTPRO-TOTXMED;  
END;  
TOTACUM:=TOTACUM+SUB;  
WRITE(TOTACUM:12:2);  
END;
```

```
*****  
* PROCEDURE CEROS3 *  
*****
```

```
PROCEDURE CEROS3;  
BEGIN  
SIGUE:=' '  
TOTXMED:=0;  
SUB:=0;  
UNIDACT:=0;  
PRECHED:=0;  
IMPHEM:=0;  
UNIVEN:=0;  
UNIDEPE:=0;  
UNVENPED:=0;  
UNIRES:=0;  
PREXUN:=0;  
PRXUNXIM:=0;  
END;
```

```
*****
*  PROCEDURE ADICION3 *
*****
```

```
PROCEDURE ADICION3(VAR SUBTPRO,TOADXFA:REAL);
VAR
  UNIADIC:INTEGER;
  PREADIC,IMPADIC,PRXUNAD,PXUXIAD,TOTXPAD:REAL;
BEGIN
  TOADXFA:=0;
  WHILE ((CONTINUA='A')OR(CONTINUA='a')) DO
    BEGIN
      WRITELN('PREC UNID IMP');
      READ(PREADIC,UNIADIC,IMPADIC);
      WRITE('B)IEN o M)ALT');
      READLN(CONTINUA);
      IF ((CONTINUA='B')OR(CONTINUA='b')) THEN
        BEGIN
          IF IMPADIC>1 THEN
            IMPADIC:=IMPADIC/100;
          PRXUNAD:=PREADIC*UNIADIC;
          PXUXIAD:=PRXUNAD*IMPADIC;
          TOTXPAD:=PRXUNAD+PXUXIAD;
          SUBTPRO:=SUBTPRO+TOTXPAD;
          WRITE(SUBTPRO:12:2,' '19,'EX',' '13,PREADIC:10:2,' '13);
          WRITE(UNIADIC:5,TOTXPAD:12:2,' '13,' ');
          READLN(ACCEPTA);
          IF ACCEPTA='*' THEN
            BEGIN
              TOADXFA:=TOADXFA+TOTXPAD;
              WRITE(SUBTPRO:12:2);
            END
          ELSE
            IF SUBTPRO>0 THEN
              SUBTPRO:=SUBTPRO-TOTXPAD;
            END;
          WRITE(' A)DIC o T)OT? ');
          READLN(CONTINUA);
        END;
    END;
END;
```

```

*****
*   PROCEDURE  IMFINFAC  *
*****
PROCEDURE IMFINFAC;
VAR
    CAMBIO,DINERO:REAL;
BEGIN
    WRITELN(TOTACUM:12:2,' '120,' T O T A L    $ ',' '13,TOTACUM:14:2);
    REPEAT
        WRITE('DINERO ');
        READLN(DINERO);
        WRITE('B)IEN O M)AL? ');
        READLN(CONTINUA);
        IF ((DINERO>=TOTACUM)AND((CONTINUA='B')OR(CONTINUA='b')) ) THEN
            BEGIN
                CAMBIO:=DINERO-TOTACUM;
                CONTINUA:='K';
                WRITE(DINERO:12:2,' '121,' D I N E R O    $ ',' '13);
                WRITELN(DINERO:14:2);
                WRITE(TOTACUM:12:2,' '121,' T O T A L    $ ',' '13);
                WRITELN(TOTACUM:14:2);
                WRITELN(CAMBIO:12:2,' '121,' C A M B I O    $ ',' '13);
                WRITELN(CAMBIO:14:2);
                WRITELN(' '133,'GRACIAS POR SU COMPRA ');
                NUMFAC:=NUMFAC+1;
            END
        ELSE
            WRITELN('ERROR ANOTE NUEVAMENTE');
    UNTIL CONTINUA='K';
END;

```

```

*****
*   PROCEDURE  CERRAR3  *
*****
PROCEDURE CERRAR3;
BEGIN
    J:=0;
    CHECA3;
    AH:=VA .MEDS1;
    MH:=VA .MEDS2;
    DH:=VA .MEDS3;
    TOADGRA:=VA .PRECIO;
    TOADXSE:=TOADXSE+TOADGRA;
    VA .MEDS1:=AH;
    VA .MEDS2:=MH;
    VA .MEDS3:=DH;
    VA .PRECIO:=TOADXSE;
    VA .UNID:=LIMIT3;
    VA .UVEN:=NUMFAC;
    VA .UPE1:=1;
    VA .UVPD:=USO;
    SEEKWRITE(VA,J);
    ERRCHK;
    J:=LIMIT3;
    CLOSE(VA,J);
    J:=0;
END;

```

```
*****
*   PROCEDURE FACTURA3 *
*****
```

```
PROCEDURE FACTURA3;
```

```
BEGIN
```

```
  WRITELN('COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA');
```

```
  WRITE('PRESIONE < RETURN > ');
```

```
  READLN(SIGUE);
```

```
  WRITELN;
```

```
  TOADXSE:=0;
```

```
  TOADGRA:=0;
```

```
  REPEAT
```

```
  WRITELN;
```

```
  SIGUE:=' ';
```

```
  TOTACUM:=0;
```

```
  SUBTPRO:=0;
```

```
  WRITE('6 FAC: ',NUMFAC:5,' '':19,'FACTURA NUM. ',NUMFAC:5;
```

```
  WRITELN(' '':10,AM,' / ',MH,' / ',DH);
```

```
  WRITE(' '':29,'CLAVE', ' '':2,'P R E C I O', ' '':14,'CANT.');
```

```
  WRITELN(' '':2,'SUBTOTAL');
```

```
  REPEAT
```

```
  CEROS3;
```

```
  WRITE('CLA CAN :');
```

```
  READLN(J,UNISOLI);
```

```
  WRITE('B)IEN o M)AL ? ');
```

```
  READLN(CONTINUA);
```

```
  IF ((CONTINUA='B')OR(CONTINUA='b')) THEN
```

```
  BEGIN
```

```
    IF ((J>0)AND(J<=LIMIT3)) THEN
```

```
      FACTMED3
```

```
    ELSE
```

```
      WRITE('NO EXISTE, ');
```

```
  END;
```

```
  WRITE(' T)OT o RET? ');
```

```
  READLN(SIGUE)
```

```
  UNTIL ((SIGUE='T')OR(SIGUE='t'));
```

```
  TOADXFA:=0;
```

```
  CONTINUA:='C';
```

```
  WRITE('T)OT o A)DIC ? ');
```

```
  READLN(CONTINUA);
```

```
  IF ((CONTINUA='A')OR(CONTINUA='a')) THEN
```

```
  BEGIN
```

```
    ADICION3(SUBTPRO,TOADXFA);
```

```
    TOTACUM:=TOTACUM+TOADXFA;
```

```
    TOADXSE:=TOADXSE+TOADXFA
```

```
  END;
```

```
  IF TOTACUM>0 THEN IMFINFAC;
```

```
  WRITE('DESEA DTRA FACTURAT ?');
```

```
  READLN(SIGUE)
```

```
  UNTIL ((SIGUE='N')OR(SIGUE='n'));
```

```
  WRITELN;
```

```
  CERRAR3;
```

```
END;
```

```

*****
*                                     *
*           PROGRAMA      PRINCIPAL   *
*                                     *
*****

```

```

BEGIN
  ENTRADA3)
  IF ((AXHXD=0)OR(LIMIT3=0)) THEN
    BEGIN
      WRITE('ERROR      LA FECHA DEL ARCHIVO DE TRABAJO CONTIENE UN')
      WRITELN(' CERO.')
```

```

      WRITE('          LISTE EL MENU PRINCIPAL DE FUNCIONES,')
      WRITELN(' ELIJA LA')
      WRITE('          FUNCION <Q> PARA SALIR DEL SUBSISTEMA ')
      WRITELN(' Y CONSULTE')
      WRITELN('          EL MANUAL CORRESPONDIENTE.')
```

```

    END
  ELSE
    BEGIN
      IF USO=5 THEN
        BEGIN
          WRITELN('ESTE ARCHIVO DE TRABAJO YA NO PUEDE USARSE')
          WRITELN('PORQUE CON EL CORRIO LA OPCION <F>')
```

```

          WRITELN('POR LO TANTO DEBE CREAR OTRO ARCHIVO DE TRABAJO.')
```

```

          WRITELN('SI TIENE DUDA CONSULTE EL MANUAL.')
```

```

        END
      ELSE
        BEGIN
          WRITELN)
          WRITE('L E  S I R V E ? S I O N O : ')
          READLN(SIGUE)WRITELN)
          WRITE('DESEA CONTINUAR ? S I O N O : ')
          READLN(CONTINUA)
        END
      IF (((SIGUE='S')OR(SIGUE='s'))AND((CONTINUA='S')OR(CONTINUA='s')))
        THEN
          FACTURAS
        ELSE
          WRITE('ELIMINE ESTE ARCHIVO Y OBTENGA EL ADECUADO ')
          WRITELN('INMEDIATAMENTE')
        END)
      WRITELN)WRITELN)WRITELN)
    END)
  MODEND.

```

V.2.5 LISTADO FUENTE DEL PROGRAMA:

< M O D U 4 >

```
*****  
*                                     *  
*           PROGRAMA           MODU4           *  
*                                     *  
*****
```

MODULE MODU4:

TYPE

ARCHIII=RECORD

```
CLAVE:INTEGER;  
NOMBRE:STRING1306;  
R1,R2,R3,R4:INTEGER;  
PROV1:INTEGER;  
PRE1,DP1:REAL;  
PROV2:INTEGER;  
PRE2,DP2:REAL;  
PROV3:INTEGER;  
PRE3,DP3:REAL;  
DD,MM,AA:INTEGER;  
END;
```

ARCHII=RECORD

```
CLAVE:INTEGER;  
MEDS1:INTEGER;  
MEDS2:INTEGER;  
MEDS3:INTEGER;  
PRECIO:REAL;  
UNID:INTEGER;  
IMP:REAL;  
UVEN:INTEGER;  
UPE1:INTEGER;  
UVPD:INTEGER;  
END;
```

VAR

```
NH:EXTERNAL FILE OF ARCHIII; (*ARCHIVO MAESTRO*)  
VA:EXTERNAL FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)  
ERROR:EXTERNAL BOOLEAN;  
BLANCNO:EXTERNAL INTEGER;  
NOMBREMP:EXTERNAL STRING1306;  
JJ,UNIVENT,ANOCORTE,MESCORTE,DIACORTE,LIMIT4:INTEGER;  
VENPARC,SUMAVP,PP,FUE,IMPUESTO,PRECTOT,SUMAPTOT:REAL;  
PREIXUNI,GANANCIA,SUMAPXUN,SUMAGANA,SUMAINP,UV,IMP:REAL;  
SEGUIR:CHAR;
```

EXTERNAL PROCEDURE ERRCHK;

* PROCEDURE CHECLEVA4 *

PROCEDURE CHECLEVA4;
BEGIN
 SEEKREAD(VA, JJ);
 ERRCHK;
 IF ERROR THEN EXIT;
END;

* PROCEDURE CHECLENM4 *

PROCEDURE CHECLENM4;
BEGIN
 SEEKREAD(NM, JJ);
 ERRCHK;
 IF ERROR THEN EXIT;
END;

* PROCEDURE ABRIR4 *

PROCEDURE ABRIR4;
BEGIN
 ASSIGN(NM, 'B:INOMYEST.FIL');
 RESET(NM);
 ASSIGN(VA, 'B:TRABAJO.FIL');
 RESET(VA);
 JJ:=0;
 CHECLEVA4;
 ANOCORTE:=VA .MEDS1;
 MESCORTE:=VA .MEDS2;
 DIACORTE:=VA .MEDS3;
 LIMIT4:=VA .UNID;
END;

```
*****  
* PROCEDURE CABEZA4 *  
*****
```

```
PROCEDURE CABEZA4;
```

```
  BEGIN
```

```
    WRITELN;
```

```
    WRITE(' ':30,' ':BLANCNOM,NOMBREMP,' ':10,ANOCORTE:2,' / ');
```

```
    WRITELN(MESCORTE:2,' / ',DIACORTE:2);
```

```
    WRITELN;
```

```
    WRITE('=====');
```

```
    WRITE('=====');
```

```
    WRITELN('=====');
```

```
    WRITELN;
```

```
    WRITE('M E D I C I N A UNIDADES PRECIO SUBTOTAL');
```

```
    WRITELN(' I V A T O T A L PRECIO PREC.COMP. GANANCIA');
```

```
    WRITE('CLAVE N O M B R E VENDIDAS DE $ ');
```

```
    WRITELN(' $ $ DE X UNID N E T A');
```

```
    WRITE(' COMPRA VENTAS ');
```

```
    WRITELN(' COMPRA VENDIDAS $');
```

```
    WRITELN;
```

```
    WRITE('=====');
```

```
    WRITELN('=====');
```

```
    WRITELN('=====');
```

```
    WRITELN;
```

```
  END;
```

```
*****  
* PROCEDURE CORTE4 *  
*****
```

```
PROCEDURE CORTE4;
```

```
  VAR
```

```
    H,REGLON,BLANCOS:INTEGER;
```

```
    NOMBMED:STRING(306);
```

```
  BEGIN
```

```
    SUMAVP:=0;
```

```
    SUMAPTOT:=0;
```

```
    SUMAIMP:=0;
```

```
    SUMAPXUN:=0;
```

```
    SUMAGANA:=0;
```

```
    REGLON:=1;
```

```
    FOR H:=1 TO LIMIT4 DO
```

```
      BEGIN
```

```
        JJ:=H;
```

```
        CHECLEVA4;
```

```
        CHECLENM4;
```

```

IF ((VA .UVEN>0)OR(VA .UVPD>0)) THEN
  BEGIN
    VENPARC:=0;
    UNIVENT:=VA .UVEN+VA .UVPD;
    PP:=VA .PRECIO;
    VENPARC:=UNIVENT*VA .PRECIO;
    SUMAVP:=SUMAVP+VENPARC;
    IMPUESTO:=VA .IMP*VENPARC;
    SUMAIMP:=SUMAIMP+IMPUESTO;
    PRECTOT:=VENPARC+IMPUESTO;
    SUMAPTOT:=SUMAPTOT+PRECTOT;
    PREIXUNI:=NMN .PREIX*UNIVENT;
    GANANCIA:=VENPARC-PREIXUNI;
    SUMAPXUN:=SUMAPXUN+PREIXUNI;
    SUMAGANA:=SUMAGANA+GANANCIA;
    NOMBMED:=NMN .NOMBRE;
    BLANCOS:=30-LENGTH(NOMBMED);
    WRITE(VA .CLAVE:5,' '12,NM .NOMBRE,' '13,UNIVENT:6);
    WRITE(' '12,VA .PRECIO:9:2,' '13,VENPARC:10:2,' '13);
    WRITE(IMPUESTO:9:2,' '13,PRECTOT:10:2,' '12,NMN .PREIX:8:2,' '13);
    WRITELN(PREIXUNI:9:2,' '12,GANANCIA:10:2);
    RENGLON:=RENGLON+1;
    IF RENGLON=40 THEN
      BEGIN
        CABEZA4;
        RENGLON:=1;
        WRITELN;WRITELN;WRITELN
      END;
    END;
  END;
END;

```

```

*****
* PROCEDURE TOTALES4 *
*****

```

```

PROCEDURE TOTALES4;
BEGIN
  WRITE('VENTA TOTAL DEL DIA $ ',' '13,SUMAVP:15:2,' '12);
  WRITE(SUMAIMP:13:2,SUMAPTOT:15:2,' '14,SUMAPXUN:15:2);
  WRITELN(SUMAGANA:14:2);
  JJ:=0;
  CHECLEVA4;
  FUE:=VA .PRECIO;
  VA .MEDS1:=ANOCORTE;
  VA .MEDS2:=MESCORTE;
  VA .MEDS3:=DIACORTE;
  VA .PRECIO:=FUE;
  VA .UPE1:=1;
  SEEKWRITE(VA,JJ);
  ERRCHK;
  CLOSE(VA,JJ);
  CLOSE(NM,JJ);
  WRITE('VENTA TOTAL DEL DIA DE ARTICULOS FUERA DE INVENTARIO $ ');
  WRITELN(' '10,FUE:15:2);
END;

```

```
*****  
* PROCEDURE NOTAS4 *  
*****
```

```
PROCEDURE NOTAS4;
```

```
BEGIN
```

```
WRITELN;WRITELN;WRITELN;
```

```
WRITELN('N O T A : NO OLVIDE HACER LO SIGUIENTE:');)
```

```
WRITE('      1o. REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO ');)
```

```
WRITELN('AL ARCHIVO MAESTRO, USE LA FUNCION <F> DEL MENU.');
```

```
WRITE('      14. RECUERDE QUE LA FUNCION <F> SOLO PUEDE USARSE UNA ');)
```

```
WRITELN('VEZ CON CADA ARCHIVO DE TRABAJO, TENGA CUIDADO');
```

```
WRITELN;
```

```
WRITE('      2o. PASE EL ARCHIVO DE TRABAJO AL DISCO');
```

```
WRITELN(' ALMACENADOR');
```

```
WRITELN;WRITELN;WRITELN;
```

```
JJ:=0;
```

```
END;
```

```
*****  
* PROCEDURE CORTE *  
*****
```

```
PROCEDURE CORTE;
```

```
*****  
*          PROGRAMA      PRINCIPAL      *  
*****
```

```
BEGIN
```

```
ABRIR4;
```

```
WRITELN('COLOQUE EL PAPEL AL INICIO DE UN NUEVA HOJA');
```

```
WRITELN('PRESIONE <RETURN>');
```

```
READLN(SEGUIR);)
```

```
CABEZA4;
```

```
CORTE4;
```

```
TOTALES4;
```

```
NOTAS4;
```

```
END;
```

```
MODEND.
```

V.2.6 LISTADO FUENTE DEL PROGRAMA :

< M O D U 5 >

```
*****  
*                                     *  
*           PROGRAMA           MODUS           *  
*                                     *  
*****
```

MODULE MODUS;

TYPE

```
  ARCHI=RECORD  
    CLAVE,MS1,MS2,MS3:INTEGER;  
    PRECIO:REAL;  
    UNID,ANN,MES,DIA:INTEGER;  
    CP:INTEGER;  
    IMP,PDU:REAL;  
    UVEN,UPED,UVP1:INTEGER;  
    AA,MM,DD:INTEGER;  
  END;
```

```
  ARCHII=RECORD  
    CLAVE:INTEGER;  
    MEDS1:INTEGER;  
    MEDS2:INTEGER;  
    MEDS3:INTEGER;  
    PRECIO:REAL;  
    UNID:INTEGER;  
    IMP:REAL;  
    UVEN:INTEGER;  
    UPE1:INTEGER;  
    UVPD:INTEGER;  
  END;
```

VAR

```
  BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)  
  VA:EXTERNAL FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)  
  ERROR:EXTERNAL BOOLEAN;  
  LIMMAST:EXTERNAL INTEGER;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****
* PROCEDURE REGRESA *
*****
```

```
PROCEDURE REGRESA;
VAR
  J,ANO,MES,DIA:INTEGER;
```

```
*****
* PROCEDURE ENTRADAS *
*****
```

```
PROCEDURE ENTRADAS;
BEGIN
  ASSIGN(VA,'B:TRABAJO.FIL');
  RESET(VA);
  ASSIGN(BF,'B:MAESTRO.FIL');
  RESET(BF);
  J:=0;
  SEEKREAD(VA,J);
  ERRCHK;
  ANO:=VA .MEDS1;
  MES:=VA .MEDS2;
  DIA:=VA .MEDS3;
  WRITELN(' EL ARCHIVO DE TRABAJO QUE ACTUALIZARA');
  WRITELN(' AL ARCHIVO MAESTRO ES DE FECHA :');
  WRITELN('');
  WRITELN(' DIA : ',DIA:2,' MES : ',MES:2,' ANO : ',ANO:2);
END;
```

```
*****
* PROCEDURE RETORNAS *
*****
```

```
PROCEDURE RETORNAS;
VAR
  INDICEJ:INTEGER;
BEGIN
  IF VA .UNID=LIMMAST THEN
    BEGIN
      IF VA .UVPD<>5 THEN
        BEGIN
          J:=1;
          WRITELN('EL ARCHIVO DE TRABAJO ES EL CORRECTO.');
```

```

FOR INDICEJ:=1 TO LIMMAST DO
  BEGIN
    J:=INDICEJ;
    SEEKREAD(VA,J);
    ERRCHK;
    IF ERROR THEN EXIT;
    SEEKREAD(BF,J);
    ERRCHK;
    IF ERROR THEN EXIT;
    BF .UNID:=VA .UNID;
    BF .UVEN:=BF .UVEN+VA .UVEN;
    BF .UPED:=VA .UPE1;
    BF .UVP1:=BF .UVP1+VA .UVPD;
    SEEKWRITE(BF,J);
    ERRCHK;
    END;

CLOSE(BF,J);
Writeln;
Writeln('EL SISTEMA TERMINO DE ACTUALIZAR AL MAESTRO');
END
ELSE
  BEGIN
    CLOSE(BF,J);
    Writeln('ESTE ARCHIVO DE TRABAJO YA FUE USADO UNA VEZ. ');
    Writeln;
    Writeln('NO ESTA PERMITIDO ACTUALIZAR DOS VECES ');
    Writeln('CON EL MISMO ARCHIVO DE TRABAJO');
    END;
END
ELSE
  BEGIN
    Writeln('ESTA USANDO UN ARCHIVO DE TRABAJO EQUIVOCADO. ');
    Writeln('YA QUE NO TIENE EL MISMO NUMERO DE REGISTROS');
    Writeln('      QUE EL ARCHIVO MAESTRO. ');
    Writeln;
    Writeln('VERIFIQUE LOS ARCHIVOS ');
    END;
END;
END;

```

```
*****  
*   PROCEDURE   CIERRAS   *  
*****
```

```
PROCEDURE CIERRAS;  
BEGIN  
  J:=0;  
  VA .MEDS1:=ANO;  
  VA .MEDS2:=MES;  
  VA .MEDS3:=DIA;  
  VA .UVPD:=5;  
  SEEKWRITE(VA,J);  
  ERRCHK;  
  CLOSE(VA,J);  
END;
```

```
*****  
*   PROGRAMA   PRINCIPAL   *  
*****
```

```
BEGIN  
  ENTRADAS;  
  RETORNAS;  
  CIERRAS;  
END;  
MODEND.
```

V.2.7 LISTADO FUENTE DEL PROGRAMA:
< M O D U 6 >

```
*****  
*  
*          PROGRAMA          MODU6          *  
*  
*****
```

MODULE MODU6;

TYPE

```
ARCHI=RECORD  
  CLAVE,MS1,MS2,MS3:INTEGER;  
  PRECIO:REAL;  
  UNID,ANN,MES,DIA:INTEGER;  
  CP:INTEGER;  
  IMP,PDU:REAL;  
  UVEN,UFED,UVP1:INTEGER;  
  AA,MM,DD:INTEGER;  
  END;
```

```
ARCHIII=RECORD  
  CLAVE:INTEGER;  
  NOMBRE:STRING(306);  
  R1,R2,R3,R4:INTEGER;  
  PROV1:INTEGER;  
  PRE1,DP1:REAL;  
  PROV2:INTEGER;  
  PRE2,DP2:REAL;  
  PROV3:INTEGER;  
  PRE3,DP3:REAL;  
  DD,MM,AA:INTEGER;  
  END;
```

VAR

```
BF:EXTERNAL FILE OF ARCHI; (*ARCHIVO MAESTRO*)  
NM:EXTERNAL FILE OF ARCHIII; (*ARCHIVO DE NOMBRES Y ESTADISTICAS*)  
LIMMAST,MESEX,DIAEX,VER:EXTERNAL INTEGER;  
ERROR:EXTERNAL BOOLEAN;
```

EXTERNAL PROCEDURE ERRCHK;
EXTERNAL PROCEDURE FECHAEX;

* PROCEDURE TRANSFERIR *

PROCEDURE TRANSFERIR;
VAR
DITO, I, MESEX, DIAEX: INTEGER;
SIG: CHAR;

* PROCEDURE TRANS6 *

PROCEDURE TRANS6;
VAR
IT: INTEGER;
BEGIN
WRITELN;
WRITELN('EL SISTEMA ESTA TRANSFIRIENDO INFORMACION');
WRITELN;
WRITELN('ESPERE LA SENAL PARA PODER CONTINUAR');
MESEX:=MESEX-1;
DITO:=(MESEX*30)+DIAEX;
DITO:=DITO-8;
FOR IT:=1 TO LIMMAST DO
BEGIN
IT:=IT;
SEEKREAD(BF, I);
ERRCHK;
IF ERROR THEN EXIT;

```

IF BF .UNID=0 THEN
  BEGIN
  SEEKREAD(NM,I);
  ERRCHK;
  IF ERROR THEN EXIT;
  IF ((DITO>=81) AND (DITO<=172)) THEN
    NM .R1:=NM .R1+BF .UVEN;
  IF ((DITO>=173) AND (DITO<=264)) THEN
    NM .R2:=NM .R2+BF .UVEN;
  IF ((DITO>=265) AND (DITO>=355)) THEN
    NM .R3:=NM .R3+BF .UVEN;
  IF ((DITO<81) OR (DITO>=356)) THEN
    NM .R4:=NM .R4+BF .UVEN;
  SEEKWRITE(NM,I);
  BF .UNID:=BF .UPED;
  BF .UPED:=0;
  BF .UVEN:=BF .UVP1;
  BF .UVP1:=0;
  BF .ANN:=BF .AA;
  BF .AA:=0;
  BF .MES:=BF .MM;
  BF .MM:=0;
  BF .DIA:=BF .DD;
  BF .DD:=0;
  SEEKWRITE(BF,I);
  END;
END;
END;

```

```

*****
*          PROGRAMA   PRINCIPAL          *
*****

```

```

BEGIN
  ASSIGN(BF,'B:MAESTRO.FIL');
  RESET(BF);
  ASSIGN(NM,'B:NONYEST.FIL');
  RESET(NM);
  SIG:=' ';
  REPEAT
    FECHAEX;
    WRITE('REALMENTE ESTA CORRECTA LA FECHA ? ANOTE S)I o N)O => ');
    READLN(SIG);
  UNTIL ((SIG='S')OR(SIG='s'));
  IF VER=1 THEN
    TRANS6;
  CLOSE(NM,I);
  CLOSE(BF,I);
  WRITELN;
  WRITELN('EL SISTEMA YA TERMINO.').
  WRITELN;
  WRITELN('PUEDE USTED CONTINUAR');
END;
NODEND.

```

V.2.8 LISTADO FUENTE DEL PROGRAMA:
< M O D U 7 >

```
*****  
*                                     *  
*           PROGRAMA           MODU7           *  
*                                     *  
*****
```

MODULE MODU7;

TYPE

```
ARCHII=RECORD  
  CLAVE:INTEGER;  
  MEDS1:INTEGER;  
  MEDS2:INTEGER;  
  MEDS3:INTEGER;  
  PRECIO:REAL;  
  UNID:INTEGER;  
  IMP:REAL;  
  UVEN:INTEGER;  
  UPE1:INTEGER;  
  UVPD:INTEGER;  
  END;
```

VAR

```
VA:EXTERNAL FILE OF ARCHII; (*ARCHIVO DE TRABAJO*)  
ERROR:EXTERNAL BOOLEAN;
```

EXTERNAL PROCEDURE ERRCHK;

```
*****  
*   PROCEDURE  ACTUALI7  *  
*****
```

```
PROCEDURE ACTUALI7;
```

```
VAR  
    CLASECP,I,X,UNI,AXMXD,ACEP,LIMIT7:INTEGER;  
    CLAPRIP:REAL;  
    OTRQ:CHAR;
```

```
*****  
*   PROCEDURE  CABEZA7  *  
*****
```

```
PROCEDURE CABEZA7;
```

```
BEGIN
```

```
    WRITELN(' LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES :');  
    WRITELN('=====');  
    WRITELN(' C           A           M           P           O');  
    WRITELN(' E           N           O           H           B           R           E');  
    WRITELN('=====');  
    WRITELN(' 1 UNIDADES EN EXISTENCIA ACTUALES');  
    WRITELN(' 2 UNIDADES VENDIDAS ACTUALES');  
    WRITELN(' 3 UNIDADES PEDIDAS');  
    WRITELN(' 4 UNIDADES VENDIDAS DEL PEDIDO');  
    WRITELN(' 5 NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)');  
    WRITELN(' 6 VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)');  
    WRITELN('=====');  
    WRITELN;  
END;
```

```
*****  
*   PROCEDURE  CHECA7  *  
*****
```

```
PROCEDURE CHECA7;  
BEGIN  
  SEEKREAD(VA,I);  
  ERRCHK;  
  IF ERROR THEN EXIT;  
END;
```

```
*****  
*   PROCEDURE  ASIGNAR7  *  
*****
```

```
PROCEDURE ASIGNAR7(VAR AXMXD,CLASECA:INTEGER;VAR CLAPRIA:REAL);  
BEGIN  
  ASSIGN(VA,'B:TRABAJO.FIL');  
  RESET(VA);  
  I:=0;  
  CHECA7;  
  AXMXD:=VA .MEDS1*VA .MEDS2*VA .MEDS3;  
  LIMIT7:=VA .UNID;  
  CLAPRIA:=VA .IMP;  
  CLASECA:=VA .UPE1;  
  WRITELN('ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA : ');  
  WRITE('AND : ',VA .MEDS1:2,' MES : ',VA .MEDS2:2);  
  WRITELN(' DIA : ',VA .MEDS3:2);  
  WRITELN;  
  CLOSE(VA,I);  
  ASSIGN(VA,'B:TRABAJO.FIL');  
  RESET(VA);  
END;
```

```
*****
*   PROCEDURE   ACCES07   *
*****
```

```
PROCEDURE ACCES07(VAR CLASECC,ACEPC:INTEGER;VAR CLAPRIC:REAL);
VAR
    CLAPRIM,CLASECU:INTEGER;
BEGIN
    ACEPC:=2;
    WRITELN;
    WRITE('ANOTE LA CLAVE PRIMARIA DE SEGURIDAD   => ');
    READLN(CLAPRIM);
    WRITELN;
    WRITE('ANOTE LA CLAVE SECUNDARIA DE SEGURIDAD => ');
    READLN(CLASECU);
    WRITELN;
    IF ((CLAPRIM=CLAPRIC)AND(CLASECU=CLASECC)) THEN
        ACEPC:=1;
END;
```

```
*****
*   PROCEDURE   SIMODIF7  *
*****
```

```
PROCEDURE SIMODIF7;
BEGIN
    SEEKWRITE(VA,I);
    ERRCHK;
    IF ERROR THEN EXIT
END;
```

```
*****
*   PROCEDURE   MODIFI7A  *
*****
```

```
PROCEDURE MODIFI7A;
VAR
    TOTFUE:REAL;
BEGIN
    CASE X OF
    5:BEGIN
        WRITE('DATO VIEJO : ',VA .UVEN:5,' ANOTE EL DATO NUEVO : ');
        READLN(UNI);
        VA .UVEN:=UNI;
        END;
    6:BEGIN
        WRITE('DATO VIEJO : ',VA .PRECIO:10:2,' ANOTE EL NUEVO DATO : ');
        READLN(TOTFUE);
        VA .PRECIO:=TOTFUE;
        END;
    END;
```

```

ELSE
  BEGIN
  WRITELN('UNICAMENTE PUEDE CORREGIR LOS CAMPOS 5 Y 6 ');
  OTRO:='C'
  END;
END;
END;

```

```

*****
*  PROCEDURE  MODIFI7  *
*****

```

```

PROCEDURE MODIFI7;
BEGIN
  OTRO:='N';
  IF I>0 THEN
    BEGIN
      CASE X OF
        1:BEGIN
          WRITE('DATO VIEJO : ',VA .UNID:5,' , ANOTE EL NUEVO DATO : ');
          VA .UNID:=UNI
          END;
        2:BEGIN
          WRITE('DATO VIEJO : ',VA .UVEN:5,' , ANOTE EL NUEVO DATO : ');
          VA .UVEN:=UNI
          END;
        3:BEGIN
          WRITE('DATO VIEJO : ',VA .UPE1:5,' , ANOTE EL NUEVO DATO : ');
          VA .UPE1:=UNI
          END;
        4:BEGIN
          WRITE('DATO VIEJO : ',VA .UVPD:5,' , ANOTE EL NUEVO DATO : ');
          VA .UVPD:=UNI
          END
        ELSE
          BEGIN
            WRITELN('UNICAMENTE PUEDE CORREGIR LOS CAMPOS DEL 1 AL 4');
            OTRO:='C'
            END;
          END;
        END
      ELSE
        BEGIN
          MODIFI7A;
          IF OTRO<>'C' THEN
            BEGIN
              WRITELN;
              WRITE('ESTA CORRECTO ? ANOTE S/I O N/O => ');
              READLN(OTRO);
              IF ((OTRO='S')OR(OTRO='s')) THEN
                SIMODIF7;
              END;
            END;
          END;
        END;
      END;
    END;
  END;

```

```
*****
*   PROCEDURE   CUALCAM7   *
*****
```

```
PROCEDURE CUALCAM7;
```

```
BEGIN
```

```
  CHECA7;
```

```
  WRITELN;
```

```
  CABEZA7;
```

```
  WRITELN;
```

```
  REPEAT
```

```
    WRITE('CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => ');
```

```
    READLN(X);
```

```
    IF ((X>0)AND(X<=6)) THEN
```

```
      MODIFI7
```

```
    ELSE
```

```
      BEGIN
```

```
        WRITELN(' * * *   E R R O R   * * * ');
```

```
        WRITELN('NO EXISTE, EL CAMPO ',X:3,' ESTA FUERA DEL RANGO');
```

```
      END;
```

```
    WRITELN('DESEA CAMBIAR OTRO CAMPO DEL REGISTRO ',I:4,' ?');
```

```
    WRITE('ANOTE S)I O N)O => ');
```

```
    READLN(OTRO)
```

```
  UNTIL ((OTRO='N')OR(OTRO='n'));
```

```
END;
```

```
*****
*   PROCEDURE   CUALREG7   *
*****
```

```
PROCEDURE CUALREG7;
```

```
BEGIN
```

```
  OTRO:=' ';
```

```
  REPEAT
```

```
    WRITELN;
```

```
    WRITE('CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => ');
```

```
    READLN(I);
```

```
    IF ((I>=0)AND(I<=LIMIT7)) THEN
```

```
      CUALCAM7
```

```
    ELSE
```

```
      BEGIN
```

```
        WRITELN(' * * *   E R R O R   * * * ');
```

```
        WRITE('NO EXISTE EL REGISTRO ',I:3);
```

```
        WRITELN(' , ESTA FUERA DEL RANGO ( ',LIMIT7:5,' )');
```

```
        WRITELN;
```

```
      END;
```

```
    WRITELN('SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN> ');
```

```
    WRITE('EN CASO CONTRARIO, ANOTE LA LETRA Q => ');
```

```
    READLN(OTRO)
```

```
  UNTIL ((OTRO='Q')OR(OTRO='q'));
```

```
END;
```

```
*****
*          PROGRAMA      PRINCIPAL      *
*****
```

```
BEGIN
  ASIGNAR7(AXHXD,CLASECP,CLAPRIP);
  IF AXHXD=0 THEN
    WRITE('* * * TIENE CEROS LA FECHA DEL ARCHIVO DE');
    WRITELN(' TRABAJO !!! * * *');
  ELSE
    BEGIN
      WRITELN('ESTE ES EL ARCHIVO DE TRABAJO QUE USTED BUSCA T');
      WRITE(' ANOTE S)I o N)O => ');
      READLN(OTRO);
      IF ((OTRO='S')OR(OTRO='s')) THEN
        BEGIN
          ACCESO7(CLASECP,ACEP,CLAPRIP);
          IF ACEP=1 THEN
            CUALREG7
          ELSE
            WRITE('USTED NO ESTA AUTORIZADO PARA HACER');
            WRITELN(' ACTUALIZACIONES');
        END;
      END;
    END;
  I:=LIMIT7;
  CLOSE(VA,I);
END;

MODEND.
```

VI D E S C R I P C I O N D E
E J E M P L O S G E N E R A D O S
P O R L A E J E C U C I O N
D E L O S
P R O G R A M A S D E L S I S T E M A

VI.1 DESCRIPCION DE EJEMPLOS
GENERADOS POR LA EJECUCION
DE LOS PROGRAMAS DEL
SUBSISTEMA
DE
ACTUALIZACION
Y
CONSULTA
DE
REQUERIMIENTOS

1.- Página 623, en ésta se muestra un ejemplo de lo que sucede, cuando se intenta entrar al Subsistema, y en el manejador de discos lado B no se encuentran presentes los archivos de datos, necesarios para que funcione el programa.

En esta situación el programa le despliega un mensaje, indicándole que los archivos de datos no se encuentran en el lado B del manejador de discos y por lo tanto no le permite entrar al programa del Subsistema de Actualización y Consulta de Requerimientos.

Ante este problema se le sugiere que antes de intentar entrar al programa, verifique la existencia de los archivos de datos en el disco que vaya a meter en el lado B del manejador.

Los archivos de datos que deben existir para que el programa funcione correctamente son los siguientes; ARCHIVO MAESTRO (MAESTRO.FIL), ARCHIVO DE NOMBRES Y ESTADISTICAS (NOMYEST.FIL), ARCHIVO DE TRABAJO (TRABAJO.FIL), ARCHIVO DE PEDIDOS (PEDIDOS.FIL) y el ARCHIVO DE CLAVES DISPONIBLES (CVEDISP.FIL).

Sin estos archivos no podrá entrar al programa del Subsistema y le seguirá apareciendo el mismo mensaje de error.

2.- Página 624, en ésta se muestra un ejemplo en donde el programa le da la bienvenida y le solicita transmita la fecha del día en que está siendo consultado el Subsistema, en este ejemplo los datos son correctamente transmitidos, después el programa a petición de que se le solicita, despliega el menú de las funciones que se pueden ejecutar dentro del Subsistema, para este ejemplo la función que se le transmitió fue para terminar el proceso del programa del Subsistema.

3.- Página 625, en ésta se muestra un ejemplo de como el programa valida el dato del año, cuando éste se le es solicitado y como el programa no permite que continúe hasta no haber transmitido correctamente el dato, después solicita el mes y el día, y por último lista el menú de funciones del programa del Subsistema.

4.- Página 626, en ésta se muestra un ejemplo de como el programa valida el dato del mes, cuando éste se le es solicitado, y como el programa no le permite continuar hasta no haber transmitido correctamente el dato.

5.- Página 627, en ésta se muestra un ejemplo de como el programa valida el dato del día, de la fecha solicitada y como el programa no le permite continuar hasta no haber transmitido correctamente el dato.

6.- Página 628, en ésta se muestra un ejemplo de como el programa valida la letra de la función que se desea ejecutar, y como el programa no le permite continuar hasta no haber transmitido correctamente el dato.

7.- De la página 629 a la 637, se le muestran algunos ejemplos de lo que sucede cuando se intenta entrar en alguno de los programas pero resulta que los archivos de datos aun se encuentran vacíos.

8.- Página 638, en ésta se muestra un ejemplo de bienvenida al Subsistema y como el programa le solicita la función a ejecutar, y partir de aquí puede entrar a cualquier actividad del Subsistema siempre y cuando los archivos de datos contengan información.

9.- Página 639, en ésta se muestra un ejemplo de como el programa le despliega alguna información de un medicamento al cual sus unidades en existencia están abajo del porcentaje mínimo indicado. En este ejemplo se elabora el pedido tomando los datos proporcionados por el programa, además se le lista la cantidad que lleva pedida.

10.- Página 640, en ésta se muestra otro ejemplo de solicitud de pedido de medicamentos, proporcionando algunos datos para la elaboración del mismo.

11.- Página 641, en ésta se muestra un ejemplo de solicitud de pedidos de medicamentos en donde se proporcionan todos los datos para la elaboración del mismo.

12.- Página 642, en ésta se muestra un ejemplo de solicitud de pedidos de medicamentos por temporada, en donde primeramente el programa le despliega el menú de las temporadas que puede consultar, para luego solicitarle el número de unidades que para usted representan una buena venta en la temporada elegida, y con esto listarle aquellos medicamentos que cumplan con la cantidad de unidades vendidas en dicha temporada y así poder elaborar algunos pedidos. En este ejemplo se proporcionan algunos datos para la elaboración del mismo.

13.- Página 643, en ésta se muestra otro ejemplo de como el programa solicita los datos para la elaboración del pedido del medicamento listado.

14.- Págsina 644, en ésta se muestra un ejemplo de un reporte de pedidos de medicamentos, el cual es generado por proveedor.

15.- Págsina 645, en ésta se muestra otro ejemplo de reporte de pedidos de medicamentos.

16.- Págsina 646, en ésta se muestra un ejemplo, de un reporte de medicamentos que están próximos a caducar dentro del periodo indicado. En este ejemplo el programa le solicita primeramente la fecha que desea lleve el encabezado del reporte, luego le solicita el número de días del periodo, y a partir de este número el programa genera el reporte de los medicamentos que caen dentro de este periodo.

17.- Págsina 647, en ésta se muestra otro ejemplo de reporte de medicamentos que están próximos a caducar, pero con la diferencia de que en este reporte la fecha transmitida estuvo incorrecta y por lo tanto la vuelve a solicitar el programa.

18.- Págsina 648, en ésta se muestra un ejemplo de como dar de baja un medicamento que ya no conviene tener dentro del Sistema, este proceso es generado con algunos errores, para que se visualicen todas las posibles alternativas que proporciona este programa.

19.- Págsina 649, en ésta se muestra otro ejemplo del programa para dar de baja algún medicamento.

20.- Págsina 650, en ésta se muestra un ejemplo del proceso que se sigue, para dar de alta algún medicamento, que se desea tener dentro del Sistema.

21.- Págsina 651, en ésta se muestra un ejemplo del programa para dar de alta algún medicamento, en éste, se presenta un caso en que la clave que se desea dar de alta no es la correcta y el programa le vuelve a solicitar transmita nuevamente la clave, el proceso posterior es igual al del ejemplo anterior.

22.- Págsina 652, en ésta se muestra un ejemplo del programa de listar información, en éste listado se le presenta el proceso para listar todos los medicamentos contenidos dentro del Sistema.

23.- Página 653, en ésta se muestra otro ejemplo de como listar la información contenida dentro del Sistema, en éste listado se le presenta como el programa valida la opción para listar información y además se muestra la opción para listar uno a uno la información de los medicamentos.

24.- Página 654, en ésta se muestra la opción para listar información de una clave inicial a otra clave final, además se le presenta, como el programa valida estas dos claves para poder generar el listado de la información de los medicamentos solicitados.

25.- Página 655, en ésta se muestra un ejemplo del programa para actualizar información, de datos del medicamento, éste listado le presenta el proceso que sigue el programa para poder permitir actualizar información y como éste le solicita los datos para ser actualizados.

26.- Página 656, en ésta se muestra otro ejemplo del programa de actualización de información, pero mostrándole en el proceso lo importante que resulta, proporcionar las claves de autorización de datos.

27.- Página 657, en ésta se muestra un ejemplo del programa de carga de unidades pedidas, en donde se le presenta el proceso que se sigue para este fin.

28.- Página 658, en ésta se muestran varios ejemplos de como cargar unidades pedidas a medicamentos que requieran de este proceso.

29.- Página 659, en ésta se muestra un ejemplo del programa, para obtener un listado de las ventas por temporada de los medicamentos, así también se le muestra, como el programa valida el número del reporte que se desea obtener.

30.- Página 660, en ésta se muestra un ejemplo del programa, para obtener un listado de precios, descuentos y número de proveedor, por medicamento.

SABADO

**** ERROR EL DISCO NO CONTIENE EL ARCHIVO MAESTRO DE DATOS ****

A.

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:19 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 1

DEME DIA => 20

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => K

*** TERMINA PROGRAMA DE ACTUALIZACION Y CONSULTA ***
*** DE REQUERIMIENTOS ***

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:19 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 87

ERROR : FAVOR DE DAR EL A&O IGUAL AL DE LA ULTIMA CONSULTA
DEBIDO A QUE LA DIFERENCIA DE MESES SERA MAYOR A DOS

DEME A&O => 85

ERROR: EL A&O DEBE SER MAYOR O IGUAL AL DE LA FECHA DE LA
ULTIMA CONSULTA AL SUBSISTEMA Y NO MAYOR A UN A&O
DE DIFERENCIA, FAVOR DE VOLVER A TRANSMITIR EL A&O

DEME A&O => 86

DEME MES => 1

DEME DIA => 19

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:20 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 0

ERROR: EL MES 0 NO EXISTE, FAVOR DE VOLVER A TRANSMITIR
EL MES

DEME MES => 15

ERROR: EL MES 15 NO EXISTE, FAVOR DE VOLVER A TRANSMITIR
EL MES

DEME MES => 4

ERROR : EL MES DEBE SER MAYOR O IGUAL AL DE LA FECHA DE LA
ULTIMA CONSULTA AL SUBSISTEMA Y NO MAYOR A DOS ME-
SES DE DIFERENCIA, FAVOR DE VOLVER A DAR EL MES

DEME MES => -3

ERROR: EL MES -3 NO EXISTE, FAVOR DE VOLVER A TRANSMITIR
EL MES

DEME MES => 1

DEME DIA => 20

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:21 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 1

DEME DIA => 15

ERROR : EL DIA DEBE SER MAYOR O IGUAL AL DE LA FECHA DE LA
ULTIMA CONSULTA AL SUBSISTEMA, FAVOR DE VOLVER A
TRANSMITIR EL DIA

DEME DIA => 0

ERROR : EL DIA 0 DEL MES 1 NO EXISTE FAVOR DE VOLVER A
TRANSMITIR EL DIA

DEME DIA => 32

ERROR : EL DIA 32 DEL MES 1 NO EXISTE FAVOR DE VOLVER A
TRANSMITIR EL DIA

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

[628]

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => Z

*** E R R O R ***

TRANSMITA UNICAMENTE DE LA <A> A LA <K>

DEME LA FUNCION QUE DESEA => 1

*** E R R O R ***

TRANSMITA UNICAMENTE DE LA <A> A LA <K>

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => A

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

*** TERMINA PROGRAMA DE SUGERIR PEDIDOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => B

*** ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION ***
*** FAVOR DE VOLVER A LISTAR EL MENU Y EJECUTE LA OPCION ***
*** <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO ***

*** TERMINA PROGRAMA DE LISTAR PEDIDOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => F
ERROR TRANSMITA UNICAMENTE <S> O <N>

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => C

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

**** TERMINA PROGRAMA DE PEDIDOS POR TEMPORADA ****

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => D

ERROR EL ARCHIVO MAESTRO NO CONTIENE INFORMACION.
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA
LA OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => E

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

*** TERMINA PROGRAMA DE BAJA DE MEDICAMENTOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => G

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

*** TERMINA PROGRMA DE LISTAR INFORMACION ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION OPERACION

A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => H

ERROR EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION,
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 A&O:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&O => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => I

ERROR : EL ARCHIVO MAESTRO AUN NO CONTIENE INFORMACION
FAVOR DE VOLVER A LISTAR EL MENU Y TRANSMITA LA
OPCION <F>, PARA DAR DE ALTA ALGUN MEDICAMENTO

*** TERMINA PROGRAMA DE CARGA DE UNIDADES PEDIDAS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **
** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:22 MES: 1 AÑO:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME AÑO => 86

DEME MES => 1

DEME DIA => 22

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION	OPERACION
A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA => J

ERROR : EL ARCHIVO DE NOMBRES Y ESTADISTICAS NO CONTIENE
INFORMACION, FAVOR DE VOLVER A LISTAR EL MENU Y
TRANSMITA LA OPCION <F>, PARA DAR DE ALTA ALGUN
MEDICAMENTO

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA =>

SABADO

*** BUENAS NOCHES ***

** ACABA DE ENTRAR A UNA NUEVA SESION DEL SUBSISTEMA DE **

** ACTUALIZACION Y CONSULTA DE REQUERIMIENTOS **

** FECHA DEL ULTIMO DIA DE ACCESO, DIA:31 MES: 1 A&D:86 **

** A CONTINUACION SE LE PIDE LA FECHA DE HOY **

DEME A&D => 86

DEME MES => 1

DEME DIA => 31

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => S

FUNCION OPERACION

A	SUGERIR PEDIDOS
B	LISTAR PEDIDOS
C	SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D	CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A DARSE DE BAJA POR CADUCIDAD
E	DAR DE BAJA ALGUN ARTICULO
F	DAR DE ALTA ALGUN ARTICULO
G	LISTAR LA INFORMACION
H	ACTUALIZAR INFORMACION
I	CARGA DE UNIDADES PEDIDAS
J	LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K	SALIRSE DEL SISTEMA

DEME LA FUNCION QUE DESEA =>

A

NOMBRE DEL MEDICAMENTO : NEOMELUBRINA

CLAVE	UNIDADES	UNIDADES	DIAS DE VENTA TRANSCURRIDOS		
	EN EXIT. (31/ 1/86)	QUE HABIA (7/12/85)	DE ESTA CANTIDAD DE UNIDADES		
1	2	15	56 DIAS		
PRECIO X UNIDAD			DESCUENTO DEL PROVEEDOR		
PROVEEDOR1	PROVEEDOR2	PROVEEDOR3	1	2	3
\$ 450.00	\$ 1200.00	\$ 1100.00	10.0 %	5.0 %	1.0 %

CANTIDAD DE MEDICAMENTOS
QUE SE RECOMIENDAN COMPRAR
13

ACTUALMENTE ESTE ARTICULO
ES SURTIDO POR EL PROVEEDOR
1

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O => S
DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD
SUGERIDA?, TRANSMITA S)I O N)O => S
GRABA PEDIDO

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO?, TRANSMITA S)I O N)O => S
HASTA AHORA LLEVA PEDIDO.- \$ 5265.00

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

NOMBRE DEL MEDICAMENTO : RUMAETS

CLAVE	UNIDADES EN EXIT. (31/ 1/86)	UNIDADES QUE HABIA (24/ 2/86)	DIAS DE VENTA TRANSCURRIDOS DE ESTA CANTIDAD DE UNIDADES		
3	3	25	36 DIAS		
PRECIO X UNIDAD			DESCUENTO DEL PROVEEDOR		
PROVEEDOR1	PROVEEDOR2	PROVEEDOR3	2	3	1
\$ 650.00	\$ 700.00	\$ 700.00	10.0 %	12.0 %	6.0 %

CANTIDAD DE MEDICAMENTOS
QUE SE RECOMIENDAN COMPRAR
22

ACTUALMENTE ESTE ARTICULO
ES SURTIDO POR EL PROVEEDOR
2

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O => S
DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD
SUGERIDA?, TRANSMITA S)I O N)O => n

A QUE PROVEEDOR DESEA HACER EL PEDIDO?
DEME NUMERO DE PROVEEDOR => 2

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O => n

DESEA DAR EL NUMERO DE MEDICAMENTOS A COMPRAR?
TRANSMITA, S)I O N)O => s

CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO? => 15

DESEA DAR EL PRECIO X MEDICAMENTO?, TRANSMITA S)I O N)O => s
DEME PRECIO POR UNIDAD => 650

DESEA DAR EL DESCUENTO DEL PROVEEDOR?, TRANSMITA S)I O N)O => N
SE TOMA EL DESC. DEL PROVEEDOR ESCOGIDO

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO?, TRANSMITA S)I O N)O => s
HASTA AHORA LLEVA PEDIDO.- \$ 14040.00

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

S

NOMBRE DEL MEDICAMENTO : VITACILINA

CLAVE	UNIDADES		DIAS DE VENTA TRANSCURRIDOS DE ESTA CANTIDAD DE UNIDADES		
	EN EXIT. (31/ 1/86)	QUE HABIA (7/12/85)			
4	3	15	56 DIAS		
PRECIO X UNIDAD			DESCUENTO DEL PROVEEDOR		
PROVEEDOR1	PROVEEDOR2	PROVEEDOR3	1	2	3
\$ 300.00	\$ 250.00	\$ 300.00	10.0 %	8.0 %	15.0 %

CANTIDAD DE MEDICAMENTOS
QUE SE RECOMIENDAN COMPRAR
12

ACTUALMENTE ESTE ARTICULO
ES SURTIDO POR EL PROVEEDOR
1

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O => S
DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD
SUGERIDA?, TRANSMITA S)I O N)O => N

A QUE PROVEEDOR DESEA HACER EL PEDIDO?
DEME NUMERO DE PROVEEDOR => 1

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O => S
CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO? => 10

DEME PRECIO POR UNIDAD => 300

DEME EL DESCUENTO QUE LE HACE EL PROVEEDOR => 10

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO?, TRANSMITA S)I O N)O => S
HASTA AHORA LLEVA PEDIDO.- \$ 19665.00

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

C

TEMPORADA	PERIODO
A	DEL 21 DE MARZO AL 20 DE JUNIO
B	DEL 21 DE JUNIO AL 23 DE SEPTIEMBRE
C	DEL 24 DE SEPTIEMBRE AL 22 DE DICIEMBRE
D	DEL 23 DE DICIEMBRE AL 20 DE MARZO

DEME TEMPORADA QUE DESEA => D

DEME EL NUM DE UNIDADES QUE PARA USTED REFLEJEN
UNA BUENA VENTA => 20

CLAVE	NOMBRE DEL MEDICAMENTO	UNIDADES VENDIDAS EN ESTA TEMPORADA	UNIDADES EN EXISTENCIA
2	TETRACILINA	40	2

CANTIDAD DE MEDICAMENTOS QUE SE RECOMIENDAN COMPRAR 13	ACTUALMENTE ESTE ARTICULO ES SURTIDO POR EL PROVEEDOR 2
--	---

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O => S
DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD
SUGERIDA?, TRANSMITA S)I O N)O => N

A QUE PROVEEDOR DESEA HACER EL PEDIDO?
DEME NUMERO DE PROVEEDOR => 2

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O => N

DESEA DAR EL NUMERO DE MEDICAMENTOS A COMPRAR?
TRANSMITA, S)I O N)O => N
SE TOMA LA CANTIDAD SUGERIDA

DESEA DAR EL PRECIO X MEDICAMENTO?, TRANSMITA S)I O N)O => N
SE TOMA EL PRECIO DEL PROVEEDOR ESCOGIDO

DESEA DAR EL DESCUENTO DEL PROVEEDOR?, TRANSMITA S)I O N)O => N
SE TOMA EL DESC. DEL PROVEEDOR ESCOGIDO

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO?, TRANSMITA S)I O N)O => S
HASTA AHORA LLEVA PEDIDO.- \$ 11700.00

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

S

CLAVE	NOMBRE DEL MEDICAMENTO	UNIDADES VENDIDAS EN ESTA TEMPORADA	UNIDADES EN EXISTENCIA
4	VITACILINA	25	3

CANTIDAD DE MEDICAMENTOS
QUE SE RECOMIENDAN COMPRAR
12

ACTUALMENTE ESTE ARTICULO
ES SURTIDO POR EL PROVEEDOR
1

DESEA HACER UN PEDIDO DE ESTE MEDICAMENTO? S)I O N)O => S
DESEA HACER EL PEDIDO CON EL PROVEEDOR Y CANTIDAD
SUGERIDA?, TRANSMITA S)I O N)O => N

A QUE PROVEEDOR DESEA HACER EL PEDIDO?
DEME NUMERO DE PROVEEDOR => 1

DESEA DAR TODOS LOS DATOS DEL PEDIDO?, TRANSMITA S)I O N)O => N

DESEA DAR EL NUMERO DE MEDICAMENTOS A COMPRAR?
TRANSMITA, S)I O N)O => S

CUANTAS UNIDADES DESEA PEDIR DE ESTE ARTICULO? => 12

DESEA DAR EL PRECIO X MEDICAMENTO?, TRANSMITA S)I O N)O => N
SE TOMA EL PRECIO DEL PROVEEDOR ESCOGIDO

DESEA DAR EL DESCUENTO DEL PROVEEDOR?, TRANSMITA S)I O N)O => N
SE TOMA EL DESC. DEL PROVEEDOR ESCOGIDO

DESEA CONOCER EL MONTO DE LO QUE LLEVA HASTA
AHORA PEDIDO?, TRANSMITA S)I O N)O => N

DESEA CONTINUAR HACIENDO PEDIDOS?, TRANSMITA S)I O N)O =>

REPORTE DE PEDIDOS DE MEDICAMENTOS DEL DIA 31 DE ENERO DE 1986

PROVEEDOR : JORGE GONZALEZ

CLAVE	NOMBRE DEL ARTICULO	NUM DE UNID.	PRECIO X UNIDAD	DESCUENTO	PAGO NETO
1	NEOMELUBRINA	13	\$ 450.00	10.0 %	\$ 5265.00
4	VITACILINA	12	\$ 300.00	10.0 %	\$ 3240.00
5	SULFATIACINA	13	\$ 670.00	10.0 %	\$ 7839.00
6	DARALG	13	\$ 300.00	10.0 %	\$ 3510.00
7	ASPIRINA	22	\$ 200.00	10.0 %	\$ 3960.00
TOTAL DEL PEDIDO A ESTE PROVEEDOR			\$ 23814.00		

PROVEEDOR : GERARDO SANCHES

CLAVE	NOMBRE DEL ARTICULO	NUM DE UNID.	PRECIO X UNIDAD	DESCUENTO	PAGO NETO
2	TETRACILINA	13	\$ 1000.00	10.0 %	\$ 11700.00
3	RUMALET	22	\$ 650.00	10.0 %	\$ 12870.00
TOTAL DEL PEDIDO A ESTE PROVEEDOR.-			\$ 24570.00		

PROVEEDOR : MAURICIO ESTRADA

CLAVE	NOMBRE DEL ARTICULO	NUM DE UNID.	PRECIO X UNIDAD	DESCUENTO	PAGO NETO
8	MANIDE-C	23	\$ 80.00	10.0 %	\$ 1656.00
TOTAL DEL PEDIDO A ESTE PROVEEDOR.-			\$ 1656.00		

B

REPORTE DE PEDIDOS DE MEDICAMENTOS DEL DIA 31 DE ENERO DE 1986

PROVEEDOR : JORGE GONZALEZ

CLAVE	NOMBRE DEL ARTICULO	NUM DE UNID.	PRECIO X UNIDAD	DESCUENTO	PAGO NETO
1	NEOMELUBRINA	13	\$ 450.00	10.0 %	\$ 5265.00
TOTAL DEL PEDIDO A ESTE PROVEEDOR.-			\$ 5265.00		

PROVEEDOR : GERARDO SANCHES

CLAVE	NOMBRE DEL ARTICULO	NUM DE UNID.	PRECIO X UNIDAD	DESCUENTO	PAGO NETO
2	TETRACILINA	13	\$ 1000.00	10.0 %	\$ 11700.00
TOTAL DEL PEDIDO A ESTE PROVEEDOR.-			\$ 11700.00		

N

DEME LA FUNCION QUE DESEA => D

FAVOR DE TRANSMITIR LA FECHA DE HOY :

DEME ANIO => 86

DEME MES => 2

DEME DIA => 24

ANOTO LA FECHA CORRECTAMENTE AÑO 86 MES 2 DIA 24

RELACION DE MEDICAMENTOS QUE CADUCARAN EN UN PERIODO NO MAYOR DE N DIAS

ANTE EL VALOR DE N : 1000

FARMACIA DE DIOS, S. A.

FECHA : 86 / 2 / 24

RELACION DE MEDICAMENTOS QUE CADUCARAN EN UN PERIODO NO MAYOR DE 1000 DIAS

```

=====
CLAVE . NOMBRE DEL MEDICAMENTO . UNIDADES ACTUALES : UNIDADES PEDIDAS
CLAVE DEL :UNIDADES EN :FECHA DE :FALTAN : CLAVE DEL :UNIDADES :FECHA DE :FALTAN
PROVEEDOR :EXISTENCIA :CADUCIDAD :X DIAS : PROVEEDOR :PEDIDAS :CADUCIDAD :X DIAS
=====

```

CLAVE	NOMBRE DEL MEDICAMENTO	UNIDADES	ACTUALES	UNIDADES	PEDIDAS				
CLAVE DEL	PROVEEDOR	EXISTENCIA	FECHA DE CADUCIDAD	FALTAN X DIAS	CLAVE DEL	PROVEEDOR	PEDIDAS	FECHA DE CADUCIDAD	FALTAN X DIAS
1	NEOMELUBRINA	1	2	86/ 1/ 3	-51				
2	TETRACILINA	2	2	87/ 4/30	426				
3	RUMALET	2	3	87/ 6/30	486				
4	VITACILINA	1	3	87/ 6/30	486				
5	SULFATIACINA	1	2	87/ 8/30	546				
6	BARALG	1	2	86/ 6/30	126				
7	ASPIRINA	1	3	87/ 1/25	331				
8	MANIBE-C	3	2	87/ 1/31	337				

EN 1000 DIAS, CADUCARAN 8 MEDICAMENTOS.

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

DEME LA FUNCION QUE DESEA => D

FAVOR DE TRANSMITIR LA FECHA DE HOY :
DEME ANIO => 86

DEME MES => 13

DEME DIA => 24

***** E R R O R *****
ANOTE LA FECHA NUEVAMENTE

SI DESEA CONTINUAR OPRIMA LA TECLA <RETURN>
SI DESEA SALIRSE TRANSMITA LA LETRA <Q> =>

DEME ANIO => 86

DEME MES => 2

DEME DIA => 24

ANOTO LA FECHA CORRECTAMENTE AÑO 86 MES 2 DIA 24
RELACION DE MEDICAMENTOS QUE CADUCARAN EN UN PERIODO NO MAYOR DE N DIAS
ANOTE EL VALOR DE N : 300

FARMACIA DE DIOS, S. A.

FECHA : 86 / 2 / 24

RELACION DE MEDICAMENTOS QUE CADUCARAN EN UN PERIODO NO MAYOR DE 300 DIAS

```

=====
CLAVE  NOMBRE  DEL  MEDICAMENTO .  UNIDADES  ACTUALES  :  UNIDADES  PEDIDAS
      CLAVE DEL :UNIDADES EN :FECHA DE :FALTAN  :  CLAVE DEL :UNIDADES :FECHA DE :FALTAN
      PROVEEDOR :EXISTENCIA  :CADUCIDAD :X DIAS  :  PROVEEDOR :PEDIDAS  :CADUCIDAD :X DIAS
=====
1  NEOMELUBRINA          1          2      86/ 1/ 3    -51
6  BARALG                 1          2      86/ 6/30    126
=====

```

EN 300 DIAS, CADUCARAN 2 MEDICAMENTOS.

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

FUNCION OPERACION

```

A      SUGERIR PEDIDOS
B      LISTAR PEDIDOS
C      SUGERIR COMPRA DE ARTICULOS POR TEMPORADA
D      CONSULTA DE ARTICULOS QUE ESTAN PROXIMOS A
        DARSE DE BAJA POR CADUCIDAD
E      DAR DE BAJA ALGUN ARTICULO
F      DAR DE ALTA ALGUN ARTICULO
G      LISTAR LA INFORMACION
H      ACTUALIZAR INFORMACION
I      CARGA DE UNIDADES PEDIDAS
J      LISTA RANGOS DE TEMPORADAS Y LISTA PROVEEDORES
K      SALIRSE DEL SISTEMA
    
```

DEME LA FUNCION QUE DESEA => E

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 3

** ERROR: CLAVE PUESTA DISPONIBLE YA ANTERIORMENTE **

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 2

ES CORRECTA LA CLAVE DEL MEDICAMENTO : TETRACILINA
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O => N

DESEA DAR DE BAJA OTRO MEDICAMENTO?, TRANSMITA S)I O N)O => S

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 9

ERROR SE EXCEDE DE LA CLAVE MAXIMA EXISTENTE
LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES LA: 8

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 1

ES CORRECTA LA CLAVE DEL MEDICAMENTO : NEOMELUBRINA
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O => N

DESEA DAR DE BAJA OTRO MEDICAMENTO?, TRANSMITA S)I O N)O => N

DESEA LISTAR LAS CLAVES QUE SE ENCUENTRAN
DISPONIBLES?, TRANSMITA S)I O N)O => S

CLAVES DISPONIBLES

• 3

• 6

*** TERMINA PROGRAMA DE BAJA DE MEDICAMENTOS ***

C 649 J

N

DEME LA FUNCION QUE DESEA => E

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 0
ERROR LAS CLAVES INICIAN DESDE LA 00001

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 24

ERROR SE EXCEDE DE LA CLAVE MAXIMA EXISTENTE
LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES LA: 13

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 4

ES CORRECTA LA CLAVE DEL MEDICAMENTO : VITACILINA
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O => N

DESEA DAR DE BAJA OTRO MEDICAMENTO?, TRANSMITA S)I O N)O => S

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 12

ES CORRECTA LA CLAVE DEL MEDICAMENTO : DEXTROPROPOXIFEDO
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O => S

DESEA DAR DE BAJA OTRO MEDICAMENTO?, TRANSMITA S)I O N)O => S

DEME LA CLAVE DEL MEDICAMENTO QUE DESEA DAR DE BAJA => 7

ES CORRECTA LA CLAVE DEL MEDICAMENTO : ASPIRINA
SE PUEDE PONER DISPONIBLE?, TRANSMITA S)I O N)O => N

DESEA DAR DE BAJA OTRO MEDICAMENTO?, TRANSMITA S)I O N)O => N

DESEA LISTAR LAS CLAVES QUE SE ENCUENTRAN
DISPONIBLES?, TRANSMITA S)I O N)O => S

CLAVES DISPONIBLES

* 12

*** TERMINA PROGRAMA DE BAJA DE MEDICAMENTOS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

F

ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION :

CLAVE DEL MEDICAMENTO => # 9

NOMBRE DEL MEDICAMENTO => CIMETASE

CLAVE DEL PROVEEDOR PRINCIPAL => # 3

PRECIO DE COMPRA DEL PROVEEDOR PRINCIPAL => \$ 850

DESCUENTO DE COMPRA DEL PROVEEDOR PRINCIPAL => % 10

CLAVE DEL SEGUNDO PROVEEDOR => # 1

PRECIO DE COMPRA DEL SEGUNDO PROVEEDOR => \$ 900

DESCUENTO DE COMPRA DEL SEGUNDO PROVEEDOR => % 9

CLAVE DEL TERCER PROVEEDOR => # 2

PRECIO DE COMPRA DEL TERCER PROVEEDOR => \$ 100

DESCUENTO DE COMPRA DEL TERCER PROVEEDOR => % 10

CLAVE DE MEDICINA SIMILAR 1 => # 1

CLAVE DE MEDICINA SIMILAR 2 => # 5

CLAVE DE MEDICINA SIMILAR 3 => # 12

EL PRECIO DE VENTA DEL MEDICAMENTO => \$ 1500

EL NUMERO DE UNIDADES EN EXISTENCIA => 20

FECHA DE CADUCIDAD DEL MEDICAMENTO

DAME ANIO => 87

DAME MES => 3

DAME DIA => 30

DEME EL IMPUESTO DE VENTA DEL MEDICAM. => % 6

DEME EL PORCENTAJE DE UNIDADES MINIMAS
QUE DEBEN EXISTIR => % 20

TERMINO DE GRABAR EL MEDICAMENTO CLAVE # 9

SI DESEA SEGUIR METIENDO MAS INFORMACION OPRIMA LA TECLA
<RETURN>, EN CASO CONTRARIO TRANSMITA LA LETRA <Q> =>

ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION :

CLAVE DEL MEDICAMENTO => # 0
ERROR EL SIGUIENTE REGISTRO DISPONIBLE ES EL 12

SI DESEA SEGUIR METIENDO MAS INFORMACION OPRIMA LA TECLA
<RETURN>, EN CASO CONTRARIO TRANSMITA LA LETRA <Q> =>
ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION :

CLAVE DEL MEDICAMENTO => # 25
ERROR EL SIGUIENTE REGISTRO DISPONIBLE ES EL 12

SI DESEA SEGUIR METIENDO MAS INFORMACION OPRIMA LA TECLA
<RETURN>, EN CASO CONTRARIO TRANSMITA LA LETRA <Q> =>
ANOTE LOS DATOS QUE SE PIDEN A CONTINUACION :

CLAVE DEL MEDICAMENTO => # 12

NOMBRE DEL MEDICAMENTO => DEXTROPROPOXIFEDO

CLAVE DEL PROVEEDOR PRINCIPAL => # 2

PRECIO DE COMPRA DEL PROVEEDOR PRINCIPAL => \$ 500

DESCUENTO DE COMPRA DEL PROVEEDOR PRINCIPAL => % 9

CLAVE DEL SEGUNDO PROVEEDOR => # 3

PRECIO DE COMPRA DEL SEGUNDO PROVEEDOR => \$ 600 .

DESCUENTO DE COMPRA DEL SEGUNDO PROVEEDOR => % 8

CLAVE DEL TERCER PROVEEDOR => # 1

PRECIO DE COMPRA DEL TERCER PROVEEDOR => \$ 650

DESCUENTO DE COMPRA DEL TERCER PROVEEDOR => % 8

CLAVE DE MEDICINA SIMILAR 1 => # 3

CLAVE DE MEDICINA SIMILAR 2 => # 5

CLAVE DE MEDICINA SIMILAR 3 => # 9

EL PRECIO DE VENTA DEL MEDICAMENTO => \$ 890

EL NUMERO DE UNIDADES EN EXISTENCIA => 20

FECHA DE CADUCIDAD DEL MEDICAMENTO

DAME ANIO => 87

DAME MES => 3

DAME DIA => 15

DEME EL IMPUESTO DE VENTA DEL MEDICAM. => % 6

DEME EL PORCENTAJE DE UNIDADES MINIMAS
QUE DEBEN EXISTIR => % 20

TERMINO DE GRABAR EL MEDICAMENTO CLAVE # 12

G

COMO DESEA LISTAR SU INFORMACION ?

DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO => T

CLAVE	MEDICAMENTOS SIMILARES			PRECIO X UNIDAD	NUM DE UNIDADES	FECHA DE CADUCIDAD	CLAVE DEL VENDEDOR	IMPUESTO %	PORCENTAJE MIN DE UNI.	UNIDADES VENDIDAS	UNIDADES PEDIDAS	FECHA CAD/PED
1	2/	6/	7	1500.00	2	3/ 1/86	1	6.0 %	20.0 %	13	0	0 0/ 0
2	4/	3/	1	1000.00	2	30/ 4/87	2	6.0 %	20.0 %	13	0	0 0/ 0
3	9/	1/	6	750.00	3	30/ 6/87	2	6.0 %	20.0 %	22	0	0/ 0/ 0
4	6/	8/	5	560.00	3	30/ 6/87	1	6.0 %	20.0 %	12	0	0. 0/ 0
5	7/	6/	3	1000.00	2	30/ 8/87	1	6.0 %	20.0 %	13	0	31/12 87
6	2/	9/	4	850.00	2	30/ 6/86	1	6.0 %	20.0 %	13	0	0/ 0/ 0
7	4/	9/	2	450.00	3	25/ 1/87	1	6.0 %	20.0 %	22	0	30/ 7/87
8	4/	6/	1	100.00	2	31/ 1/87	3	6.0 %	20.0 %	23	0	0/ 0/ 0

DESEA CONTINUAR LISTANDO INFORMACION ?

TRANSMITA S)I O N)O => N

*** TERMINA PROGRAMA DE LISTAR INFORMACION ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

G

COMO DESEA LISTAR SU INFORMACION ?

DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO => T

CLAVE	MEDICAMENTOS SIMILARES			PRECIO X UNIDAD	NUM DE UNIDADES	FECHA DE CADUCIDAD	CLAVE DEL VENDEDOR	IMPUESTO %	PORCENTAJE MIN DE UNI.	UNIDADES VENDIDAS	UNIDADES PEDIDAS	FECHA CAD/PED
1	2/	6/	7	1500.00	2	3/ 1/86	1	6.0 %	20.0 %	13	0	0 0/ 0
2	4/	3/	1	1000.00	2	30/ 4/87	2	6.0 %	20.0 %	13	0	0 0/ 0
3	9/	1/	6	750.00	3	30/ 6/87	2	6.0 %	20.0 %	22	0	0/ 0/ 0
4	6/	8/	5	560.00	3	30/ 6/87	1	6.0 %	20.0 %	12	0	0. 0/ 0
5	7/	6/	3	1000.00	2	30/ 8/87	1	6.0 %	20.0 %	13	0	31/12 87
6	2/	9/	4	850.00	2	30/ 6/86	1	6.0 %	20.0 %	13	0	0/ 0/ 0
7	4/	9/	2	450.00	3	25/ 1/87	1	6.0 %	20.0 %	22	0	30/ 7/87
8	4/	6/	1	100.00	2	31/ 1/87	3	6.0 %	20.0 %	23	0	0/ 0/ 0

DESEA CONTINUAR LISTANDO INFORMACION ?

TRANSMITA S)I O N)O => N

*** TERMINA PROGRAMA DE LISTAR INFORMACION ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

5

COMO DESEA LISTAR SU INFORMACION ?

DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO => Y

E R R O R

TRANSMITA UNICAMENTE LAS SIGUIENTES LETRAS/ U, T O R

COMO DESEA LISTAR SU INFORMACION ?

DE U)NO EN UNO, T)ODA, O DE UN R)ANGO A OTRO => U

TRANSMITA LA CLAVE DEL MEDICAMENTO, QUE DESEA CONSULTAR => 24

EL NUMERO DE LA CLAVE EXCEDE EL RANGO PERMITIDO
LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES LA: 8

FAVOR DE VOLVER A TRANSMITIR LA CLAVE => 4

CLAVE	MEDICAMENTOS SIMILARES			PRECIO X UNIDAD	NUM DE UNIDADES	FECHA DE CADUCIDAD	CLAVE DEL VENDEDOR	IMPUESTO %	PORCENTAJE MIN DE UNI.	UNIDADES VENDIDAS	UNIDADES PEDIDAS	FECHA CAD/PED
4	6/	8/	5	560.00	3	30/ 6/87	1	6.0 %	20.0 %	12	0	0/ 0/ 0

DESEA CONSULTAR LA INFORMACION DE OTRO MEDICAMENTO?,
TRANSMITA S)I O N)O => S

TRANSMITA LA CLAVE DEL MEDICAMENTO, QUE DESEA CONSULTAR => 7

CLAVE	MEDICAMENTOS SIMILARES			PRECIO X UNIDAD	NUM DE UNIDADES	FECHA DE CADUCIDAD	CLAVE DEL VENDEDOR	IMPUESTO %	PORCENTAJE MIN DE UNI.	UNIDADES VENDIDAS	UNIDADES PEDIDAS	FECHA CAD/PED
7	4/	9/	2	450.00	3	25/ 1/87	1	6.0 %	20.0 %	22	0	30/ 7/87

DESEA CONSULTAR LA INFORMACION DE OTRO MEDICAMENTO?,
TRANSMITA S)I O N)O => N

DESEA CONTINUAR LISTANDO INFORMACION ?
TRANSMITA S)I O N)O =>

S

COMO DESEA LISTAR SU INFORMACION ?

DE UNO EN UNO, TODA, O DE UN RANGO A OTRO => R

DEME CLAVE DE INICIO => 0

ERROR: LAS CLAVES INICIAN DESDE LA 00001

DEME CLAVE DE INICIO => 24

ERROR : SE HA EXCEDIDO DEL RANGO PERMITIDO, EL NUMERO
DE LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES LA : 8

DEME CLAVE DE INICIO => 3

DEME CLAVE FINAL => 2

ERROR: TRANSMITA UNA CLAVE MAYOR O IGUAL A LA INICIAL

DEME CLAVE FINAL => 24

ERROR : SE HA EXCEDIDO DEL RANGO PERMITIDO, EL NUMERO
DE LA CLAVE MAXIMA QUE CONTIENE INFORMACION ES LA : 8

DEME CLAVE FINAL => 6

CLAVE	MEDICAMENTOS SIMILARES			PRECIO X UNIDAD	NUM DE UNIDADES	FECHA DE CADUCIDAD	CLAVE DEL VENDEDOR	IMPUESTO %	PORCENTAJE MIN DE UNI.	UNIDADES VENDIDAS	UNIDADES PEDIDAS	FECHA CAD/PEU
3	9/	1/	6	750.00	3	30/ 6/87	2	6.0 %	20.0 %	22	0	0/ 0/ 0
4	6/	8/	5	560.00	3	30/ 6/87	1	6.0 %	20.0 %	12	0	0/ 0/ 0
5	7/	6/	3	1000.00	2	30/ 8/87	1	6.0 %	20.0 %	13	0	31/12/87
6	2/	9/	4	850.00	2	30/ 6/86	1	6.0 %	20.0 %	13	0	0/ 0/ 0

DESEA CONTINUAR LISTANDO INFORMACION ?
TRANSMITA S)I O N)O =>

ANOTE LA PRIMERA CONTRASENA : 10
 ANOTE LA SEGUNDA CONTRASENA : 20

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES:

#	C A M P O N O M B R E	#	C A M P O N O M B R E
1)	MEDICINA SIMILAR 1	14)	NOMBRE DEL PRODUCTO
2)	MEDICINA SIMILAR 2	15)	UNIDADES DEL RANGO 1
3)	MEDICINA SIMILAR 3	16)	UNIDADES DEL RANGO 2
4)	PRECIO	17)	UNIDADES DEL RANGO 3
5)	UNIDADES EN EXISTENCIA	18)	UNIDADES DEL RANGO 4
6)	PRIMERA CADUCIDAD ANO MES DIA	19)	CLAVE DEL PROVEEDOR PRINCIPAL
7)	PROVEEDOR PRINCIPAL	20)	PRECIO DEL PROVEEDOR PRINCIPAL
8)	IMPUESTO	21)	DESCUENTO DEL PROVEEDOR PRINCIPAL
9)	PORCENTAJE MINIMO DE UNID	22)	CLAVE DEL SEGUNDO PROVEEDOR
10)	UNIDADES VENDIDAS	23)	PRECIO DEL SEGUNDO PROVEEDOR
11)	UNIDADES PEDIDAS	24)	DESCUENTO DEL SEGUNDO PROVEEDOR
12)	UNID. VEND. DEL PEDIDO	25)	CLAVE DEL TERCER PROVEEDOR
13)	SEGUNDA CADUCIDAD ANO MES DIA	26)	PRECIO DEL TERCER PROVEEDOR
		27)	DESCUENTO DEL TERCER PROVEEDOR
		28)	FECHA DEL MOVIMIENTO

CUAL REGISTRO QUIERE MODIFICAR? ANOTE EL NUMERO : 3
 CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : 21
 DATO VIEJO : 10.00 ANOTE EL NUEVO DATO : 0.10

ESTA CORRECTO ? ANOTE S)I O N)O : S
 DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 3 ? ANOTE S)I O N)O : S
 CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : 24
 DATO VIEJO : 12.00 ANOTE EL NUEVO DATO : 0.12

ESTA CORRECTO ? ANOTE S)I O N)O : S
 DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 3 ? ANOTE S)I O N)O : S
 CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : 27
 DATO VIEJO : 6.00 ANOTE EL NUEVO DATO : 0.06

ESTA CORRECTO ? ANOTE S)I O N)O : S
 DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 3 ? ANOTE S)I O N)O : N
 SI DESEA SEGUIR ACTUALIZANDO OPRIMA LA TECLA <RETURN>
 EN CASO CONTRARIO TRANSMITA LA LETRA <Q> => Q

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA => K

*** TERMINA PROGRAMA DE ACTUALIZACION Y CONSULTA ***
 *** DE REQUERIMIENTOS ***

N

DEME LA FUNCION QUE DESEA => H

ANOTE LA PRIMERA CONTRASENA : 12
 ANOTE LA SEGUNDA CONTRASENA : 20
 USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O => N

DEME LA FUNCION QUE DESEA => H

ANOTE LA PRIMERA CONTRASENA : 10
 ANOTE LA SEGUNDA CONTRASENA : 20

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES:

C A M P O					C A M P O								
#	N	O	M	B	R	E	#	N	O	M	B	R	E
1)	M	E	D	I	C	I	14)	N	O	M	B	R	E
2)	M	E	D	I	C	I	15)	U	N	I	D	A	D
3)	M	E	D	I	C	I	16)	U	N	I	D	A	D
4)	P	R	E	C	I	O	17)	U	N	I	D	A	D
5)	U	N	I	D	A	D	18)	U	N	I	D	A	D
6)	P	R	I	M	E	R	19)	C	L	A	V	E	
	A	N	O	M	E	S	20)	P	R	E	C	I	O
7)	P	R	O	V	E	E	21)	D	E	S	C	U	E
8)	I	M	P	U	E	S	22)	C	L	A	V	E	
9)	P	O	R	C	E	N	23)	P	R	E	C	I	O
	M	I	N	I	M	O	24)	D	E	S	C	U	E
10)	U	N	I	D	A	D	25)	C	L	A	V	E	
11)	U	N	I	D	A	D	26)	P	R	E	C	I	O
12)	U	N	I	D	.	V	27)	D	E	S	C	U	E
13)	S	E	G	U	N	D	28)	F	E	C	H	A	
	A	N	O	M	E	S							

CUAL REGISTRO QUIERE MODIFICAR? ANOTE EL NUMERO : 6

CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : 8

DATO VIEJO : 0.00 ANOTE EL NUEVO DATO : 0.06

ESTA CORRECTO ? ANOTE S)I O N)O : 9

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 6 ? ANOTE S)I O N)O : 8

CUAL CAMPO DESEA ACTUALIZAR? ANOTE EL NUMERO : 9

DATO VIEJO : 0.00 ANOTE EL NUEVO DATO : 0.20

ESTA CORRECTO ? ANOTE S)I O N)O : 9

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 6 ? ANOTE S)I O N)O :

I

A CONTINUACION SE LE PIDEN LOS DATOS DE LOS MEDICAMENTOS, A
LOS CUALES SE LES DESEA CARGAR UNIDADES PEDIDAS

DEME CLAVE DEL MEDICAMENTO => 5

CLAVE DEL MEDICAMENTO : SULFATIACINA
DESEA CARGARLE NUEVAS UNIDADES ? S)I O N)O => S

DEME LA CANTIDAD DE UNIDADES PEDIDAS => 50

DEME EL PRECIO DE VENTA DEL MEDICAMENTO => 1200

DEME EL IMPUESTO DE VENTA DEL MEDICAMENTO => 6

DEME FECHA DE CADUCIDAD, EN ESTE ORDEN: AA MM DD
DONDE: AA ES EL A&O,MM ES EL MES Y DD EL DIA => 87 6 30

DEME CLAVE DEL PROVEEDOR PRINCIPAL DEL MEDICAMENTO => 1

DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O => S

DEME CLAVE DEL MEDICAMENTO => 7

CLAVE DEL MEDICAMENTO : ASPIRINA
DESEA CARGARLE NUEVAS UNIDADES ? S)I O N)O => N

DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O =>

DEME LA FUNCION QUE DESEA => I

A CONTINUACION SE LE PIDEN LOS DATOS DE LOS MEDICAMENTOS, A
LOS CUALES SE LES DESEA CARGAR UNIDADES PEDIDAS

DEME CLAVE DEL MEDICAMENTO => 24

ERROR: CLAVE DE MEDICAMENTO EXCEDE EL RANGO PERMITIDO
LA CLAVE MAXIMA DEL MEDICAMENTO QUE CONTIENE INFORMACION
ES LA : 13

DEME CLAVE DEL MEDICAMENTO => 0

ERROR: LA CLAVE 0 NO EXISTE

DEME CLAVE DEL MEDICAMENTO => B

CLAVE DEL MEDICAMENTO ; MANIBE-C
DESEA CARGARLE NUEVAS UNIDADES ? S)I O N)O => S

DEME LA CANTIDAD DE UNIDADES PEDIDAS => 40

DEME EL PRECIO DE VENTA DEL MEDICAMENTO => 350

DEME EL IMPUESTO DE VENTA DEL MEDICAMENTO => 6

DEME FECHA DE CADUCIDAD, EN ESTE ORDEN: AA MM DD
DONDE: AA ES EL AÑO, MM ES EL MES Y DD EL DIA => 87 5 15

DEME CLAVE DEL PROVEEDOR PRINCIPAL DEL MEDICAMENTO => 3

DESEA SEGUIR CARGANDO MAS INFORMACION? S)I O N)O => N

*** TERMINA PROGRAMA DE CARGA DE UNIDADES PEDIDAS ***

DESEA LISTAR EL MENU?, TRANSMITA S)I O N)O =>

N

DEME LA FUNCION QUE DESEA => J

CUAL LISTADO DESEA OBTENER:

- 1.- NOMBRE Y RANGOS
- 2.- PROVEDORES

ANOTE EL NUMERO DEL LISTADO QUE DESEA : 3
 ERROR : ANOTE UNICAMENTE 1 o 2
 DESEA SACAR OTRO LISTADO S)I o N)O: S

CUAL LISTADO DESEA OBTENER:

- 1.- NOMBRE Y RANGOS
- 2.- PROVEDORES

ANOTE EL NUMERO DEL LISTADO QUE DESEA : 1

=====

CLAVE . NOMBRE DEL MEDICAMENTO . RANGO 1 RANGO 2 RANGO 3 RANGO 4 FECHA DE TRABAJO

=====

1	NEOMELUBRINA	0	0	0	15	05/12/ 7
2	TETRACILINA	0	0	0	40	05/12/ 7
3	RUMALETS	0	0	0	0	04/ 2/24
4	VITACILINA	0	0	0	25	05/12/ 7
5	SULFATIACINA	0	0	0	22	05/12/ 7
6	BARALO	0	0	0	0	05/ 5/12
7	ASPIRINA	0	0	0	0	06/ 1/17
8	MANIBE-C	0	0	0	0	06/ 1/17

DESEA SACAR OTRO LISTADO S)I o N)O:

CUAL LISTADO DESEA OBTENER:

- 1.- NOMBRE Y RANGOS
- 2.- PROVEEDORES

ANOTE EL NUMERO DEL LISTADO QUE DESEA : 2

```
=====
```

CLAVE	NOMBRE DEL MEDICAMENTO	PROVEEDOR CLAVE	PRECIO	PRINCIPAL DESC.	SEGUNDO CLAVE	PRECIO	DESC.	TERCER CLAVE	PRECIO	DESC.
1	NEOMELUBRINA	1	450.00	0.10	2	1200.00	0.05	3	1100.00	0.01
2	TETRACILINA	2	1000.00	0.10	1	1000.00	0.06	3	1050.00	0.10
3	RUMALETS	2	650.00	0.10	3	700.00	0.12	1	700.00	0.06
4	VITACILINA	1	300.00	0.10	2	250.00	0.08	3	300.00	0.15
5	SULFATIACINA	1	670.00	0.10	1	360.00	0.09	3	370.00	0.02
6	BARALG	2	300.00	0.10	3	250.00	0.10	1	350.00	0.10
7	ASPIRINA	1	200.00	0.10	3	700.00	0.08	2	750.00	0.09
8	MANIBE-C	3	80.00	0.10	1	600.00	0.10	2	640.00	0.08

```
=====
```

DESEA SACAR OTRO LISTADO SÍ o NO:

VI.2 DESCRIPCION DE EJEMPLOS
GENERADOS POR LA EJECUCION
DE LOS PROGRAMAS DEL
SUBSISTEMA
DE
CONTROL
DE
VENTAS

A continuación aparece la explicación de los ejemplos más representativos que se generan por el funcionamiento del Subsistema 'CONTROL DE VENTAS'.

La explicación está formada por: la página donde se encuentra el ejemplo y una breve narración del mismo.

1.- Página 668. Cuando se inicia una sesión de trabajo, aparece un mensaje de bienvenida y los nombres de los archivos que deben estar presentes para que funcione el sistema y que están identificados como: Archivo Maestro, Archivo de Trabajo y Archivo de Nombres y Estadísticas. En este caso, no existe ningún archivo, por lo que se marca el error y se especifica en qué consiste. Como se observa, el sistema no contiene ningún medicamento. No se permite continuar con la sesión.

2.- Página 669. Si únicamente existe el archivo maestro pero los otros dos no, se especifica el error y no se permite continuar con la sesión. Es necesario que estén presentes los tres archivos. Cuando el error es muy grave como en este caso, el sistema lo saca de sesión para que verifique las condiciones mínimas de funcionamiento.

3.- Página 670. Cuando existen los tres archivos, pero el de trabajo está vacío, no se permite entrar a ninguna función, si antes no es llenado por medio de la función <A>. En este caso se pretende usar la función . Es inútil insistir en el acceso a una función y esto es lógico, ya que no se puede trabajar, si no existe el archivo con información. Por lo tanto el sistema le indica que el archivo de trabajo está vacío y que debe usar la función <A> para llenarlo y poder entrar de lleno al sistema. Mientras no contenga información el archivo de trabajo, no podrá hacer uso de alguna función del subsistema, excepto la <A>.

4.- Página 671. Mientras no esté lleno el archivo de trabajo, no se podrá usar ninguna función. En este ejemplo, se pretende usar la función <C>, pero el sistema no lo permite por lo explicado anteriormente.

5.- Págsina 672. Este ejemplo le muestra como trabaja la función <A> identificada en el menú principal de funciones como "DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO". Observe que el sistema le indica el inicio y final del proceso; además le solicita la fecha del archivo de trabajo, la cual será usada en todos los procesos del subsistema.

6.- Págsina 673. Qué sucede si pretende usar la función <A> cuando ya existe el archivo de trabajo con información. Simplemente el sistema no se lo permite, indicándole por medio de un mensaje de error, que no puede usar esa función porque el archivo ya contiene información.

7.- Págsina 674. Cuando ha sido aceptado para trabajar en el sistema, únicamente puede usar aquellas funciones que están especificadas en el menú. Si por error marca alguna letra que no corresponda con las funciones, el sistema lo detecta y especifica. Para este caso, se pretende usar la función <P> la cual no es reconocida por el sistema y por lo tanto no existe.

8.- Págsina 675. Ejemplo similar al anterior, solo que aquí se ha seleccionado a la función <S>, la cual no existe en el menú. Como se observa, después del mensaje de error el sistema pregunta si desea listar el menú principal de funciones. Esto es opcional, ya que para seleccionar la función no es necesario que aparezca el menú.

9.- Págsina 676. En este ejemplo se muestra el uso de la función para leer el archivo de trabajo del día. Observe que después de seleccionar la función, le presenta la fecha y además le pide que coloque el papel al inicio de una nueva hoja, si está trabajando en impresora. Si se encuentra en la pantalla, simplemente presione la tecla <RETURN> y continúe.

10.- Págsina 677. Ejemplo del reporte generado por la función . Primero despliega el encabezado del reporte, con su respectiva identificación. Después aparecen dos signos el de interrogación y el de número. Este es el momento para escribir el número o la clave del medicamento que desea consultar, cuya información aparecerá en el siguiente renglón. Esto se hace para cada medicina.

11.- Págsina 678. Se muestra otro ejemplo de la función . Como puede notarse, al momento que el sistema solicita la clave del medicamento a consultar, ésta puede anotarse sin seguir un orden, es decir, Usted puede solicitar la consulta de cualquier medicamento en el momento que guste y en el orden que quiera. La única condición es que exista.

12.- Págsina 679. El sistema únicamente reconoce números en la función . Así, cuando se anota una letra, se queda esperando hasta que se escriba un número. Lo mismo sucede si se escribe una palabra o cualquier otro signo diferente de un número. La función trabajará adecuadamente hasta que se anote un número que corresponda a la clave del medicamento que se desea consultar. Si se anota una clave que está fuera del rango, el sistema lo detecta y especifica. Para salir de la función , únicamente anote la letra Q después del listado del registro.

13.- Págsina 680. Este ejemplo ilustra el uso de la función <C> que sirve para elaborar facturas para el cliente. Una vez que se entra a la función, se despliega la fecha del archivo de trabajo y aparecen dos preguntas. En este ejemplo, el archivo si le sirve, pero no desea continuar, por lo que la respuesta a la segunda pregunta es negativa. Cuando una de las respuestas es negativa, se sale de la función.

14.- Págsina 681. Para este caso, después de ver la fecha del archivo, se contesta que no le sirve y que no desea continuar, por lo que el sistema sale de la función <C> y regresa al menú principal de funciones.

15.- Págsina 682. Después de seleccionar la función <C>, el sistema presenta la fecha del archivo. Una vez que éste ha sido acertado y que se desea continuar con el proceso, entonces es necesario colocar el papel al inicio de una nueva hoja y presionar <RETURN>.

16.- Página 683. Aquí se muestra el ejemplo de una factura. Primero aparece el encabezado de la misma. Recuerde que la factura tiene dos secciones: una para el usuario y otra para el cliente. En la sección del usuario, aparecen las preguntas con respecto a la medicina demandada, las ratificaciones de la información y las respuesta de control. En la sección del cliente, están únicamente la clave, el precio, cantidad y subtotal de cada medicamento. Además, al final de cada medicina, aparecen dos puntos (:) e inmediatamente la confirmación de la venta, la cual se especifica por medio de un asterisco (*) en caso afirmativo o por cualquier otro signo en caso negativo. La factura se cierra con los totales de la misma. El sistema pregunta al usuario se desea otra factura.

17.- Página 684. Cuando se solicita una medicina de la cual ya no se tienen existencias, entonces se despliegan las medicinas similares para que el cliente seleccione la que juzgue conveniente. Se sigue el mismo procedimiento. Al final aparecen los totales de la factura y el cambio después del pago.

18.- Página 685. El proceso de facturación también incluye a aquellos productos cuyos datos no están contenidos en los archivos del sistema. Estos artículos se conocen como adicionales y en la factura aparecen identificados como "EX". Para entrar a esta sección de la factura, simplemente conteste con una letra A, cuando el sistema le pregunte: A) DIC o T) OT ? En este caso usted debe anotar los datos del producto, como son: Precio, Unidades e Impuesto, todo en el mismo renglón y separados por al menos un espacio. Al final, los totales incluyen también a los productos adicionales.

19.- Página 686. Se selecciona la función <D> del menú principal de funciones. Después se coloca al inicio de una nueva hoja.

20.- Página 687. Ejemplo de un listado del reporte del Corte de Caja de las Ventas del día.

21.- Página 688. Para entrar a la función <E> para hacer correcciones al archivo de trabajo, es necesario anotar correctamente las claves de seguridad. Si las dos claves son las adecuadas, el sistema le permite hacer las correcciones que juzgue convenientes.

22.- Página 689. Cuando ha sido aceptado en la función <E>, el sistema le hace la primera pregunta relacionada con el número del registro que desea modificar. Observe que si existe el registro, inmediatamente despliega el menú de correcciones, en el cual aparece la identificación de los campos factibles de ser alterados. A continuación le pide el campo que desea modificar. Únicamente acepta números como respuesta a estas dos preguntas. Para cada modificación, primero le presenta el "dato viejo" y a continuación le pide el "dato nuevo". Es el momento para hacer la corrección. Posteriormente, le pide la ratificación del cambio. Este es un ejemplo sin errores.

23.- Página 690. Aquí, primero se pretende modificar un registro que no existe. Se marca el error y especifica. Para continuar le solicita su aceptación. En el segundo caso, se seleccionó un registro existente, pero el error consiste en pretender modificar un campo no válido.

24.- Página 691. Ejemplo con errores. En este caso, se pretende trabajar con el registro 3, el cual si existe, pero se quiere modificar el campo 34, el cual no es reconocido por el programa. Posteriormente se pretenden hacer cambios en el campo 5, el cual es exclusivo del registro 0. Se marca y especifica el error.

25.- Página 692. Para este caso, se está trabajando con el registro cero; los errores consisten en querer modificar los campos 1 y 4, los cuales son exclusivos de los registros diferentes de cero.

26.- Págsina 693. Este es un ejemplo del uso de la función <E>, cuando la persona que pretende hacer correcciones al archivo de trabajo, no está autorizada y esto se demuestra, porque las claves que dió no son reconocidas por el sistema.

27.- Págsina 694. Cuando pretende hacer uso de la función <F>, para regresar información del archivo de trabajo al archivo maestro y hacer transferencias de información con los demás archivos del sistema, se selecciona esta función del menú. Primero aparece la fecha del archivo de trabajo; el sistema lo valida; si no ha sido usado anteriormente con la misma función, continua. Así, le solicita la fecha del día en que está realizando esta actividad. La valida y pide su ratificación. Ahora inicia la transferencia. Para que Usted tenga una referencia de lo que está haciendo el sistema, aparece el número del registro con el que está trabajando.

28.- Págsina 695. En este ejemplo se muestra un error al pretender entrar a la función <F> con un archivo de trabajo que anteriormente ya fue usado para tal fin. Simplemente no está permitido, actualizar dos veces con el mismo archivo de trabajo.

PROY1

! ! ! BIENVENIDO A UNA NUEVA SESION DE TRABAJO ! ! !

EL SUBSISTEMA 'CONTROL DE VENTAS', REQUIERE QUE
EXISTAN EN EL DISCO B LOS SIGUIENTES ARCHIVOS:

- 1.- ARCHIVO MAESTRO
- 2.- ARCHIVO DE TRABAJO
- 3.- ARCHIVO DE NOMBRES Y ESTADISTICAS

LEYENDO DONDE NO HAY DATOS
LEYENDO DONDE NO HAY DATOS

EL SISTEMA TIENE 0 MEDICAMENTOS

*** ERROR ***

EL ARCHIVO MAESTRO NO EXISTE O ESTA VACIO

*** ERROR ***

EL ARCHIVO DE TRABAJO NO EXISTE O ESTA VACIO

*** ERROR ***

EL ARCHIVO DE NOM. Y EST. NO EXISTE O ESTA VACIO

PROY1

! ! ! BIENVENIDO A UNA NUEVA SESION DE TRABAJO ! ! !

EL SUBSISTEMA "CONTROL DE VENTAS", REQUIERE QUE
EXISTAN EN EL DISCO B LOS SIGUIENTES ARCHIVOS:

- 1.- ARCHIVO MAESTRO
- 2.- ARCHIVO DE TRABAJO
- 3.- ARCHIVO DE NOMBRES Y ESTADISTICAS

LEYENDO DONDE NO HAY DATOS
LEYENDO DONDE NO HAY DATOS

EL SISTEMA TIENE 8 MEDICAMENTOS

* * * E R R O R * * *

EL ARCHIVO DE TRABAJO NO EXISTE O ESTA VACIO

* * * E R R O R * * *

EL ARCHIVO DE NOM. Y EST. NO EXISTE O ESTA VACIO

A.

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : B

*** ERROR ***

EL ARCHIVO DE TRABAJO NO CONTIENE INFORMACION.
FAVOR DE LISTAR EL MENU PRINCIPAL DE FUNCIONES
Y ELIJA LA FUNCION <A> PARA LLENARLO

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O :

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : C

*** ERROR ***

EL ARCHIVO DE TRABAJO NO CONTIENE INFORMACION.
FAVOR DE LISTAR EL MENU PRINCIPAL DE FUNCIONES
Y ELIJA LA FUNCION <A> PARA LLENARLO

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O : 8

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : A

EL SISTEMA INICIA EL PROCESO PARA OBTENER EL ARCHIVO DE TRABAJO A PARTIR DEL ARCHIVO MAESTRO

SE ESTA DUPLICANDO EL ARCHIVO.

ESPERE LA SENAL PARA PODER CONTINUAR ...

ANOTE LA FECHA DE HOY :

A N O => 86

M E S => 1

D I A => 30

ANOTO LA FECHA CORRECTAMENTE ANO: 86, MES: 1, DIA: 30

REALMENTE ESTA CORRECTA LA FECHA ? ANOTE S)I O N)O : S

EL SISTEMA TERMINO DE DUPLICAR, PUEDE CONTINUAR

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O :

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : A

*** E R R O R ***

EL ARCHIVO DE TRABAJO YA CONTIENE INFORMACION.

POR LO TANTO NO PUEDE USAR ESTA FUNCION.

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O :

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : P

*** ERROR ***

UNICAMENTE SE ACEPTA A, B, C, D, E, F o Q

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O :

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : S

*** ERROR ***

UNICAMENTE SE ACEPTA A, B, C, D, E, F o Q

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE SI O NO !

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : B

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :

ANO : 86, MES : 1, DIA : 30

COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA

PARA CONTINUAR PRESIONE <RETURN>

FARMACIA DE DIOS, S. A.

ARCHIVO DE TRABAJO

FECHA : 86 / 1 / 30

```

=====
CLAVE: P R E C I O : I M P : U N I D A D E S
      : D E V E N T A : % : E N : V E N D I D A S : P E D I D A S : V E N D I D A S
      : S I N I M P C O N I M P : : E X I S T E N C I A : ( A C T ) ( U L T R E M ) : D E L P E D .
=====

```

? # 1

1	1500.00	1590.00	6.	2	0	0	0
---	---------	---------	----	---	---	---	---

? # 2

2	1000.00	1060.00	6.	5	0	0	0
---	---------	---------	----	---	---	---	---

? # 3

3	0.00	0.00	0.	0	0	0	0
---	------	------	----	---	---	---	---

? # 4

4	560.00	593.60	6.	7	0	0	0
---	--------	--------	----	---	---	---	---

? # 5

5	1000.00	1060.00	6.	2	0	20	0
---	---------	---------	----	---	---	----	---

? # 6

6	0.00	0.00	0.	0	0	0	0
---	------	------	----	---	---	---	---

? # 7

7	450.00	477.00	6.	20	0	25	0
---	--------	--------	----	----	---	----	---

? # 8

8	100.00	106.00	6.	15	0	0	0
---	--------	--------	----	----	---	---	---

? # 5000

NO EXISTE, EL REGISTRO 5000 ESTA FUERA DEL RANGO (8)

Q

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE SI O NO :

FARMACIA DE DIOS, S. A.

ARCHIVO DE TRABAJO

FECHA : 86 / 1 / 30

```

=====
CLAVE: P R E C I O : I M P : U N I D A D E S
      : D E V E N T A : % : E N : V E N D I D A S : P E D I D A S : V E N D I D A S
      : S I N I M P C O N I M P : : E X I S T E N C I A : ( A C T ) : ( U L T R E M ) : D E L P E D .
=====

```

```

? # 5
  5  1000.00  1060.00  6.      2      0      20      0
? # 1
  1  1500.00  1590.00  6.      2      0      0      0
? # 8
  8   100.00   106.00  6.     15      0      0      0
? # 7
  7   450.00   477.00  6.     20      0     25      0
? # 6
  6    0.00    0.00  0.      0      0      0      0
? # 5
  5  1000.00  1060.00  6.      2      0     20      0
? # 4
  4   560.00   593.60  6.      7      0      0      0
? # 3
  3    0.00    0.00  0.      0      0      0      0
? # 2
  2  1000.00  1060.00  6.      5      0      0      0
? # 1
  1  1500.00  1590.00  6.      2      0      0      0
a

```

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O :

FARMACIA DE DIOS, S. A.

ARCHIVO DE TRABAJO

FECHA : 86 / 1 / 30

```

=====
CLAVE: P R E C I O : I M P : U N I D A D E S
      : D E V E N T A : % : E N : V E N D I D A S : P E D I D A S : V E N D I D A S
      : S I N I M P C O N I M P : : E X I S T E N C I A : ( A C T ) : ( U L T R E M ) : D E L P E D .
=====

```

7 # B

2

2	1000.00	1060.00	6.	5	0	0	0
---	---------	---------	----	---	---	---	---

7 # CASA

5

5	1000.00	1060.00	6.	2	0	20	0
---	---------	---------	----	---	---	----	---

BT # 6

6	0.00	0.00	0.	0	0	0	0
---	------	------	----	---	---	---	---

7 # NUMERO

3

3	0.00	0.00	0.	0	0	0	0
---	------	------	----	---	---	---	---

7 # (

7

7	450.00	477.00	6.	20	0	25	0
---	--------	--------	----	----	---	----	---

7 # 0

NO EXISTE, EL REGISTRO 0 ESTA FUERA DEL RANGO (8)

7 # 0

NO EXISTE, EL REGISTRO 0 ESTA FUERA DEL RANGO (8)

7 # 4500

NO EXISTE, EL REGISTRO 4500 ESTA FUERA DEL RANGO (8)

7 # 4

4	560.00	593.60	6.	7	0	0	0
---	--------	--------	----	---	---	---	---

0

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O :

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : C

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :

ANO : 86, MES : 1, DIA : 30

LE SIRVE ? S)I O N)O : S

DESEA CONTINUAR ? S)I O N)O : N

ELIMINE ESTE ARCHIVO Y OBTENGA EL ADECUADO INMEDIATAMENTE

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O : B

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : C

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :

AÑO : 86, MES : 1, DIA : 30

LE SIRVE ? S)I O N)O : N

DESEA CONTINUAR ? S)I O N)O : N

ELIMINE ESTE ARCHIVO Y OBTENGA EL ADECUADO INMEDIATAMENTE

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O : S

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : C

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :

ANO : 86, MES : 1, DIA : 30

LE SIRVE ? S)I o N)O : S

DESEA CONTINUAR ? S)I o N)O : S
 COLOQUE EL PAPEL AL INICIO DE UNA NUEVA HOJA

PRESIONE < RETURN >

# FAC:	1	FACTURA NUM.	1	86 / 1 / 30
		CLAVE P R E C I O	CANT.	SUBTOTAL
CLA CAN :1 2				
B) IEN o M) AL ? B				
3180.00	1	1590.00	2	3180.00
3180.00 T) OT o RET?				:*
CLA CAN :2 2				
B) IEN o M) AL ? B				
5300.00	2	1060.00	2	2120.00
5300.00 T) OT o RET?				:*
CLA CAN :4 2				
B) IEN o M) AL ? B				
6487.20	4	593.60	2	1187.20
6487.20 T) OT o RET?				:*
CLA CAN :5 2				
B) IEN o M) AL ? B				
8607.20	5	1060.00	2	2120.00
8607.20 T) OT o RET? T				:*
T) OT o A) DIC ? T				
8607.20		T O T A L \$		8607.20
DINERO 9000				
B) IEN o M) AL ? B				
9000.00		D I N E R O \$		9000.00
8607.20		T O T A L \$		8607.20
392.80		C A M B I O \$		392.80

GRACIAS POR SU COMPRA

DESEA OTRA FACTURA? S

# FACI	3	FACTURA NUM.	3	86 / 1 / 30
		CLAVE P R E C I O	CANT.	SUBTOTAL
CLA CAN :1 2				
B)IEN o M)AL ? B				
1: 2: 3:				
2 6 7				
QUIERE ALGUNA S)I o N)O: N				
B)IEN o M)AL? B				
0.00 T)OT o RET?				
CLA CAN :2 1				
B)IEN o M)AL ? B				
1060.00	2	1060.00	1	1060.00 :*
1060.00 T)OT o RET?				
CLA CAN :5 1				
B)IEN o M)AL ? B				
2120.00	5	1060.00	1	1060.00 :*
2120.00 T)OT o RET?				
CLA CAN :7 1				
B)IEN o M)AL ? B				
2597.00	7	477.00	1	477.00 :*
2597.00 T)OT o RET?				
CLA CAN :8 1				
B)IEN o M)AL ? B				
2703.00	8	106.00	1	106.00 :*
2703.00 T)OT o RET? T				
T)OT o A)DIC ? T				
2703.00		T O T A L	\$	2703.00
DINERO 3000				
B)IEN o M)AL? B				
3000.00		D I N E R O	\$	3000.00
2703.00		T O T A L	\$	2703.00
297.00		C A M B I O	\$	297.00

GRACIAS POR SU COMPRA

DESEA OTRA FACTURA? 8

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : D

COLOQUE EL PAPEL AL INICIO DE UN NUEVA HOJA
PRESIONE <RETURN>

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : D

COLOQUE EL PAPEL AL INICIO DE UN NUEVA HOJA
PRESIONE <RETURN>

FARMACIA DE DIOS, S. A.

86 / 1 / 30

M E D I C I N A	UNIDADES	PRECIO	SUBTOTAL	I V A	T O T A L	PRECIO	PREC.COMP	GANANCIA
CLAVE	N O M B R E	DE	\$	\$	\$	DE	X UNID	N E T A
		VENTA				COMPRA	VENDIDAS	\$
1	NEOMELUBRINA	2	1500.00	3000.00	180.00	3180.00	450.00	2100.00
2	TETRACILINA	5	1000.00	5000.00	300.00	5300.00	400.00	3000.00
4	VITACILINA	7	560.00	3920.00	235.20	4155.20	300.00	1820.00
5	SULFATIACINA	6	1000.00	6000.00	360.00	6360.00	670.00	1980.00
7	ASPIRINA	2	450.00	900.00	54.00	954.00	200.00	500.00
8	MANIBE-C	7	100.00	700.00	42.00	742.00	80.00	140.00
VENTA TOTAL DEL DIA \$				19520.00	1171.20	20691.20	9980.00	9540.00
VENTA TOTAL DEL DIA DE ARTICULOS FUERA DE INVENTARIO \$						0.00		

N O T A : NO OLVIDE HACER LO SIGUIENTE:

1o. REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO, USE LA FUNCION <F> DEL MENU.
RECUERDE QUE LA FUNCION <F> SOLO PUEDE USARSE UNA VEZ CON CADA ARCHIVO DE TRABAJO, TENGA CUIDADO

2o. PASE EL ARCHIVO DE TRABAJO AL DISCO ALMACENADOR

DESEA LISTAR EL MENU PRINCIPAL DE FUNCIONES ? ANOTE S)I O N)O : N

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
G	SALIRSE

FUNCION DESEADA : E

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :
 AÑO : 86, MES : 1, DIA : 30

ESTE ES EL ARCHIVO DE TRABAJO QUE USTED BUSCA ?
 ANOTE S)I o N)O => S

ANOTE LA CLAVE PRIMARIA DE SEGURIDAD => 10

ANOTE LA CLAVE SECUNDARIA DE SEGURIDAD => 20

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => 1

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES :

```
=====
C           A           M           P           O
#   N           O           M           B           R           E
=====
```

```
1  UNIDADES EN EXISTENCIA ACTUALES
2  UNIDADES VENDIDAS ACTUALES
3  UNIDADES PEDIDAS
4  UNIDADES VENDIDAS DEL PEDIDO
5  NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)
6  VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)
=====
```

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 1

DATO VIEJO : 2 , ANOTE EL NUEVO DATO : 0

ESTA CORRECTO ? ANOTE S)I O N)O => S

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 1 ?

ANOTE S)I O N)O => S

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 2

DATO VIEJO : 5 , ANOTE EL NUEVO DATO : 10

ESTA CORRECTO ? ANOTE S)I O N)O => S

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 1 ?

ANOTE S)I O N)O => N

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN>

EN CASO CONTRARIO, ANOTE LA LETRA Q =>

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => 4500

*** ERROR ***

NO EXISTE EL REGISTRO 4500 , ESTA FUERA DEL RANGO (8)

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN>

EN CASO CONTRARIO, ANOTE LA LETRA Q =>

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => 2

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES :

```

=====
C           A           M           P           O
*  N       O           M           B           R           E
=====

```

- 1 UNIDADES EN EXISTENCIA ACTUALES
- 2 UNIDADES VENDIDAS ACTUALES
- 3 UNIDADES PEDIDAS
- 4 UNIDADES VENDIDAS DEL PEDIDO
- 5 NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)
- 6 VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 8

*** ERROR ***

NO EXISTE, EL CAMPO 8 ESTA FUERA DEL RANGO

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 2 ?
ANOTE S)I O N)O => N

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN>

EN CASO CONTRARIO, ANOTE LA LETRA Q =>

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => 3
 LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES !

```
=====
C           A           M           P           O
#   N       O           H           B           R           E
=====
1  UNIDADES EN EXISTENCIA ACTUALES
2  UNIDADES VENDIDAS ACTUALES
3  UNIDADES PEDIDAS
4  UNIDADES VENDIDAS DEL PEDIDO
5  NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)
6  VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)
=====
```

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 34

*** E R R O R ***

NO EXISTE, EL CAMPO 34 ESTA FUERA DEL RANGO

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 3 ?
 ANOTE S)I O N)O => S

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 5

UNICAMENTE PUEDE CORREGIR LOS CAMPOS DEL 1 AL 4

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 3 ?
 ANOTE S)I O N)O => N

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN>

EN CASO CONTRARIO, ANOTE LA LETRA Q =>

CUAL REGISTRO QUIERE MODIFICAR ? ANOTE EL NUMERO => 0

LOS CAMPOS QUE PUEDE MODIFICAR SON LOS SIGUIENTES :

```
=====
C          A          M          P          O
#  N      O      M      B      R      E
=====
1  UNIDADES EN EXISTENCIA ACTUALES
2  UNIDADES VENDIDAS ACTUALES
3  UNIDADES PEDIDAS
4  UNIDADES VENDIDAS DEL PEDIDO
5  NUMERO DE LA ULTIMA FACTURA (REGISTRO CERO)
6  VENTA TOTAL FUERA DE INVENTARIO (REGISTRO CERO)
=====
```

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 1

UNICAMENTE PUEDE CORREGIR LOS CAMPOS 5 Y 6

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 0 ?
ANOTE S)I 0 N)O => 5

CUAL CAMPO DESEA ACTUALIZAR ? ANOTE EL NUMERO => 4

UNICAMENTE PUEDE CORREGIR LOS CAMPOS 5 Y 6

DESEA CAMBIAR OTRO CAMPO DEL REGISTRO 0 ?
ANOTE S)I 0 N)O => N

SI DESEA SEGUIR ACTUALIZANDO, TRANSMITA <RETURN>

EN CASO CONTRARIO, ANOTE LA LETRA 0 => 0

FARMACIA DE DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
Q	SALIRSE

FUNCION DESEADA : E

ESTA USTED USANDO UN ARCHIVO DE TRABAJO DE FECHA :
 ANO : 86, MES : 1, DIA : 27

ESTE ES EL ARCHIVO DE TRABAJO QUE USTED BUSCA ?
 ANOTE S)I o N)O => S

ANOTE LA CLAVE PRIMARIA DE SEGURIDAD => 34

ANOTE LA CLAVE SECUNDARIA DE SEGURIDAD => 57

USTED NO ESTA AUTORIZADO PARA HACER ACTUALIZACIONES

FUNCION DESEADA : F

EL ARCHIVO DE TRABAJO QUE ACTUALIZARA
AL ARCHIVO MAESTRO ES DE FECHA :

DIA : 30 MES : 1 AÑO : 86

EL ARCHIVO DE TRABAJO ES EL CORRECTO.

SE ESTA ACTUALIZANDO EL ARCHIVO MAESTRO.
ESPERE LA SENAL PARA PODER CONTINUAR ...

EL SISTEMA TERMINO DE ACTUALIZAR AL MAESTRO

ANOTE LA FECHA DE HOY :

A N O => 86

M E S => 1

D I A => 31

ANOTO LA FECHA CORRECTAMENTE AÑO: 86, MES: 1, DIA: 31

REALMENTE ESTA CORRECTA LA FECHA ? ANOTE S/I o N/D => S

EL SISTEMA ESTA TRANSFIRIENDO INFORMACION

ESPERE LA SENAL PARA PODER CONTINUAR

1
2
3
4
5
6

EL SISTEMA YA TERMINO.

PUEDA USTED CONTINUAR

A.

FARMACIA DEL DIOS, S. A.

FUNCION	DESCRIPCION DE LA FUNCION
A	DUPLICAR EL ARCHIVO MAESTRO PARA OBTENER EL ARCHIVO DE TRABAJO.
B	LEER EL ARCHIVO DE TRABAJO DEL DIA.
C	FACTURACION PARA CLIENTES.
D	CORTE DE CAJA DE LAS VENTAS DEL DIA.
E	CORRECCIONES AL ARCHIVO DE TRABAJO.
F	REGRESAR INFORMACION DEL ARCHIVO DE TRABAJO AL ARCHIVO MAESTRO. INCLUYE TRANSFERENCIA DE INFORMACION DEL ARCHIVO MAESTRO AL ARCHIVO DE NOMBRES Y ESTADISTICAS
G	SALIRSE

FUNCION DESEADA : F

EL ARCHIVO DE TRABAJO QUE ACTUALIZARA
AL ARCHIVO MAESTRO ES DE FECHA :

DIA : 30 MES : 1 AÑO : 86

ESTE ARCHIVO DE TRABAJO YA FUE USADO UNA VEZ.

NO ESTA PERMITIDO ACTUALIZAR DOS VECES
CON EL MISMO ARCHIVO DE TRABAJO

VII. C O N C L U S I O N E S

VII.1 CONCLUSIONES: SISTEMA Y METODOLOGIA.

Actualmente el mundo del procesamiento de datos por computadora, se ha extendido a prácticamente todas las actividades del hombre.

Por eso, es necesario buscar procedimientos para manejar de una manera ordenada, toda la información que generen sus elementos de desarrollo y producción, para poder tomar las mejores decisiones en el momento adecuado.

Nuestro trabajo, tiene dos aspectos fundamentales a saber: el desarrollo de un sistema de información, llamado "CONTROL DE VENTAS E INVENTARIOS DE UNA FARMACIA POR MEDIO DE UNA MICROCOMPUTADORA" y la presentación de una metodología para la realización de los sistemas de información.

Actualmente, es difícil encontrar en el mercado nacional, un producto como el que estamos presentando, el cual ha sido aprobado por varios administradores de farmacias ubicadas en el área metropolitana.

Hemos expuesto un método, que no resulta nuevo, al cual lo presentamos de manera clara y por escrito para que sirva como modelo o guía en el desarrollo de nuevos sistemas de información.

Nuestro trabajo, está orientado hacia la implantación de un método que ayude de manera eficaz al proceso de elaboración de sistemas de datos.

No pretendemos establecer reglas rígidas e inmutables, pero si deseamos dejar en claro que, mientras no se siga un método perfectamente establecido, los sistemas seguirán teniendo los mismos defectos y limitaciones.

Es necesario aclarar que la mayor parte del presupuesto para el desarrollo de un sistema, se invierte en el aspecto de la programación y mantenimiento, por lo que pretendemos normalizar y estandarizar estos procesos, con el fin de que todos los que intervienen en su desarrollo hablen el mismo idioma.

Nos hemos encontrado con el siguiente problema. Existe mucha bibliografía, pero muy pocos autores tocan el aspecto del desarrollo "físico" de un sistema, donde ponsan en práctica todo aquello que argumentan en sus textos. Por otra parte, es muy importante, mostrar de manera clara, sencilla y precisa, lo que se pretende lograr.

Nuestro trabajo abarca eso y más.

VII.2 VENTAJAS Y DESVENTAJAS.

Las ventajas y desventajas son analizadas desde dos puntos de vista:

1. El sistema "CONTROL DE VENTAS E INVENTARIOS DE UNA FARMACIA POR MEDIO DE UNA MICROCOMPUTADORA".

2.- La metodología para el desarrollo del sistema.

Tomando en cuenta todos los procesos que realiza el Sistema "CONTROL DE VENTAS E INVENTARIOS DE UNA FARMACIA POR MEDIO DE UNA MICROCOMPUTADORA", se observan las siguientes ventajas:

1.- Se satisfacen plenamente las necesidades del usuario.

2.- Se puede usar en cualquier farmacia, ya que su diseño se amolda perfectamente al tipo de manejo de información que se tiene actualmente, buscando su estandarización.

3.- Proporciona enormes ahorros en dinero, ya que, como quedó asentado, todo lo que entra y sale de la farmacia, es tomado en cuenta por el sistema. Por otra parte específicamente en los aspectos de caducidad, pedidos, inventarios, facturación, etc. la información siempre está al día, por lo que, cuando es necesario tomar decisiones urgentes, se tienen a la mano todos los datos.

4.- Los reportes que genera el sistema son de tipo universal, ya que satisfacen las necesidades de información de prácticamente todas las farmacias del país.

5.- Rapidez. Como las actualizaciones de archivos se hace en el preciso instante en que se presenta un movimiento, ya sea de ventas, altas, bajas, etc. siempre se tiene la información actualizada.

6. Seguridad. Es uno de los puntos más importantes del sistema, ya que la confidencialidad de los datos es un aspecto fundamental que se ha cuidado hasta el más mínimo detalle.

7.- Economía. Este sistema está orientado hacia farmacias medianas y grandes, las cuales pueden pagar este servicio, obteniendo el mayor beneficio posible.

8. Adaptabilidad. El sistema está diseñado de tal manera que puede instalarse en casi todos los equipos de micros (PC` s) que existen actualmente en el mercado. La única condición es que el equipo use el lenguaje PASCAL.

9.- Crecimiento. Si por algún motivo, el usuario presenta alguna nueva necesidad, no es problema, ya que el sistema acepta incrementos en sus funciones, sin hacer cambios estructurales en el mismo.

10.- Este sistema puede aplicarse a cualquier establecimiento comercial, aun cuando no fuese farmacia. Solo serían necesarias pequeñas modificaciones.

Desde el punto de vista de la metodología:

1.- Los pasos que se siguen para el desarrollo e implantación de un sistema, son muy claros y se pueden aplicar prácticamente a todos los procesos.

2.- Normalizar y estandarizar los procesos de análisis y desarrollo de sistemas de información.

Con respecto a las desventajas, podemos decir:

En realidad, no existen desventajas notorias que pongan en peligro el funcionamiento del sistema.

Sin embargo, es posible que exista alguna.

En el aspecto metodológico, tampoco las hay.

Pero recuerde una cosa. Proponer una metodología significa, dar una alternativa de desarrollo.

VII.3 MEJORAS.

Entre las mejoras que pueden implantarse están las siguientes:

En cuanto al sistema.

1.- Adaptar el número de proveedores a las condiciones del mercado.

2.- Validar la duplicidad de un registro tomando como base el nombre del medicamento.

La metodología puede ser enriquecida, con otros aspectos que fueron ignorados o tocados con aparente simplicidad.

VII.4 SUGERENCIAS PARA TRABAJOS FUTUROS.

Debido a los avances de la tecnología, presentamos las siguientes sugerencias:

1.- Establecer una red de computadoras, para que se conecten a ella, todos los que tengan algo que ver con el proceso. Por ejemplo: laboratorios, distribuidores, administradores de farmacias y clientes.

De que manera?

El cliente haría sus pedidos a una farmacia, por teléfono. El administrador atendería a la solicitud, sin necesidad de que el cliente acudiese personalmente hasta el local comercial. La entrega se haría de manera tradicional.

Por otra parte, los pedidos de las farmacias a los proveedores, se harían por la red de computadoras, sin necesidad de la intervención humana, ya que los reportes que genera actualmente, serían enviados al laboratorio o distribuidor correspondiente.

De esta manera, el administrador solo sería consultado en casos extremos, ya sea para tomar una decisión o simplemente para informarle lo que ya se hizo.

La comunicación sería entre las computadoras de la red.

VIII. B I B L I O G R A F I A .

VIII. B I B L I O G R A F I A .

AWAD, M. ELIAS.
Systems Analysis and Design
Richard D. Irwin
USA, 1979

DAVIS, GORDON BITTER.
Computers and Information Processing
McGraw Hill
Tokyo, Japan, 1977

DE MARCO, TOM
Structured Analysis and Systems Specification
Prentice Hall
New Jersey, 1979

DIGITAL RESEARCH
Pascal/MT+, User's Guide
Commercial Press
California, 1982

DIJKSTRA, E. W.
A Discipline of Programming
Prentice Hall
New Jersey, 1976

GANE, C. AND SORSON, T.
Structured Systems Analysis: Tool and Techniques
Prentice Hall
New Jersey, 1979

GEREZ G. V., MIER M. M., NIEVA G. R Y RODRIGUEZ O. G.
Desarrollo y Administración de Programas de Computadora.
Cecsa.
México, 1984.

KNUTH, D. E.
The Art of Computer Programming, Vol.1
Addison Wesley
Mass, 1968

METZGER, PHILIP W.
Managing a Programming Project
Prentice Hall
New Jersey, 1973

RANGEL GUTIERREZ R.H.
Apuntes de Programación Estructurada.
Fac. de Ingeniería, UNAM
México, 1985

SANDERS, DONALD H.
Computers Today
McGraw Hill
USA, 1983

VERSELLO, ROBERT J. AND REUTER, J.
Data Processing: System and Concepts
McGraw Hill
USA, 1982

YOURDON, EDWARD AND CONSTANTINE, LARRY L.
Structured Design
Prentice Hall
New Jersey, 1979