UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE INGENIERIA

DISEÑO DE UN MANEJADOR DE ARCHIVOS DE ACCESO DIRECTO POR MULTIPLES LLAVES

T E S I S

QUE PARA OBTENER EL TITULO DE

INGENIERO EN COMPUTACION

P R E S E N T A N :

JAVIER RIVERA CHAVEZ
HECTOR CONRADO JIMENEZ SILVA

DIRECTOR DE TESIS
ING. LUIS G. CORDERO BORBOA.

MEXICO, D.F. ABRIL 1991

UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DISEÑO DE UN MANEJADOR DE ARCHIVOS DE ACCESO DIRECTO

POR MULTIPLES LLAVES

I N D I C E

INTRODUCCION

Se entiende por informática, palabra formada por la abreviación de los términos información y automática, el conjunto de métodos y mecanismos que tienen como objetivo el tratamiento racional y automático de la información. La informática nació cuando el hombre sintió la necesidad de almacenar y ordenar los múltiples conocimientos heredados de sus antepasados para tenerlos a su alcance y utilizarlos a su debido tiempo. La computadora, máquina destinada a procesar los datos, ha llegado a liberar de los trabajos puramente mecánicos y rutinarios al ser humano que, de este modo, tiene la posibilidad de dedicarse a tareas más útiles y creativas. Las empresas, grandes y pequeñas se esfuerzan por disponer de computadoras. En la actualidad, ningún Estado, aunque carezca de medios, puede prescindir de esta nueva técnica ya que la potencia económica de un país depende en gran parte de ella.

El desarrollo informático en nuestro país en los últimos años ha tenido un creciente aumento, lo cual es originado por la gran necesidad que existe en todas las áreas de tener un alto nivel de automatización, aunado a la relativa facilidad con que se puede disponer de recursos de cómputo.

Uno de los principales usuarios de los recursos de cómputo de reciente tecnología son las instituciones bancarias las que han procurado mantenerse actualizadas con sistemas de cómputo de nueva tecnología, buscando siempre como finalidad ofrecer a sus clientes mejores servicios automatizados.

En la actualidad la rapidez con que una institución pueda reaccionar a desarrollar un nuevo sistema es muy importante, ya que los cambios constantes en el mercado de dinero hacen necesario tener una rápida respuesta de las áreas de desarrollo de sistemas.

Para todos es sabido que el desarrollo de los programas es más lento que el desarrollo de los equipos, siendo esto una preocupación para las firmas constructoras de computadoras. Es por esto que se han desarrollado lenguajes de cuarta generación tales como bases de datos, generadores de aplicaciones etc., los cuales contribuyen en gran medida a aprovechar eficientemente los recursos humanos.

Sin embargo no todos los usuarios de computadoras tienen una infraestructura de desarrollo de sistemas que incluya herramientas modernas de desarrollo que les permita lograr estos resultados. En la actualidad con tantas marcas de computadoras y capacidades de los mismos, es difícil establecer una estrategia de desarrollo informático global.

Esto representa en general una gran desventaja para algunas empresas que no han definido una línea de desarrollo, la cual los obliga a renovar sus equipos de cómputo con una frecuencia aproximada de cinco años sin riesgo de caer en la obsolescencia.

Estos cambios en muchas ocasiones representan un retroceso en cuanto automatización de sistemas se refiere, debido a que en pocas palabras es necesario obligar los viejos sistemas a los equipos nuevos perdiéndose valiosas horas de trabajo.

Esta situación obliga a la institución a exigir esfuerzos extras en su personal para salir a tiempo al mercado con un producto nuevo y en algunas ocasiones con posterioridad a la de las instituciones líderes en la banca nacional.

Una computadora es una máquina que puede resolver problemas ejecutando unas instrucciones dadas. Se llama programa a una secuencia de instrucciones que describe como ejecutar cierta tarea. Los circuitos electrónicos de cada computadora pueden reconocer y ejecutar directamente un conjunto limitado de instrucciones simples. Todos los programas que se desean ejecutar en una computadora deben convertirse previamente en una secuencia de estas instrucciones simples. Estas instrucciones básicas pocas veces rebasan la complejidad de sumar dos números, comprobar si un número es cero, mover datos de una parte de la memoria a otra, etc.

l conjunto de las instrucciones primitivas de una computadora forma un lenguaje con el cual podemos comunicarnos con ella. Dicho lenguaje se llama lenguaje máquina. Existen lenguajes que son más fáciles de usar que el lenguaje máquina, los cuales hacen uso de un compilador, a estos lenguajes se les conoce como lenguajes de alto nivel. El compilador se encarga de transformar las instrucciones en lenguaje de alto nivel a lenguaje máquina.

n sistema de datos bien diseñado proporciona a los usuarios la información necesaria para tomar decisiones mediante a recopilación eficiente de datos, su procesamiento y la comunicación de la información resultante. Un sistema de ste naturaleza puede incluir procedimientos manuales, empleo de formas comerciales, archivos de datos y, por upuesto, programas computacionales.

l procesamiento de datos administrativos tiene varias características especiales respecto a la entrada de datos, su rocesamiento y la salida de dichos sistemas. La entrada de datos normalmente es voluminosa y consiste en registros e datos numéricos y no numéricos de longitudes variables. El procesamiento de la entrada de datos comerciales se aracteriza por la actualización de los archivos. La salida se caracteriza por la producción de reportes que agrupan resumen datos por categorías significativas que corresponden a varias funciones.

uando se tiene una gran cantidad de registros para procesar lo más conveniente es usar archivos de acceso directo. l acceso directo se ha creado exclusivamente para computadoras que utilizan dispositivos de almacenamiento de isco. Los archivos de acceso directo tienen frecuente uso en directorios, tablas de artículos y precios, listas de ombres, etc. De hecho son frecuentemente utilizados en aplicaciones donde el acceso rápido a los datos es esencial iendo éstos accesados de manera simple y rápida.

omo origen de la presente tesis se planteó la posibilidad de crear una herramienta de acceso a archivos la cual mpla con algunas de las funciones propias de un manejador de bases de datos, desarrollada con algún lenguaje de ercera generación como es el COBOL.

Esta herramienta deberá proporcionar la flexibilidad de acceso a los datos que proporciona una base de datos aprovechando los recursos de la empresa. Por otra parte esta herramienta deberá cubrir las características propias de un sistema en línea.

Como una primera opción se determinó que podría desarrollarse con archivos de tipo secuencial o secuenciales indexados por múltiples llaves, sin embargo como se verá más adelante este tipo de archivos son poco eficientes con sistemas en línea ya que son de acceso serial.

Como segunda opción se pensó solucionar el problema con archivos de acceso directo, sin embargo la experiencia que se tenía en este tipo de archivos es que sólo podrían accesarse por una sóla llave principal. Por tal razón se decidió realizar un manejador de archivos de acceso directo por diferentes llaves, lo cual viene siendo un manejador de bases de datos elemental. Con esta herramienta se podrán accesar archivos por llaves primarias y alternas relacionando los datos entre sí.

La finalidad de la presente tesis es mostrar las bases de este proyecto así como el desarrollo del manejador de archivos de acceso directo por múltiples llaves.

1 DESCRIPCION DE ARCHIVOS TRADICIONALES

La mayoría de los sistemas operativos proveen un conjunto básico de organización de archivos que son muy populares entre los usuarios del sistema. Los tres tipos de organización de archivos más comunes son: el secuencial, el secuencial indexado y el directo.

1.1 Archivos Secuenciales

En un archivo secuencial, los registros son almacenados uno después de otro en el dispositivo de almacenamiento. Debido a que la asignación secuencial es conceptualmente simple, todavía es lo bastante flexible para hacer frente a muchos de los problemas asociados con la manipulación de grandes volúmenes de datos, un archivo secuencial ha sido la más popular de las estructuras de archivos básicas usadas en la industria del procesamiento de datos.

Todos los tipos de dispositivos de almacenamiento externo soportan un archivo de organización secuencial. Algunos dispositivos, por su naturaleza física, solamente pueden soportar archivos secuenciales, por ejemplo, la información es almacenada en una cinta magnética como una serie continua de registros a lo largo de la cinta. Para acceder un registro en particular se requiere accesar todos los registros previos en el archivo. Otros dispositivos que son estrictamente secuenciales por naturaleza son: lectoras de cinta de papel, lectoras de tarjetas, cintas de cassette e impresoras en línea.

los discos y tambores magnéticos proveen acceso secuencial y directo a los registros. Los registros de un archivo secuencial son almacenados físicamente en lugares adyacentes de una pista de un disco o tambor. Si el archivo es más grande que la cantidad de espacio existente en una pista, entonces los registros son almacenados en pistas adyacentes. Este concepto de adyacencia física puede ser extendido a cilindros y dispositivos de almacenamiento completo donde más de un dispositivo es tomado como una unidad de control común.

Las operaciones que pueden ser realizadas en un archivo secuencial pueden diferir significativamente, dependiendo del dispositivo de almacenamiento usado. Por ejemplo, un archivo en cinta magnética puede ser un archivo de entrada o un archivo de salida pero no ambos a la vez. Un archivo secuencial en disco puede ser usado estrictamente para entrada, estrictamente para salida o para ser actualizado. Actualizar significa que el registro leído más recientemente puede ser reescrito en el mismo archivo si se desea. Algunos sistemas operativos proveen facilidades para acceso de archivos que permiten que un archivo sea extendido para la escritura de registros después del último registro actual. También, algunas veces es posible saltar hacia atrás o hacia adelante un cierto número de registros en el archivo sin hacer lectura o escritura.

Antes de explicar el tipo de procesamiento que es normalmente aplicado a los archivos secuenciales es importante examinar como la información en un archivo es transmitida al programa de usuario y viceversa. Se sabe que a menudo es ventajoso agrupar un número de registros lógicos dentro de un simple registro físico o bloque. Los bloques completos son transferidos entre la memoria principal y el almacenamiento externo.

Cada vez que una instrucción de lectura es ejecutada, el próximo registro del archivo secuencial activo es movido dentro del área del programa y asignado a la estructura de datos. Sin embargo, cada vez que una operación de lectura o escritura es ejecutada para un dispositivo particular de almacenamiento, un bloque de registros lógicos es transferido. La diferencia entre las instrucciones de lectura y escritura de un programa con los comandos de lectura y escritura usados en un dispositivo particular se resuelve usando un área de almacenamiento temporal entre el almacenamiento externo y el área de datos de un programa. Esta área de almacenamiento temporal es una sección de memoria principal que es igual al tamaño máximo de un bloque de registros lógicos usados por un programa. Las rutinas del manejador de datos del sistema operativo usan dicha área para el bloqueo y desbloqueo de registros.

Para ilustrar como el bloqueo y desbloqueo de registros es realizado usando un área de almacenamiento temporal, consideremos el uso de un archivo secuencial como archivo de entrada. Cuando la primera instrucción de lectura es ejecutada, un bloque de registros es movido desde el almacenamiento externo al almacenamiento temporal. El primer registro en el bloque es después transferido al área de datos del programa. Por cada ejecución subsecuente de

instrucciones de lectura, el próximo registro sucesivo en el almacenamiento temporal es transferido al área de datos. Solamente después de que cada registro en el almacenamiento externo ha sido movido al área de datos, en respuesta a instrucciones de lectura, la próxima instrucción de lectura provoca que otro bloque sea transferido al almacenamiento temporal desde el dispositivo externo. Los nuevos registros en el almacenamiento temporal son movidos al área de datos, como se describió previamente, y este proceso es repetido completamente para cada bloque que se haya leído.

De modo similar, las instrucciones de escritura provocan la transferencia de los datos del programa al almacenamiento temporal. Cuando el almacenamiento temporal llega a estar completo, entonces el bloque es escrito en el dispositivo de almacenamiento externo inmediatamente después del bloque precedente de registros.

La técnica de almacenamiento temporal descrita es comúnmente llamada de almacenamiento temporal simple. La de almacenamiento temporal múltiple hace uso de una cola de áreas de almacenamiento temporal que son normalmente controladas por el sistema operativo. La necesidad de más de un almacenamiento temporal surge debido al retardo (que es del orden de milisegundos) inherente en leer o escribir el próximo bloque de registros. Este retardo en la ejecución de un programa solamente ocurre a menudo cada n ejecuciones de las instrucciones de lectura o de escritura, cuando un factor de bloqueo de n y un solo almacenamiento temporal es usado. Sin embargo, si el programa se está ejecutando en un ambiente donde el tiempo de respuesta deseado es pequeño y donde el procesador y la actividad de entrada/salida son sobrepuestas, entonces es deseable eliminar este retardo usando múltiples almacenamientos temporales.

En algunos sistemas los bloques de registros lógicos que constituyen un archivo secuencial pueden ser de longitud fija o de longitud variable. Un bloque de longitud variable contiene un número variable de registros. Por lo tanto, no es conocido cuántos registros caben en un bloque y una longitud máxima es definida para el bloque. Esta longitud máxima es usada para estimar el tamaño del almacenamiento temporal requerido para contener el bloque, y tantos registros como es posible son agrupados dentro del bloque debido a las facilidades del manejador de datos.

la longitud máxima de un bloque depende del dispositivo de almacenamiento usado por el archivo. La longitud mínima a la longitud depende del espacio mínimo usado para el almacenamiento temporal en memoria principal con almacenamiento en disco, los bloques son generalmente limitados en tamaño a la capacidad de una pista cuando un dispositivo de sector direccionable, un bloque corresponde al número máximo de sectores

Ya que se ha explicado la vista física de un archivo secuencial, cómo los registros son transferidos a/de el área del programa de a el archivo se van a mencionar los tipos de procesamiento que para la mayoría de los archivos secuenciales son concurrentes. El procesamiento serial es el acceso de registros uno después de otro, de acuerdo al orden físico en que aparecen en el archivo. Es un asunto fácil procesar un archivo secuencial serialmente. El procesamiento secuencial es el acceso de registros uno después de otro en orden ascendente o descendente de acuerdo a la llave del registro. Una vez que el archivo ha sido ordenado por su llave el procesamiento secuencial es equivalente al procesamiento serial. La mayoría de los archivos secuenciales son ordenados por su llave cuando el archivo es creado. La llave puede ser el campo que es más a menudo buscado durante el procesamiento de el archivo.

En raras ocasiones, el procesamiento serial es totalmente requerido en un archivo, independientemente de la llave bajo la cual el archivo está ordenado.

El procesamiento secuencial y serial son más efectivos cuando un alto porcentaje de los registros en el archivo deben ser procesados, ya que cada registro en el archivo debe ser examinado. Un número de transacciones relativamente grande podría ser procesado en lote. Si los registros serán agregados al archivo, es necesario crear un nuevo archivo a menos que los registros sean agregados al final del archivo. En muchos sistemas no existen las facilidades que permiten la extensión directa de un archivo secuencial. Los registros pueden ser borrados de un archivo secuencial rutinariamente como borrados durante la actualización del archivo. Sin embargo, este procedimiento dirige a los archivos con registros ficticios o un almacenamiento ineficiente y el tiempo de procesamiento es incrementado. Usualmente, los registros borrados son físicamente eliminados creando un nuevo archivo. Cuando la creación de un nuevo archivo es necesaria, puede ser hecha tan frecuentemente como sea requerida.

Los puntos importantes concernientes al proceso secuencial de archivos secuenciales son los siguientes

- El procesamiento secuencial es más sencillo. Por ejemplo, grupos de transacciones pueden ser hechas por un solo procesamiento en lote.

- Un nuevo archivo puede ser creado si hay adiciones o si se requiere un número significativo de eliminaciones.

- El tiempo de respuesta rápido no puede ser esperado para una transacción o un lote de transacciones.

- El requerimiento de que los registros de un archivo secuencial sea ordenado por una llave particular no es fundamental si el archivo está siendo examinado para realizar la misma operación sobre todos los registros.

1.2 Archivos secuenciales indexados

Un aspecto importante que afecta la estructura del archivo es el tipo de medio físico en que reside el archivo. La capacidad de accesar directamente un registro basado en una llave puede solamente ser llevada a cabo si el dispositivo externo de almacenamiento usado soporta este tipo de acceso. Dispositivos tales como lectoras de tarjetas y unidades de cinta impiden el acceso de un registro particular solamente después de haber leído todos los otros registros que físicamente ocurren antes del registro deseado en el archivo. Por lo tanto el acceso directo de registros es imposible para estos tipos de dispositivos. Los dispositivos de almacenamiento existen que soportan el acceso directo y secuencial son las unidades de disco y tambor magnético.

Los conceptos de la estructura del archivo secuencial indexado son más ejemplificados cuando se considera un disco magnético como el medio de almacenamiento. Debido a su baja relación precio/rendimiento , a su gran capacidad, los discos son generalmente escogidos cuando se usan archivos secuenciales indexados.

Se presentarán dos tipos de organización de archivo secuencial indexado. La primer organización de archivo secuencial indexado consiste de tres áreas separadas, el área primaria, el área indexada y el área de sobreflujo. El área primaria es una área en la cual los registros de datos son escritos cuando el archivo es creado. El archivo es creado secuencialmente por la escritura de registros en el área primaria en la secuencia dictada por el orden léxico de las llaves de los registros. El proceso de escritura se inicia en la segunda pista de un cilindro particular, por decir algo, en el cilindro n de un disco. Cuando este cilindro es llenado, la escritura continua en la segunda pista

e el primer cilindro (self) y continua en esta fase hasta que la creación del archivo es completada. Si el archivo reado es accesado secuencialmente de acuerdo a la llave, los registros son procesados en el orden en que fueron escritos.

a segunda área importante de un archivo secuencial indexado es el área de indexación, es creada automáticamente por as rutinas del manejador de datos del sistema operativo. Un número de niveles de indexación pueden ser involucrados n un archivo secuencial indexado. El nivel más bajo de indexación es el índice de pista, que está siempre escrito n la primer pista de los cilindros en que se encuentra el archivo secuencial indexado. El índice de pista contiene os entradas por cada pista primaria de el cilindro: una entrada normal y una entrada de sobreflujo. La entrada ormal está compuesta de las direcciones de la pista primaria para las cuales la entrada está asociada y de el más lto valor de las llaves de los registros clasificados en esa pista. Si no hay sobreflujo de registros, la entrada e sobreflujo es igual a la entrada normal. De la misma forma como un índice de pista describe el almacenamiento de egistros en las pistas de un cilindro, el índice de cilindro indica como los registros son distribuidos sobre un úmero de cilindros.

n nivel final de indexación existe en está estructura jerárquica indexada. Un índice maestro es usado para un rchivo extremadamente grande en donde una búsqueda del índice de cilindro es también tiempo consumido. Este índice orma el nodo raíz del árbol de índices usado en un archivo secuencial indexado.

e localización de un registro con una llave dada involucra una búsqueda en el índice maestro para encontrar el ndice de cilindro apropiado con que el registro está asociado. Posteriormente, se hace una búsqueda en el índice el cilindro para encontrar el índice de pista. Finalmente, una búsqueda de pista es requerida para localizar el egistro deseado. Se debe hacer notar que un índice maestro no siempre es necesario y solamente debe ser requerido ara archivos grandes. Cuando es usado debe residir en memoria principal durante todo el proceso de un archivo ecuencial indexado.

i se agregan registros en un archivo secuencial, un nuevo archivo secuencial debe ser creado. Podemos usar el mismo rincipio para el manejo de adiciones en un archivo secuencial indexado.

Sin embargo, porque es posible acceder registros directamente en un archivo secuencial indexado, este tipo de archivo es usado para tener acceso en un ambiente más variado, y de requerir rapidez. Por ejemplo, un ambiente en que surgen muchas adiciones y eliminaciones es durante transacciones en línea o pequeños lotes de transacciones. Tales eliminaciones y adiciones deben ser inmediatamente reflejadas en el archivo.

Los problemas de agregar registros son resueltos por la creación de una área o áreas de sobreflujo, usualmente en el mismo dispositivo en el que reside el archivo. Dos tipos de áreas de sobreflujo son posibles: una área de sobreflujo de cilindro y una área de sobreflujo independiente. Una área de sobreflujo de cilindro es un número de pistas dedicadas en un cilindro que contienen un número de pistas de áreas primarias. Si, a través de la adición de un registro, un sobreflujo es creado en las pistas del área primaria del cilindro, entonces los registros de sobreflujo son almacenados en el área de sobreflujo del cilindro.

Cuando más registros son agregados al archivo secuencial indexado, se llena el área de sobreflujo. Cuando esto sucede, los futuros registros de sobreflujo son transferidos al área de sobreflujo independiente. El área de sobreflujo independiente reside en un cilindro o en cilindros independientes de cualquier área primaria del cilindro. Estos registros de sobreflujo son ligados de la misma manera en que están en el área de sobreflujo del cilindro. Se nota, sin embargo, que para discos con cabezas móviles, el uso de áreas de sobreflujo deben ser desechadas debido al número significativo de búsquedas que son generadas cuando el brazo de acceso es movido entre las áreas primaria y de sobreflujo independiente.

Hasta ahora hemos explicado la adición de registros. Ahora a considerar la eliminación de registros. En esta organización de archivo secuencial indexado, los registros no son borrados físicamente de el archivo, pero son marcados como borrados (usando "11111111" en el primer byte del registro). Si un nuevo registro es agregado más tarde y con la misma clave que el registro previamente borrado, entonces el espacio ocupado por el registro borrado es recuperado.

los registros que son colocados en el área de sobreflujo nunca son regresados al área primaria para ser borrados. Solamente por la reorganización del archivo puede un registro de sobreflujo ser colocado en el área primaria. La reorganización es llevada a cabo secuencialmente copiando los registros del archivo en un archivo temporal y después rastreando el archivo mediante una copia secuencial de los registros de respaldo en el archivo original. Debido a que la recuperación de registros de sobreflujo puede ocasionar un sobrecalculo, la cantidad de desorganización de un archivo secuencial indexado debe ser monitoreada continuamente.

El segundo archivo secuencial indexado está organizado en bloques de datos y en bloques de índices. Ambos bloques son manejados como registros lógicos que son asignados y transferidos a y de memoria principal bajo la dirección de un monitor. El usuario no tiene control sobre el lugar físico de los bloques en el dispositivo de almacenamiento externo. Este servicio sistema de control es un requerimiento necesario debido a que el monitor permite compartir los archivos del disco en un ambiente multiusuario. El usuario tiene control sobre el tamaño de los datos y sobre los bloques de índices.

Un bloque de datos está compuesto de registros, llaves con apuntadores a los registros de datos dentro del bloque de datos y un espacio reducido en el cual los registros de sobreflujo son colocados, es importante hacer notar que este espacio de sobreflujo es un factor del tamaño del bloque completo.

Un bloque de índices contiene parejas de llaves y de direcciones y un pequeño espacio para agregar tales parejas. Una pareja llave dirección está compuesta de la llave más baja de un bloque de datos particular y la dirección del dato en que esta llave reside. El usuario puede seleccionar un factor del área de sobreflujo para el bloque de índices y puede especificar el número de niveles de índices para el archivo cuando es creado.

fuando más bloques de datos son creados, el bloque de índices llega a estar completo. Los sobreflujos en un bloque de índices son manejados en la misma forma que los bloques de datos. Un nuevo bloque es creado y la mitad de los registros índice en el bloque completo de índices son colocados en un nuevo bloque de índices. La razón para particionar el bloque de sobreflujo es eliminar el problema que continuamente se tiene al colocar registros de sobreflujo desde un bloque completo a un área de sobreflujo separada. Por supuesto que el proceso de partición requiere más memoria que el proceso de sobreflujo de un registro a la vez debido a que el espacio de partición debe ser reservado.

los registros borrados son procesados por el recolector de basura. Es decir, los agujeros dejados por los registros borrados son reemplazados por los registros activos con la llave más grande en el bloque. El área de registros activos y las áreas de partición son siempre áreas contiguas en un bloque.

Ahora se va a examinar el tipo de procesamiento que es realizado cuando se usa un archivo secuencial indexado. Por el momento debe estar claro que la organización de un archivo secuencial indexado es mucho más compleja que la de un archivo secuencial. Debido a esta complejidad, la mayoría de los sistemas operativos proveen facilidades o métodos de acceso que manejan los cambios de la estructura del archivo que pueden resultar de la inserción y eliminación de registros.

La principal ventaja de un archivo secuencial indexado es que los registros pueden ser procesados secuencial o directamente. El procesamiento secuencial de un archivo secuencial indexado es lógicamente idéntico al procesamiento secuencial de un archivo secuencial, los registros son procesados en una secuencia determinada de acuerdo a la llave. Los tipos de transacciones que pueden ser realizadas son lectura, edición, adición y eliminación de registros. Estos son acompañados en el nivel de usuario con instrucciones de lectura, escritura y reescritura. La forma en que se efectúan estas transacciones es esencialmente igual para el procesamiento secuencial tanto de los archivos secuenciales como de los archivos secuenciales indexados, la manera en que los registros son accesados es sustancialmente diferente, debido a las diferencias en las estructuras de los archivos.

Para concluir, se mencionarán las propiedades importantes de los archivos secuenciales indexados

- El archivo secuencial indexado provee un acceso razonablemente más rápido a los registros usando el procesamiento directo o el procesamiento secuencial

- Para archivos relativamente estáticos, el área de sobreflujo puede ser eliminada y el área de sobreflujo del cilindro puede ser minimizada, esto da como resultado un alto porcentaje de la utilización del disco

- Para archivos altamente volátiles, el tiempo de acceso para un registro llega a ser excesivo cuando las áreas de sobreflujo están llenas

- Las facilidades para el acceso secuencial indexado son generalmente dadas en la mayoría de los sistemas, esto releva al programador de una gran cantidad de trabajo detallado en el mantenimiento de las áreas de índices y de sobreflujo. En el nivel de programador, el procesamiento secuencial de un archivo secuencial indexado aparece idéntico al procesamiento secuencial de un archivo secuencial

1.3 Archivos directos

En un archivo directo, también llamado aleatorio, una transformación o mapeo es hecho de la llave del registro a la dirección de la localidad de almacenamiento en que reside el archivo. Un mecanismo usado para generar esta transformación es llamado algoritmo de dispersión. Los algoritmos de dispersión son aplicados para localizar los registros en una tabla de dispersión. Un algoritmo de dispersión consiste de dos componentes: una función de dispersión que define un mapeo del espacio llave al espacio dirección y una técnica de resolución de colisiones que resuelve conflictos que surgen cuando más de un registro llave es mapeado a la misma dirección

Los algoritmos de dispersión usados para archivos directos son muy similares a los usados para tablas. La principal diferencia conceptual se debe a las características físicas del almacenamiento externo, que son diferentes de las características del almacenamiento direccionado directamente usado para las tablas. En particular el tiempo de acceso para un registro en una tabla en memoria principal es del orden de microsegundos mientras que el tiempo de acceso para un registro en memoria externa es del orden de milisegundos. En suma los registros en un archivo son almacenados en áreas de almacenamiento direccionables, en las cuales cada área de almacenamiento direccionable contiene b localidades de registro. El número de registros en un área de almacenamiento direccionable es llamado la capacidad del área de almacenamiento direccionable.

Básicamente se puede pensar en un área de almacenamiento direccionable como un sector en un dispositivo direccionable por sector, o como un bloque en un dispositivo direccionable por bloque. Para un registro en particular a ser aislado, el área de almacenamiento direccionable en la que el registro reside debe ser localizada, el contenido del área de almacenamiento direccionable es traído a un almacenamiento temporal en memoria, y después el registro deseado es extraído del almacenamiento temporal.

Se va a definir un espacio de dirección A de tamaño a tal que el conjunto $A=\{i+1, i+2, \ldots, i+a\}$, donde i es una constante entera. Entonces ab registros pueden ser acomodados por A, y el factor de carga para el archivo directo es $n/(ab)$, asumiendo que una llave de tamaño n es mapeada dentro del espacio dirección.

Un conjunto llave $S=\{x_1, x_2, \ldots, x_s\}$ es un subconjunto de un conjunto x de posibles llaves que es llamado el espacio llave. Si el tamaño de x es igual al número de localizaciones de registro en A, y la llave es consecutiva, entonces una transformación puede ser definida para asignar a cada área de almacenamiento direccionable de A exactamente b llaves desde x. Este tipo de transformación uno a uno es llamado direccionamiento directo. En la mayoría de las situaciones S es un pequeño subconjunto de x y resulta muy bajo el uso del direccionamiento directo. Si S es mapeado dentro de A con todas las distintas posibilidades los registros serán asignados a la misma área de

almacenamiento direccionable a menos que un área de almacenamiento direccionable de sobreflujo se lleve a cabo cuando esto sucede una técnica de manejo de sobreflujo de área de almacenamiento direccionable debe ser usada para almacenar los registros de sobreflujo. Las técnicas de manejo de sobreflujo de área de almacenamiento direccionable son muy similares a las de resolución de colisiones en una tabla.

Las funciones de dispersión se dividen en dos clases generales, distribución independiente y distribución dependiente.

El método de división, usando un divisor primo o un divisor que es relativamente primo con el tamaño del espacio dirección, tuvo el mejor rendimiento en promedio. Esto no quiere decir que para ciertos conjuntos llave con ciertos factores de carga y capacidades de área de almacenamiento direccionable, uno de los otros métodos existentes no pueda reemplazar al método de división.

Recientemente han sido tremendamente interesantes las técnicas de dispersión que facilitan a un archivo crecer dinámicamente sin requerir una cantidad significativa de redispersión. Estas técnicas son particularmente aplicadas a la organización de archivos directos. Aunque estas involucren transformaciones de dirección más complicadas, resulta significativamente reducido el número de colisiones.

El segundo aspecto de los algoritmos de dispersión es la técnica de resolución de colisiones. En un archivo directo, la unidad más pequeña direccionable es el área de almacenamiento direccionable, que puede contener muchos registros que han sido mapeados a la misma dirección. Por lo tanto, en un archivo directo con una capacidad de área de almacenamiento direccionable dada, un cierto número de colisiones son esperadas. Cuando hay más registros colisionados que la capacidad del área de almacenamiento direccionable entonces algún método debe ser encontrado para el manejo de esos registros de sobreflujo. El término técnica del manejo de sobreflujo es usado en lugar de técnica de resolución de colisiones, que es comunmente adoptado para los métodos de tablas de dispersión.

La misma clasificación general puede ser aplicada para las técnicas de manejo del sobreflujo. Cuando usamos un área de almacenamiento direccionable con una capacidad mayor que uno, estamos de hecha implementado una forma restringida de prueba lineal de un direccionamiento abierto. Cuando un registro es agregado a un área de almacenamiento direccionable que no está llena, el nuevo registro es agregado a la próxima localidad abierta. Seguramente que la próxima localidad de registro abierta está en el área de almacenamiento direccionable y no está reservada para registros que son agregados a las direcciones de esa área de almacenamiento direccionable.

El área de almacenamiento direccionable referenciada por el cálculo de la dirección de un registro es conocido como el área de almacenamiento direccionable primaria para ese registro. Si un registro no está presente en el área de almacenamiento direccionable primaria, está localizado en un área de almacenamiento direccionable de sobreflujo, o no está en el archivo.

Desde que el contenido completo de un área de almacenamiento direccionable es traído a memoria principal por un requerimiento, es extremadamente beneficio que el registro deseado esté localizado en alguna parte del área de almacenamiento direccionable primaria. Si el registro no está en el área de almacenamiento direccionable primaria, un requerimiento debe ser hecho para traer un área de almacenamiento direccionable de sobreflujo, la cual es determinada por el método de manejo de sobreflujo.

Si el método de sobreflujo de prueba lineal de direccionamiento abierto es usado, entonces una búsqueda sucesiva es hecha de los registros en las áreas de almacenamiento direccionables sobrantes del archivo. La búsqueda es terminada satisfactoriamente cuando el registro es localizado. Es terminada insatisfactoriamente si un registro vacío es encontrado, o si la búsqueda retorna al área de almacenamiento direccionable originalmente probada.

Una prueba aleatoria de direccionamiento abierto o de doble dispersión no son necesariamente buenos métodos de sobreflujo en el uso de áreas directas. En estos métodos las secuencias de áreas de almacenamiento direccionables de sobreflujo que son examinadas no exhiben la propiedad de adyacencia física. Es decir, dos áreas de almacenamiento direccionables que son adyacentes en la secuencia de sobreflujo no son necesariamente adyacentes físicamente. La adyacencia física puede ser importante, ya que los registros que no son adyacentes físicamente tienen una alta probabilidad de requerir una búsqueda en un dispositivo de almacenamiento de cabeza móvil. El tiempo de la búsqueda extra puede ser prohibitivo.

Los registros de sobreflujo pueden ser encadenados desde el área primaria a un área de sobreflujo separada. Un registro de sobreflujo debe residir en un área de almacenamiento direccionable de sobreflujo que está en la misma área de búsqueda del área primaria para dicho registro de sobreflujo. Particularmente, una buena estrategia es reservar las últimas áreas de almacenamiento direccionables de un área de búsqueda estrictamente para los registros de sobreflujo de las áreas de almacenamiento direccionables primarias en esa área. Esta es la estrategia adoptada cuando se ve el área de sobreflujo de un cilindro en la organización secuencial indexada. Sin embargo, puede ser difícil adoptar tal estrategia de configuración, ya que las áreas de almacenamiento direccionables de sobreflujo podrían romper el esquema de direccionamiento lineal requerido para un direccionamiento directo. Por lo tanto, un área de sobreflujo independiente que está totalmente separada del área primaria puede ser requerida.

Una estrategia final de sobreflujo también involucra encadenamiento. Existe un método llamado encadenamiento con listas unidas. En este método, los registros de sobreflujo son localizados usando apuntadores de un área de almacenamiento direccionable a otra. Por lo tanto, cuando una llave es dispersada en un área de almacenamiento direccionable, una búsqueda comienza a través de una cadena de áreas de almacenamiento direccionables hasta el registro requerido o hasta que una localidad de almacenamiento vacía es encontrada. Las llaves pueden ser mapeadas a áreas de almacenamiento direccionables que no sean las primeras en una cadena; es necesario que las listas estén unidas. En una cadena de áreas de almacenamiento direccionables, las llaves que son originalmente dispersadas a diferentes direcciones pueden ser encontradas.

Está demostrado que el encadenamiento con listas separadas requiere la longitud promedio más pequeña de búsqueda sin embargo, esta aparente ventaja es superada debido a dos factores. Primero, el área de sobreflujo separada no cuenta como parte del espacio total del archivo. Por lo tanto, si n registros son agregados al archivo, todos los n registros son localizados en el área primaria, usando direccionamiento abierto o encadenamiento con listas unidas. Si hay s sobreflujos entonces n-s registros son localizados en el área primaria cuando se usa encadenamiento con listas separadas. Por lo tanto, el factor efectivo de carga es menor para el encadenamiento con listas separadas.

El segundo factor contribuye al mejor rendimiento de cualquier método de encadenamiento sobre un método de dirección abierta. Para el método de encadenamiento, la información concerniente a la localización del próximo registro de sobreflujo es almacenada en el mismo registro. Así que, al acceder un registro de sobreflujo particular, no hay necesidad de examinar un número intermedio de registros que pueden no ser registros de sobreflujo. Se debe hacer notar que las ligas del almacenamiento de sobreflujo amplían el tamaño del registro. Esto puede ser una consideración importante para el diseño del archivo, especialmente si el archivo consiste de muchos pequeños registros.

Un punto final que debe tenerse en mente es que el acceso a un área de sobreflujo independiente probablemente resultará en un requerimiento de búsqueda. Por lo tanto, si los dispositivos externos de cabeza móvil están siendo usados, el encadenamiento a un área de sobreflujo independiente puede no ser un método eficiente. Un posible compromiso es mezclar el encadenamiento con listas unidas.

Se ha centrado la explicación de las estructuras de archivo para archivos directos sobre las técnicas de dispersión. Hay otros caminos para organizar un archivo directo que son menos populares pero no obstante pueden ser aplicados en algunas situaciones.

Si el número de registros en el archivo es relativamente pequeño y el tamaño del registro es relativamente grande entonces puede ser que valga la pena considerar un esquema de direccionamiento directo. Algunos métodos para llevar a cabo la translación directa de la dirección involucran el uso de referencias cruzadas o indexación. Una tabla de

referencias cruzadas es simplemente una lista de llaves y direcciones en la que una única dirección de almacenamiento externo es asignada a cada llave. La localización de un registro desde su llave es simplemente recuperar la dirección externa asociada con la llave y después usar un comando de entrada/salida que recupere directamente el registro deseado. En la mayoría de los lenguajes de programación las listas asociadas no son previstas y el programador debe escribir la lista de referencias cruzadas. La lista puede ser guardada como una lista desordenada y tener a menudo dificultad una búsqueda cuando es deseado recuperar una dirección. Alternativamente la lista puede ser implementada como una lista ordenada por la clave y una búsqueda binaria puede ser empleada para localizar más rápidamente una dirección. La adición y eliminación de registros presentan problemas similares a que la lista debe ser mantenida en orden.

Los métodos de indexación incluyen árboles binarios, árboles B arios, y otros tipos de estructuras que pueden ser escogidas para llevar acabo directamente directo en archivos directos. Con los métodos de árbol estructurado la adición y eliminación de registros puede ser manejada más eficientemente.

En este explicación se ha asumido que si un registro deseado no está localizado en su área de almacenamiento direccionable primaria, entonces una serie de comandos de entrada salida es usado. Este comando trae un área de almacenamiento direccionable de sobreflujo el cual es examinado para buscar el registro requerido. La estrategia de búsqueda es válida para unidades de memoria externa que son direccionables por sector. Algunos dispositivos direccionables por registro son capaces de localizar un registro particular en una pista dada basándose en la llave del registro. En otros dispositivos, gracias a su soporte físico, se elimina mucho de la inspección de las áreas de almacenamiento direccionables de sobreflujo en memoria principal debido al hecho que se usan dispositivos direccionables por sector.

El procesamiento del archivo directo es dependiente de la forma en que la llave del registro es transformada a una dirección del dispositivo externo. Los archivos directos son primeramente procesados directamente. Es decir, una llave es mapeada a una dirección y dependiendo de la naturaleza del archivo de transacción, un registro es creado es borrado, es salvado o es accesado en dicha dirección o posiblemente en algunas direcciones subsecuentes si una

colisión se lleva a cabo. Esto que la dirección subsecuente es determinada por la técnica de manejo de sobreflujo que es adoptada. Cuando un manejo de sobreflujo es proporcionado usando encadenamiento con listas separadas, un apuntador a la lista ligada de los registros de sobreflujo es incluido en cada área de almacenamiento direccionable. El acceso directorio directo se ha creado exclusivamente para computadoras que utilizan dispositivos de almacenamiento en disco. Las técnicas de acceso directo tienen frecuente uso en directorios, tablas de artículos y precios, listas de nombres etc. De hecho son frecuentemente utilizados en aplicaciones en donde la longitud de los registros de datos es corta y fija, y el acceso rápido a los datos es esencial siendo estos accesados siempre de manera simple y rapida. En estos casos la organización de los archivos directos es la única adecuada. Acceso simple significa el uso de una llave simple para accesar la información.

El método de acceso directo simple es con frecuencia el más rápido conocido para accesar registros físicos, sin embargo existen tres características que hacen de este método a menudo impractico, tales características son :

- Su dependencia de registros de longitud fija.
- Los argumentos de recuperación del dato están limitados solo a un atributo de la llave.
- No existe la filosofía de acceso en serie, ya que para conocer el siguiente dato al actual se requiere conocer previamente la llave.

Actualmente en muchas aplicaciones existen adaptaciones que se pueden utilizar para superar estas limitaciones. Una de ellas es el almacenamiento en bloques. Cuando se utiliza este tipo de almacenamiento se pueden tener registros de

[illegible] registros estarán ordenados en un bloque. Para poder efectuar este tipo de organización se requiere una estructura de datos que indique cual es el siguiente registro lógico que sucede al actual. Este método de organización es muy complejo, sin embargo permite la búsqueda de información en forma serial.

Acceso directo por atributos múltiples

Este método de acceso es necesario cuando el requerimiento para acceder un registro no está limitado solo a un atributo de una llave. Bajo estas condiciones el método de acceso directo simple no soluciona el problema, ya que es necesario poder acceder con atributos diferentes un mismo registro.

Para poder accesar la información en estas condiciones se requiere tener una lista de direcciones por cada llave diferente que se quiera localizar. Cada espacio de direcciones podrá ser localizado con una función de transformación diferente para cada lista de apuntadores.

Este método ayuda a ahorrar espacio en disco ya que solo es necesario mantener un registro de datos completo el cual puede ser accesado por diferentes atributos.

Las listas de apuntadores contienen solamente la dirección en donde se encuentra el registro de datos actual. Con este método se pueden utilizar múltiples listas de apuntadores para accesar el registro de datos por llaves diferentes.

Los registros de datos serán encontrados de acuerdo a alguna llave por lo que deberá existir una lista de apuntadores y una regla de transformación para cada uno de ellos. El valor de la dirección que contenga el espacio de direcciones más pequeño indicará la dirección en donde se encuentra el registro de datos.

El registro de datos puede ser localizado directamente por su primer llave e indirectamente por sus llaves alternas. Con este método no todos los atributos requieren tener espacios de direcciones diferentes, el espacio dependerá de la longitud en caracteres del atributo más un espacio para escribir la dirección del registro de datos.

Así también el espacio de datos podrá tener una organización diferente ya sea un primario, secundario o alternativo.

Una inserción dentro de un archivo que utiliza listas de apuntadores multiples requiere que todas las listas de apuntadores sean actualizadas, así también una modificación a uno de estos atributos requiere que se actualicen todas las listas.

Todas las entradas con el mismo valor de llave tendrán la misma dirección en el espacio de apuntadores, esto incrementa el riesgo de colisiones para atributos los cuales no son únicos, por ejemplo, una fecha, nacionalidad, sexo etc.

Cuando se trabaja con archivos directos simples cuyos registros son almacenados por un solo atributo, esperamos o es deseable que estos sean únicos, por ejemplo, un número de cuenta, un número de asiento, un número de empleado, un registro federal de causantes etc., de esta forma solo corresponderá una dirección en el espacio de direcciones de datos.

Algunos llaves del registro de datos pueden no ser únicos, en estas condiciones la función de transformación asignará la misma dirección a atributos repetidos, esta situación puede resolverse mediante el mecanismo de almacenar los registros en bloques dentro de un mismo registro físico.

Las principales propiedades de este tipo de archivos son:

- El acceso directo a registros en un archivo directo es rápido, especialmente para archivos con un bajo factor de carga y pocos registros de sobreflujo.

- Debido a que cierta sección del archivo permanece sin uso para prevenir un excesivo número de registros de sobreflujo, la utilización del espacio para un archivo directo es pobre comparada con otros tipos de organizaciones de archivo.

- El rendimiento conseguido usando un archivo directo es muy dependiente del algoritmo de transformación llave a dirección adoptado. La transformación que se usada es una aplicación dependiente y es generalmente implementada y mantenida a través de los programas de usuario.

- Los registros pueden ser accesados serialmente pero no secuencialmente a menos que una lista ordenada de llaves separada sea mantenida.

2 DISEÑO DE LAS ESTRUCTURAS DE DATOS

Las estructuras de datos que se utilizarán se exponen en este capítulo pero antes es conveniente recordar los conceptos básicos de las diferentes estructuras de datos existentes.

2.1 Descripción de las estructuras de datos tradicionales

Las estructuras de datos se pueden dividir en dos grupos: lineales y no lineales. Se dice que una estructura de datos es lineal cuando los datos se encuentran almacenados en localidades de memoria adyacentes y es no lineal cuando los datos se almacenan en localidades dispersas.

2.1.1 Estructuras de datos lineales

Los elementos enteros, reales, caracteres son llamados primitivos porque dentro del repertorio de instrucciones de una computadora existen instrucciones que manipulan estas estructuras primitivas.

Las estructuras de datos no primitivas pueden ser clasificadas como arreglos, registros, y listas. Una variable de tipo estructurado difiere de una de tipo simple en que las variables de un tipo estructurado tienen más de una componente. Cada componente de un tipo estructurado es una variable que puede tener una estructura simple o estructurada. En el nivel más bajo, las componentes de una variable estructurada tienen tipos simples, y a éstos pueden ser asignados valores y usados en expresiones en la misma forma que las variables simples. La cuestión fundamental de una variable estructurada es la manera en la cual sus componentes son accesadas. Las variables del tipo arreglo y registro nos permiten usar la memoria del computador en forma más flexible. El tipo archivo permite que la información almacenada en medios externos a la memoria sea accesada.

Un arreglo es una colección de variables las cuales todas tienen el mismo tipo. Una línea de un texto puede ser representada como un arreglo de caracteres, un vector puede ser representado como un arreglo de números reales, y como una matriz consiste de columnas, cada una de las cuales es un vector, una matriz puede ser representada como un arreglo de vectores.

Un tipo arreglo es declarado en términos de un tipo índice y un tipo componente. Los elementos del arreglo son almacenados en palabras consecutivas de la memoria de la computadora. Esta es una forma eficiente de almacenar las componentes enteras y reales, porque en muchas computadoras éstas requieren una palabra entera de la memoria o más

Sin embargo, este no es siempre una manera eficiente de almacenar variables de otros tipos, porque el espacio puede desperdiciarse. La cantidad de espacio desperdiciado puede reducirse empaquetando varios componentes de un arreglo dentro de cada palabra.

Un arreglo es un conjunto ordenado que consiste de un número fijo de elementos. Las operaciones de eliminación e inserción de elementos no pueden ser realizadas sobre arreglos.

Un registro, como un arreglo, es una variable estructurada por varios elementos. Los elementos de un registro pueden tener tipos diferentes y a ellos se accesa por nombre y no por subíndice.

Tanto arreglos como registros son meras abstracciones de formas de almacenamiento de datos usados en el nivel de lenguaje de máquina.

Los nombres de los elementos deben ser únicos dentro del registro. Los nombres de los elementos se pueden usar para denotar una variable o un elemento de otro registro. No existen operadores que puedan usar registros como operandos, sin embargo, el valor de un registro puede ser asignado a otro registro por una proposición de asignación.

Una lista es un conjunto ordenado que consiste de un número variable de elementos y donde se pueden hacer inserciones y eliminaciones de sus elementos. Una lista que despliega el parentesco de adyacencia entre sus elementos se dice que es lineal. Cualquier otra lista se dice que es no lineal.

Las operaciones que se pueden hacer en una lista son las mismas que se hacen sobre un arreglo, pero hay una importante diferencia: que el tamaño de la lista puede ser cambiado continuamente, la inserción y eliminación de elementos en una lista es especificado por la posición. Es decir, se puede borrar el elemento i-ésimo de una lista o insertar un nuevo elemento antes o después del elemento i ésimo. Cada elemento en una lista está compuesto de uno o más campos. Un campo puede ser considerado la pieza más pequeña de información que puede ser referenciada en un lenguaje de programación. Las siguientes operaciones se pueden realizar sobre listas:

- Combinar dos o más listas para formar otra lista.
- Dividir una lista en varias listas.

- Copiar una lista.

- Determinar el número de elementos en una lista.

- Clasificar los elementos de una lista en orden ascendente o descendente, dependiendo de ciertos valores de uno o más campos dentro de un elemento.

- Buscar un elemento dentro de una lista que contiene un campo con cierto valor.

Un archivo es una gran lista que está almacenada en memoria externa de una computadora. Los archivos permiten almacenar grandes cantidades de datos, evitando con esto volver a dar datos a un programa cada vez que se ejecute. Los archivos pueden ser manipulados desde diferentes programas con diferentes propósitos.

La estructura de un archivo es semejante a la de un arreglo unidimensional. En un arreglo el índice de un elemento puede ser un entero negativo, cero o un entero positivo. En un archivo el valor del índice del primer elemento es siempre cero y el valor del último es n-1 siendo n el número de registros o elementos de los que consta el archivo. El elemento o registro de un archivo puede ser una variable simple (entera, real, etc.) o una variable estructurada (registro). La dimensión de un arreglo permanece sin cambio a lo largo de la ejecución de un programa, mientras que un archivo solo puede crecer (agregar registros) a lo largo de la ejecución de un programa.

Los archivos son importantes por tres razones. Primero, un proceso solo se puede comunicar con su medio ambiente por medio de archivos. Segunda, un proceso es usualmente de vida corta, un programa se carga en la memoria principal y se ejecuta, y tan pronto como termina la memoria es usada por otro programa. Si el programa no modifica un archivo durante su ejecución, no habrá evidencia de que éste corrió completamente. La tercera razón para la importancia de los archivos es que pueden almacenarse cantidades más grandes de datos en un archivo que en la memoria principal.

Una de las estructuras de datos lineales de tamaño variable más importante es la pila. Es la forma más general de una lista lineal: se pueden insertar o eliminar elementos que se encuentren en cualquier posición. Una pila es una subclase de lista que permite la inserción y la eliminación de un elemento solamente a través de un extremo. La operación de inserción es conocida como meter y la operación de eliminación es conocida como sacar. El mayor y el menor elemento accesible en la pila son conocidos como el tope y el fondo respectivamente.

Ya que las operaciones de inserción y eliminación de elementos son realizadas por uno de los extremos de la pila sus elementos solo pueden ser retirados en orden inverso en el que ellos fueron metidos a la pila, esta lista lineal es frecuentemente conocido como UEPS (último en entrar, primero en salir).

Las operaciones sobre una pila pueden ser simuladas mediante el uso de un vector consistente de un número de elementos suficientes de acuerdo a las posibles inserciones que se quieran hacer a la pila.

Un apuntador guarda el estado del tope de la pila, inicialmente, cuando la pila está vacía el apuntador tiene un valor de cero y cuando la pila tiene un solo elemento el apuntador tiene un valor de uno, y así sucesivamente. Cada vez que un nuevo elemento es insertado en la pila, el apuntador es incrementado en uno, antes de que el elemento sea colocado en la pila. El apuntador es decrementado en uno cada vez que una eliminación es hecha. Las pilas tienen aplicación sobre procesos de recursividad y de notación infija.

Otro importante tipo de lista es la cola que permite que las eliminaciones sean realizadas por uno de sus extremos y las inserciones por el otro. La información de esta lista es procesada de la misma forma en que fue recibida, es decir, primero en entrar, primero en salir (PEPS). Existen dos apuntadores en una cola, uno en el extremo por el que entran los datos y otro en el extremo por el que salen, cuando la cola está vacía o tiene un solo elemento ambos apuntadores coinciden.

2.1.2 Estructuras de datos no lineales

Una gráfica ó digrafo (como G=(A,R)), es una relación de R sobre un conjunto A. Los elementos de A son llamados nodos y los elementos de R son llamados arcos.

Si en un arco es importante considerar el nodo inicial y el nodo final entonces se está hablando de un arco dirigido. Una gráfica cuyos arcos son todos dirigidos es llamada gráfica dirigida. El grado externo de un nodo es el número de arcos que salen de él, el grado interno de un nodo es el número de arcos que llegan a dicho nodo. Si en una gráfica dirigida el nodo final de un arco es el nodo inicial de otro arco y así sucesivamente entonces se dice que se trata de una trayectoria, si el nodo inicial y final de la trayectoria es el mismo se trata de un ciclo. Una gráfica que contiene al menos un ciclo es llamada gráfica cíclica de otra forma es llamada acíclica. Cuando los arcos de la secuencia son distintos la trayectoria es simple. Si los arcos son distintos y contienen a todos los nodos de A, la trayectoria es hamiltoniana. La longitud de una trayectoria es el número de arcos que la componen.
Si en un arco no es importante considerar cuál es el nodo inicial ni cuál es el nodo final se habla de un arco no dirigido. Un arco cuyo nodo inicial y final son los mismos se conoce como un lazo, la dirección de un lazo no tiene ningún significado y puede ser considerado como un arco dirigido o no dirigido. Una gráfica es no dirigida cuando todos los arcos son no dirigidos.

Una gráfica es mixta cuando contiene arcos dirigidos y no dirigidos. Una gráfica que tiene algunos nodos unidos por más de dos arcos, inclusive, con el mismo sentido, son llamados arcos paralelos. Una gráfica que contiene arcos paralelos se llama multigráfica. En caso de contener un arco entre cualquier par de nodos se llama gráfica simple. Existen gráficas a cuyos arcos se les asigna valores, estos valores son llamados peso del arco y dan origen a una gráfica llamada pesada.

Una gráfica dirigida, con n nodos, puede ser completamente especificada con una matriz de orden n llamada matriz de adyacencia. Para definir la matriz debe considerarse que cualquier elemento X_{ij} de la matriz es igual a 1 si existe un arco entre los nodos i y j, es igual a cero si no hay arco. En una gráfica dirigida y pesada el elemento X_{ij} de la matriz de adyacencia es igual al peso del arco si el arco existe entre los nodos i y j y es igual a cero si el arco no existe. Una gráfica no dirigida, pesada o no, queda representada por una matriz triangular, ya que los arcos del nodo i al nodo j y del nodo j al nodo i son los mismos.

Un árbol es una gráfica G=(A,B) en la que

- El número de nodos es igual al número de arcos más uno.
- Todos los nodos son de grado interno uno, excepto un nodo llamado la raíz, de grado cero.
- No hay ciclos.
- Cualquier trayectoria es simple.
- Entre cualquier par de nodos sólo hay una trayectoria.
- Cualquier arco es un arco de desconexión.

En la terminología que se emplea para el estudio de los árboles se encuentran los términos siguientes :

- Se define como grado externo de un nodo al número de sus subárboles.
- Una hoja o nudo terminal es una hoja de grado externo cero.
- Un nodo ramal es un nodo de grado externo mayor que cero.
- El nivel de un nodo es el nivel de su antecesor directo más uno.
- Los nodos de un árbol reciben nombres tales como el nodo a es padre de los nodos b, c y d ya que el nudo a es su antecesor directo. Los nodos b, c y d son hermanos ya que se encuentran al mismo nivel (figura 2.1).

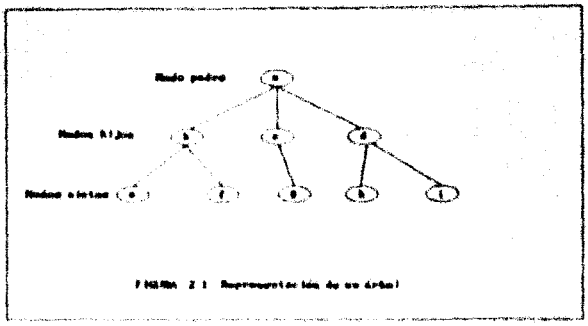

FIGURA 2.1 Representación de un árbol

Un árbol es ordenado cuando los subárboles tienen un orden cuando esto no ocurre el árbol es orientado. En este último caso, si la dirección de los arcos se ignora, el árbol es libre.

Un árbol puede ser completamente especificado por una matriz de adyacencia de forma similar a una gráfica dirigida. Para especificar el nodo raíz, es posible utilizar los elementos de la diagonal principal que siempre estarán disponibles. Cuando un árbol es libre, la matriz de adyacencia es triangular, y se especifica en forma similar a una gráfica no dirigida.

Un árbol particular que tiene gran importancia en el área de las ciencias de la computación es el árbol binario. Esto se debe fundamentalmente a la sistematización que puede lograrse para su representación. Un árbol binario es un árbol en el que cualquier nodo tiene cero, uno o dos subárboles. Cuando tiene exactamente cero o dos subárboles es llamado árbol estrictamente binario, de otra forma, es un árbol de knuth.

Un árbol estrictamente binario es balanceado cuando el número de nodos terminales n es igual a 2^m, y la longitud de cualquier trayectoria de la raíz a cualquier nodo terminal es igual a m, donde m es cualquier número entero no negativo, o si $2^m<n<2^{m+1}$ y la longitud de las trayectorias es m o m+1.

2.2 Diseño de las estructuras de datos

Las estructuras de datos más importantes que se utilizan son cinco: la primera es para el archivo directo principal para el área de sobreflujo y para el área de sobreflujo especial, que en este caso en particular es de una llave principal, dos llaves secundarias y un solo campo de información; la segunda es para los parámetros de control del archivo directo principal; la tercera es para el archivo de apuntadores por cada subllave; la cuarta es para los parámetros de control del archivo directo de apuntadores de cada subllave y; la quinta es para la localización de los archivos. Las estructuras, para este caso particular de una llave principal y dos llaves secundarias se muestran a continuación:

Registro del archivo directo principal, de su área de sobreflujo y del archivo directo de sobreflujo especial:

llave uno - llave principal de 4 caracteres

llave dos - llave secundarias de 2 caracteres

llave tres - llave secundaria de 1 caracter

otro dato - campo de información de 1 caracter

Parámetros de control del archivo directo principal:

- llave mínima - llave principal mínima de 4 caracteres
- llave máxima - llave principal máxima de 4 caracteres
- dirección mínima - dirección mínima del archivo directo principal
- dirección máxima - dirección máxima del archivo directo principal
- frecuencia - número de llaves existentes entre la llave principal mínima y la máxima
- llave promedio - promedio de las llaves existentes entre la llave principal mínima y la máxima
- dirección promedio - dirección mínima más dirección máxima entre dos

Registro del archivo de apuntadores por cada subllave

- dirección anterior - dirección del archivo directo de apuntadores que precede a la dirección actual en el encadenamiento de direcciones
- dirección siguiente - dirección del archivo directo de apuntadores que sigue de la dirección actual en el encadenamiento de direcciones
- subllave - valor de la subllave en cuestión
- dirección principal - dirección del archivo directo principal en el que se encuentra la subllave

Parámetros de control del archivo directo de apuntadores por cada subllave

- dirección inicial - dirección donde comienza el encadenamiento de direcciones
- dirección final - dirección donde termina el encadenamiento de direcciones

Tabla de localización de archivos directos

Esta tabla tiene como objetivo definir un espacio de direcciones global, los espacios de direcciones que correspondan al espacio principal así como los espacios de los apuntadores de las llaves alternas.

Tiene la siguiente estructura

- número de archivo - indica el archivo al que se debe referir el usuario para accesar datos de este espacio de direcciones.

- dirección inicial - este campo contiene un número el cual indica en donde inicia el espacio de direcciones del archivo en referencia

- dirección final - este campo contiene un número el cual indica en donde termina el espacio de direcciones del archivo en referencia. Cuando se trata del archivo principal el espacio entre dirección inicial y dirección final incluye el espacio de direcciones para el area de sobreflujo el cual se determina calculando un diez por ciento del espacio total.

2.3 Ventajas de las estructuras de datos diseñadas

Con respecto a la estructura de datos del archivo directo principal, que es la misma para las dos áreas de sobreflujo, solo se trata de ejemplificar la utilización del manejador de archivos de acceso directo por multiples llaves ya que como puede notarse sólo se tiene un campo de información de 1 caracter

La ventaja de los parametros de control del archivo directo principal es la de dividir el intervalo de llaves en $n_{numcalle}$ subintervalos ya que dichos elementos no son continuos, es decir que si se tiene una llave y no necesariamente existe una llave y-1 y otra llave y+1, si no que por lo general una llave y está entre una llave x a y una llave y+b donde a y b no son iguales en la mayoría de los casos. Por otro lado, una dirección d siempre se encuentra entre una dirección d-1 y otra dirección d+1, excepto sólo para los casos de la dirección mínima y máxima del intervalo de direcciones donde se tendrá una d+1 y una d-1 respectivamente, es decir dicho intervalo es continuo entre la dirección mínima y la dirección máxima (figura 2.2)

Se escogió dividir el intervalo en $10^{numcaltla\ 1}$ donde numcalla es el número de caracteres de la llave principal ya que esto representa una milésima parte del intervalo total de llaves lo cual implica que si se tiene una llave principal de 6 caracteres habrá 1000 subintervalos y cada subintervalo tendrá sus propios parámetros de control, con esto se asegura una dispersión más eficiente en el archivo directo principal como se mostrará más adelante en este trabajo

Cada subintervalo i tiene una llave mínima $llamin_i$, una llave promedio $llapro_i$, una llave máxima $llamax_i$ y una frecuencia de llaves $frella_i$ las cuales están asociadas con una dirección mínima $dirmin_i$, una dirección promedio $dirpro_i$, una dirección máxima $dirmax_i$ y una frecuencia de direcciones $fredir_i$ respectivamente figura (2.2).

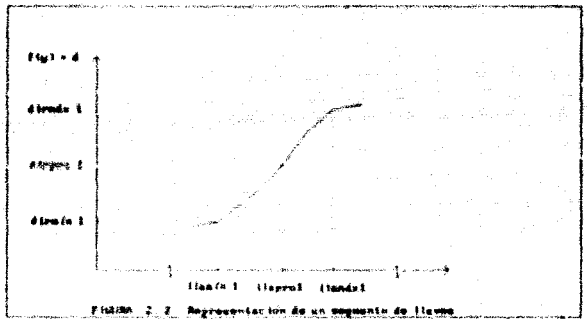

FIGURA 2.2 Representación de un segmento de llaves

Es importante hacer notar que $frella_i$ es igual a $fredir_i$ con el fin de asegurar una dirección por cada llave que exista en el intervalo i.

El registro del archivo directo de *apuntadores* por cada subllave facilita el acceso por una de las llaves secundarias al archivo directo principal ya que los registros del archivo directo de apuntadores se encuentran encadenados mediante los campos de dirección anterior y dirección siguiente para cada subllave particular, y en cada registro encadenado se encuentra la dirección del archivo directo principal en la cual aparece la subllave por la

que solo se necesitan dos accesos por registro cuando se desea accesar al archivo directo principal por alguna subllave, un acceso al archivo directo de apuntadores donde se obtiene la dirección del archivo directo principal en que se encuentra dicha subllave y el acceso al archivo directo principal.

La ventaja de los parámetros de control del archivo directo de apuntadores es que indican el inicio y el final del encadenamiento de direcciones para una subllave en particular, lo cual implica que si una subllave es de 5 caracteres habrá 1000 parámetros de control para el archivo directo que contenga los encadenamientos de direcciones para dicha subllave.

Es conveniente tener una tabla de localización de archivos ya que de esta forma se pueden conocer los límites de cada archivo directo sin tener que hacer algún otro cálculo.

3 DISEÑO DE LA FUNCION DE DISPERSION Y DE LA RESOLUCION DE COLISIONES

Un manejador de archivos de acceso directo está formado principalmente por un método de dispersión y un método de resolución de colisiones los cuales se explicarán a continuación.

3.1 Descripción de los métodos de dispersión tradicionales

Las funciones de dispersión se pueden dividir en tres grandes clases llamadas: funciones de distribución independiente, funciones de distribución dependiente y funciones de dispersión dinámica. Una función de dispersión de distribución independiente no usa la distribución de llaves para calcular la posición de un registro. Una función

de dispersión de distribución dependiente es obtenido después de haber examinado las llaves correspondientes de los registros conocidos. Una función de dispersión dinámica se utiliza cuando el tamaño de la tabla de dispersión debe ser estimado durante el avance.

3.1.1 Métodos de dispersión de distribución independiente

Algunas de las propiedades deseadas de una función de dispersión incluyen la rapidez y la generación de direcciones uniformemente. Algunas llaves tienen caracteres alfanuméricos y es conveniente convertir tales caracteres a una forma para que puedan ser más fácilmente manipulados por la función de dispersión, este proceso es conocido como preacondicionamiento. Dicho preacondicionamiento es más productivo si se usa el código numérico de representación interna, como por ejemplo, ASCII o EBCDIC.

El método de la división

Una de las primeras funciones de dispersión y quizás la más empleamente aceptada es el método de la división que se define como:

$$H(x) = x \bmod m + 1$$

donde m es un divisor entero. El término x mod m tiene un valor que es igual al residuo de dividir x por m. En general, es muy poco común que un número de llaves produzcan el mismo residuo cuando m es un número primo muy grande.

El método del centro del cuadrado

Otro método que ha sido ampliamente usado en muchas aplicaciones es el método del centro del cuadrado. En este método una llave es multiplicada por sí misma y la dirección es obtenida seleccionando un número apropiado de bits o dígitos del centro del cuadrado. El número de bits o dígitos escogidos deben ajustarse a una palabra de memoria de computadora.

El método aditivo

En el método aditivo una llave es seccionada en un número de partes cada una de las cuales tiene la misma longitud requerida para la dirección con la posible excepción de la última parte. Dichas partes son después sumadas, ignorando el acarreo final, para formar la dirección. Si las llaves están en forma binaria, la operación de or exclusivo puede ser substituida por la adición. A este método se le conoce como el método aditivo directo. Existe una variación de este método conocida como el método aditivo inverso en el cual se invierte el orden de los componentes de cada sección de la llave antes de hacer la suma.

El método de longitud dependiente

Otra técnica de dispersión que ha sido ocasionalmente usada es llamada el método de longitud dependiente. En este método la longitud de la llave es usada junto con una parte de la llave para producir la dirección inmediatamente o una llave intermedia que será usada por otro método para producir, finalmente, una dirección.

ii método multiplicativo

Para una llave entera no negativa x y una constante c tal que 0<c<1, se tiene la siguiente función de dispersión

H(x) = trunc(fcx mod 1)) + 1

Donde cx mod 1 es la parte fraccional de cx. Esta función de dispersión multiplicativa debe dar buenos resultados si la constante c es adecuadamente escogida, una selección que es difícil de hacer.

Aunque algunos de estos métodos dan, frecuentemente, una distribución uniforme de las llaves sobre las direcciones, es necesario experimentar con funciones de dispersión aplicadas a un conjunto de llaves específicas. Se necesita una medida del rendimiento para comparar las diferentes funciones de dispersión y la medida más ampliamente aceptada es la longitud promedio de búsqueda. Para un conjunto de registros en un archivo directo, es el número promedio de accesos al dispositivo de almacenamiento requerido para recuperar un dato. Usualmente, la mejor función de dispersión para usar con un conjunto particular de llaves minimiza la longitud promedio de búsqueda.

3.1.2 Metodos de dispersión de distribución dependiente

Una o más funciones del conjunto de llaves correspondientes a los registros conocidos deben ser realizados antes de que estas funciones puedan ser definidas. Ya que las inserciones y eliminaciones en el archivo pueden cambiar el conjunto de llaves drásticamente la redefinición periódica de la función de dispersión y reorganización del archivo directo puede ser requerida

El método de análisis digital

Una función de dispersión conocida como análisis digital forma las direcciones mediante la selección e inversión de los dígitos o bits de la llave original. Inicialmente un análisis en un conjunto de llaves es realizado para determinar que posiciones de la llave deben ser usadas para formar las direcciones. Las posiciones de los dígitos que tienen la distribución más uniforme son seleccionadas y posteriormente se invierte el orden de los dígitos para obtener la dirección deseada

El método de intervalo acertado lineal

El espacio llave consiste de enteros en el intervalo (a, d). Este intervalo es dividido en j subintervalos iguales de longitud L, donde L=(d-a)/j. El intervalo de localización para una llave x es encontrado por la formula:

$$r = 1 + \text{trunc}((x-a)/L)$$

los j intervalos pueden ser descritos como:

$I_1 = (a, a+L)$ para r=1

$(a+(r-1)L, a+rL)$ para $2<r<=j$

La ecuación

$$F_i(x) = (G_i * (x - B_i) / (S_i - B_i)) / N$$

de la aproximación lineal de la función de distribución de frecuencia acumulativa para una llave x en el intervalo I_i. Donde B_i y S_i son la frecuencia , la frecuencia acumulativa respectivamente del intervalo I_{i-1} , N es el número de llaves. La función de dispersión requerida para una llave x sobre un intervalo I_i es:

$$H_i(x) = \text{round}(a * F_i(x)) \quad 1 <= i <= t$$

para un espacio de dirección de tamaño a

El método de distribución de frecuencia múltiple

En este método un intervalo acertado lineal inicial estimado de la distribución de frecuencia del espacio llave es usado para mapear dicho espacio en lugar del espacio dirección. Esto tiende a propagar fuera de los grupos de llaves y condensar ligeramente los intervalos poblados. El efecto total es que la distribución de llaves en el espacio llave llega a ser más uniforme. El intervalo acertado lineal estimado de la distribución de las llaves transformadas es encontrado y puede ser usado en conjunción con la primer función del intervalo acertado lineal para formar una función de dispersión. Alternativamente, un número de iteraciones puede ser realizado hasta que la función de dispersión obtenida es juzgada como buena. El criterio usado para determinar el número de iteraciones es obtener las iteraciones cuando la longitud promedio de búsqueda no decrementa por más de una cantidad insignificante (0.2) de una iteración a otra. Una llave del espacio original de llaves es usado para generar una dirección la cual es transformada en cada espacio llave, un intervalo acertado lineal estimado es usado para cada transformación

3.1.3 Métodos de dispersión dinámica

Una de las principales desventajas de los métodos de dispersión convencionales proviene de la organización externa de los archivos en la que el tamaño de la tabla de dispersión debe ser estimado en el avance. Dos problemas pueden surgir. Si el tamaño de la tabla es pequeño la densidad de los accesos llega a ser inaceptablemente cerca cuando la tabla se llena o cuando hay sobreflujo. Si el tamaño de la tabla es grande la utilización del área de sobreflujo resulta inaceptablemente baja. Todas las claves deben ser redispersadas para el tamaño apropiado de la nueva tabla de dispersión y proveer un algoritmo para decidir cuando esta redispersión debe llevarse a cabo. En este algoritmo el número de veces que cada clave es accesada es primero estimado. Considerando la longitud promedio de búsqueda antes y después de una redispersión propuesta, el decremento esperado en cada tiempo de búsqueda, resultante de una redispersión, es determinado. La tabla es completamente redispersada si este decremento en el tiempo de búsqueda es mayor que el costo de la redispersión.

El método de dispersión lineal

En la dispersión lineal la tabla es gradualmente expandida mediante la división de las áreas de almacenamiento direccionable en orden hasta que la tabla ha duplicado su tamaño. Las áreas de almacenamiento direccionable de sobreflujo pueden ser requeridas, pero si la función de dispersión seleccionada para la redispersión es bien escogida, la longitud promedio de búsqueda puede ser reducida significativamente. Una vez que la tabla ha duplicado su tamaño, el mismo método de expansión gradual es aplicado a la tabla agrandada, y así sucesivamente.
El tamaño de la tabla original es denotado por M_0. Después de d duplicaciones el tamaño de la tabla es $2^d M_0$, que es el tamaño actual de la tabla y que se denota por M_p.

La dispersión lineal requiere el uso de una serie de funciones de dispersión H_0, H_1, H_2, ... Originalmente, la función de dispersión H_0 es usada para producir un número entre 0 y H_0-1. Cuando el tamaño de la tabla es duplicado, H_1 es usado para dar un número entre 0 y $2H_0$-1. En general H_d es usada para dar un número entre 0 y $2^d H_0$-1. El subíndice d asociado con H indica el número de veces que ha sido duplicado el tamaño de la tabla original

El método de dispersión extendible

Este método emplea una organización externa para la tabla índice, para representar la tabla índice se usa una estructura extendible en lugar de un árbol binario. La tabla índice contiene 2_d posiciones, donde d es el número de bits más a la izquierda comúnmente usados para direccionar la tabla índice. Los d bits más a la izquierda de una llave, cuando es interpretada como un número, dan la posición en la tabla índice del apuntador al área de almacenamiento direccionable que contiene la llave. Originalmente la tabla contiene solamente una posición, la cual tiene un apuntador a la única área de almacenamiento direccionable en uso. Cuando esta área de almacenamiento direccionable se llena, la tabla índice duplica su tamaño, una nueva área de almacenamiento direccionable es creada y la llaves son ordenadas apropiadamente. Las llaves del área de almacenamiento direccionable de sobreflujo son redistribuidas entre una nueva área de almacenamiento direccionable. Uno de los nuevos apuntadores en la tabla índice debe de apuntar a la que e área de almacenamiento direccionable y el otro al área de almacenamiento direccionable que no tiene actividad. Una vez las llaves apropiadas son añadidas, la tabla índice duplica su tamaño siempre que una distinción basada en el bit d+1 de las llaves es requerida para prevenir un área de almacenamiento direccionable de sobreflujo. Ya que, a menudo las llaves no tienen una distribución uniforme una función de dispersión debe ser aplicada y los resultados resultantes deben ser usados para determinar una posición en la tabla índice, las empleadas subsecuentemente en un área de almacenamiento direccionable. Una seria desventaja para este esquema es que la tabla índice crece demasiado rápidamente para guardarla en memoria principal. Cuando parte o toda la tabla índice es almacenada en memoria secundaria, entonces accesos extra son requeridos para consultar esta tabla

3.2 Diseño del algoritmo de dispersión

Se parte del hecho de que se tiene un conjunto de llaves principales siguiente $Y=\{y \mid y \in [(ficinfo, (fesupy], y \in N\}$ y se tiene un conjunto de direcciones definido por $D=\{d \mid d \in [(fdinfo, (fesupd], d \in N\}$, donde N es el conjunto de los números naturales. se sabe que el número de elementos de D por lo menos debe ser igual al número de elementos de Y, también se sabe que el $(fesupy)-(ficinfo) \leq (fesupd)-(fdinfo)$ que es $(fesupy)-(ficinfo) \leq 10^{numcarlla}-1$ donde numcarlla es el número de caracteres de la llave principal. El objetivo de la función de dispersión f es asociar a cada elemento del conjunto de llaves Y con un único elemento del conjunto de direcciones D, es decir, la función de dispersión f debe ser biunívoca como se muestra en la figura 3.1

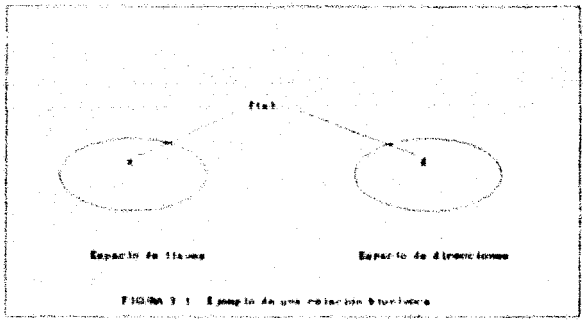

FIGURA 3.1 Ejemplo de una relación biunívoca

Sin embargo, en ocasiones varios elementos del conjunto de llaves Y están asociados con el mismo elemento del conjunto de direcciones D, en este caso se dice que se trata de una función biunívoca, cuando esto sucede se utiliza un método de resolución de colisiones para que los elementos del conjunto de llaves colisionados se puedan asociar con algún elemento del conjunto de direcciones D, dicho método de resolución de colisiones se explicará posteriormente

El primer algoritmo de dispersión que se diseñó fue el de sumas-división que es una combinación del método aditivo con el método de la división los cuales fueron explicados en el punto 3.1.1 de este trabajo.

Este método de suma división consiste en seleccionar la llave principal en un determinado número de partes donde cada parte tiene la misma longitud requerida para la dirección. dichas partes son sumadas. se ignora el acarreo final. el resultado de la suma se divide entre la frecuencia de. intervalo al que pertenece la llave principal y finalmente el residuo de dicha división se suma a la dirección mínima del intervalo. Como a la dirección mínima se le suma el residuo de la división siempre se obtienen direcciones que se encuentren entre la dirección mínima y la dirección máxima ya que dicho residuo no puede ser mayor que la frecuencia y. como se sabe la dirección máxima es igual a la dirección mínima más la frecuencia.

Posteriormente se pensó en diseñar una función de dispersión basada en siguiente gráfica y tomando en cuenta los parámetros de control del conjunto de llaves y del conjunto de direcciones I para cada intervalo (figura 3.2).

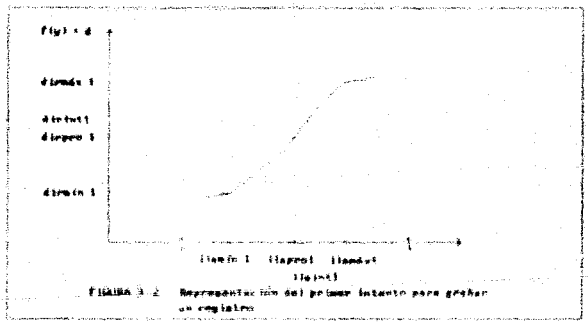

Figura 3.2 Representación del primer intento para grabar un registro

El segundo algoritmo de dispersión diseñado es una interpolación de Lagrange de primer orden mejor conocida como interpolación lineal la cual consiste en que para un intervalo i de llaves $[llamín_i, llamáx_i]$ asociado con un intervalo i de direcciones $[dirmín_i, dirmáx_i]$, a una llave intermedia $llaint_i$ se le asocia una dirección intermedia $dirint_i$ mediante la siguiente fórmula

$$dirint_i = dirmín_i + (dirmáx_i - dirmín_i)(llaint_i - llamín_i)/(llamáx_i - llamín_i)$$

En tercer lugar se diseñó una función de dispersión basada en una interpolación de Lagrange de segundo orden aprovechando que para cada intervalo de clases o fracción se conoce su clave promedio $llapro_i$ la cual está asociada con la dirección promedio $dirpro_i$, la fórmula para este caso es la siguiente:

$$dirint_i = dirafn_i * clafn_i + dirpro_i * clpro_i + dirmax_i * clmax_i$$

donde

$$clafn_i = (llaint_i - llapro_i)*(llaint_i - llamax_i) / (llamin_i - llapro_i)*(llamin_i - llamax_i)$$

$$clpro_i = (llaint_i - llamin_i)*(llaint_i - llamax_i) / (llapro_i - llamin_i)*(llapro_i - llamax_i)$$

$$clmax_i = (llaint_i - llamin_i)*(llaint_i - llapro_i) / (llamax - llamin_i)*(llamax_i - llapro_i)$$

3.3 Descripción de los métodos de resolución de colisiones tradicionales

Una función de dispersión puede mapear algunas llaves a la misma dirección, cuando esta situación aparece los registros colisionados deben ser clasificados y accesados mediante un método de resolución de colisiones. Dichos métodos se dividen en dos clases generales: direccionamiento abierto y encadenamiento. En general, el objetivo de un método de resolución de colisiones es procurar un lugar para los registros colisionados en la tabla. Esto requiere la investigación de una serie de localidades en la tabla hasta que una localidad vacía es encontrada para acomodar el registro colisionado. Se requiere un mecanismo para generar la serie de localidades en la tabla ha ser examinadas. Los principales criterios para este mecanismo es la rapidez, seguridad y reproductividad.

3.3.1 Métodos de direccionamiento abierto

Si un registro con una llave x es mapeado a una localidad con dirección d y esta localidad está ocupada, entonces otras localidades de la tabla son examinadas hasta encontrar una vacía para el nuevo registro. La secuencia en la que las localidades de la tabla son examinadas pueden ser formuladas por diferentes caminos.

El método de prueba lineal

Uno de los métodos más simples para resolver colisiones es usar la siguiente secuencia de posiciones para una tabla de n entradas: d, d+1, ..., n-1, 0, 1, 2, ..., d-1. Una localidad de registro es siempre encontrada si al menos hay una disponible. De otra forma, la búsqueda se detiene insatisfactoriamente después de examinar n localidades. Cuando se quiere recuperar un registro en particular, la misma secuencia de localidades es examinada hasta que el registro es encontrado o hasta que una localidad de registro vacía es encontrada. En este último caso el registro deseado no está en la tabla.

El método de prueba aleatoria

Este método genera una secuencia aleatoria de localidades más apropiada que en el caso de la prueba lineal. La secuencia aleatoria generada en esta fase debe contener localidades entre uno y n exactamente. Una tabla se llenada cuando la primer posición duplicada es generada.

3.3.2 Métodos de encadenamiento

Hay tres principales dificultades con el método de direccionamiento abierto. Primero, las listas de registros colisionados por diferentes valores de dispersión se llegan a entremezclar, este fenómeno requiere más pruebas. Segundo, no se puede manejar una tabla en situaciones de sobreflujo de una manera satisfactoria, al detectar un

sobreflujo la tabla completa debe ser reorganizada. el problema del sobreflujo no puede ser ignorado ya que el requerimiento de espacio puede crecer drásticamente. Tercero, el borrado físico de registros es difícil. Para resolver estos problemas se usan métodos de localidades ligadas

El método de encadenamiento separado

Es el más popular de los métodos de manejo de registros de sobreflujo. En este método los registros de sobreflujo son encadenados a un área especial de sobreflujo que es distinta del área primaria. Esta área contiene esa parte de la tabla en la cual los registros son inicialmente dispersados. Una lista ligada separada se mantiene por cada conjunto de registros ligados. Por lo tanto, un campo apuntador es requerido por cada registro en las áreas primarias y de sobreflujo. Un uso más eficiente de este método involucra el uso de una tabla intermedia o de una tabla de dispersión. Con este método todos los registros residen en el área de sobreflujo mientras que el área primaria contiene solamente apuntadores. Para una tabla con registros largos implica un área de sobreflujo densamente empaquetada, aunque mientras las entradas de la tabla de dispersión sean apuntadores los registros pueden ser largos sin desperdiciar mucho almacenamiento.

El número promedio de comparaciones, asumiendo una función de dispersión distribuida uniformemente, para accesar una entrada particular es ligeramente mayor que uno. Esto demuestra que el tiempo de búsqueda es independiente del número de entradas.

3.4 Diseño del algoritmo de resolución de colisiones

El algoritmo de resolución de colisiones que se utiliza es una combinación del método de direccionamiento abierto con el método de encadenamiento separado.

Si a una llave principal y la función de dispersión f le asocia una dirección d cuyo registro ya está ocupado, para resolver la colisión se examinan las siguientes direcciones en el orden en que se muestran: d+1, d-1, d+2, d-2.

ver figura 3.5

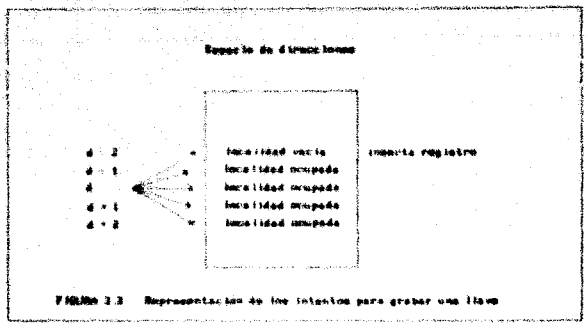

FIGURA 3.3 Representación de los intentos para grabar una llave

Si no se encontró una dirección con un registro desocupado la llave principal y es rediispersada a un área de sobreflujo que pertenece al archivo directo principal exclusivamente mediante una interpolación lineal la cual le asocia una dirección d .

d = lfsinfa+((lfsupa-lfsinfa)(y-lfsinfy)/(lfsupy-lfsinfy)

donde:

lfsupa es el límite superior del área de sobreflujo

lfsinfa es el límite inferior del área de sobreflujo

lfsupy es el límite superior del intervalo de llaves

lfsinfy es el límite inferior del intervalo de llaves

Si el registro que ocupa la dirección d ,a está ocupado se prueban las direcciones siguientes en el orden en que se muestran: d+1, d-1, d+2, d-2 , ver figura 3.4.

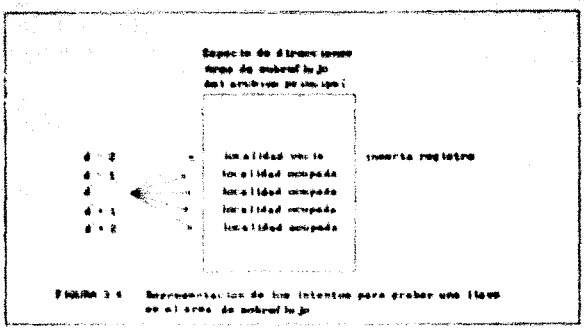

FIGURA 3.4 Representación de un intento para grabar una llave en el área de sobreflujo

Si nuevamente no se localiza una dirección con un registro desocupado la llave principal y es otra vez dispersada mediante una interpolación lineal pero ahora a una área especial de sobreflujo, la cual da servicio a más de un archivo directo principal, para obtener una dirección d''

$$d'' = (\text{ainfse}*(\text{limsupse}-\text{liminfse}))/((\text{limsupy}-\text{liminfy}))$$

donde:

limsupse es el límite superior del área de sobreflujo especial

liminfse es el límite inferior del área de sobreflujo especial

Si el registro con dirección d'' ya está ocupado se prueban las direcciones siguientes en el orden en que se muestran: d''+1, d''-1, d''+2, d''-2 , ver figura 3.5.

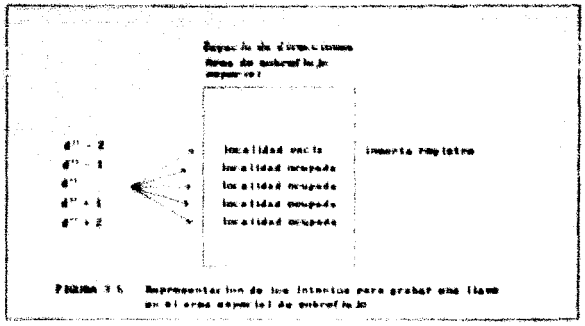

FIGURA 3.5 Representación de los intentos para grabar una llave en el area especial de sobreflujo

En el peor de los casos si no se ha encontrado una dirección desocupada hasta este nivel el registro con llave principal y se manda a un archivo secuencial alterno, esta es una posibilidad muy poco probable ya que para llegar a esta decisión se tienen que haber examinado 15 registros: 5 registros del archivo directo principal, 5 registros del área de sobreflujo del archivo directo principal y 5 registros del área de sobreflujo especial.

la representación esquemática de la localización de los archivos directos generados se muestra a continuación:

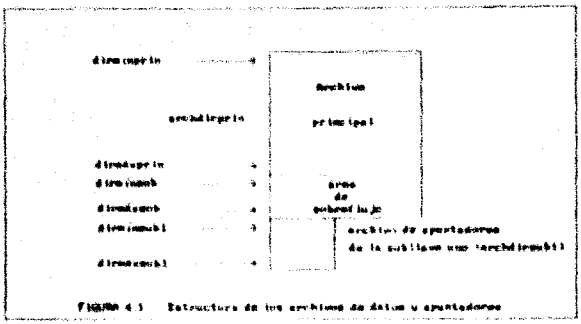

FIGURA 4.1 Estructura de los archivos de datos y apuntadores

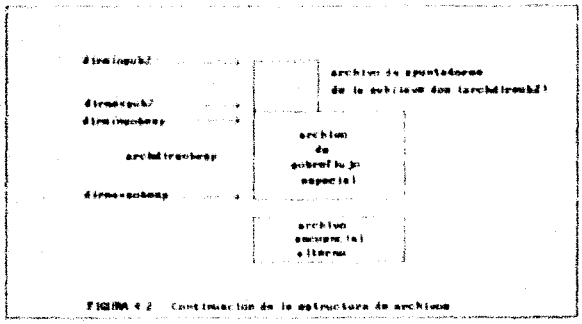

FIGURA 4.2 Continuación de la estructura de archivos

Nomenclatura

archsecent	archivo secuencial de entrada
archsecalt	archivo secuencial alterno
archdirprin	archivo directo principal
archdirsub1	archivo directo de apuntadores para la llave secundaria uno
archdirsub2	archivo directo de apuntadores para la llave secundaria dos
archdirsobesp	archivo directo de sobreflujo especial
totregent	total de registros en el archivo secuencial de entrada
numcarlla	número de caracteres de la llave principal
dirminprin	dirección mínima del archivo directo principal
dirmáxprin	dirección máxima del archivo directo principal
dirminsob	dirección mínima del área de sobreflujo del archivo directo principal
dirmáxsob	dirección máxima del área de sobreflujo del archivo directo principal
dirminsub1	dirección mínima del archivo directo de apuntadores para la llave secundaria uno
dirmáxsub1	dirección máxima del archivo directo de apuntadores para la llave secundaria uno
dirminsub2	dirección mínima del archivo directo de apuntadores para la llave secundaria dos
dirmáxsub2	dirección máxima del archivo directo de apuntadores para la llave secundaria dos
dirminsobesp	dirección mínima del archivo directo de sobreflujo especial
dirmáxsobesp	dirección máxima del archivo directo de sobreflujo especial
llavein	llave mínima del conjunto de llaves principales
llavemáx	llave máxima del conjunto de llaves principales
totint	total de intervalos del espacio de llaves

4.5 Desarrollo del manejador de archivos

Este manejador de archivos se realizó teniendo en cuenta que los datos de entrada se encuentran en un archivo secuencial por ejemplo en una cinta, pudiendo o no estar ordenados y que tienen que ser dispersados en un archivo directo de acuerdo a su llave principal para un acceso más rápido. Todos los archivos directos que se explican son direccionables por registro lo cual implica el uso de una tabla donde se tiene la longitud del registro para cada archivo directo.

En este caso, el manejador de archivos de acceso directo consta de cuatro partes:

1) Programa generador de datos - en forma aleatoria genera los registros del archivo de entrada cuya estructura es de una llave principal, dos llaves secundarias y un campo de información de un caracter sólo para ejemplificar que se puede tener almacenada más información.

2) Programa de dispersión de archivos - tiene el objetivo de dispersar cada uno de los registros de entrada por su llave principal en el archivo directo principal de acuerdo a una de las tres funciones de dispersión que se explicaron en el capítulo 3.2, también se encarga de la resolución de colisiones explicada en el capítulo 3.4 y por último, genera el archivo directo de apuntadores por cada subllave en las dos direcciones de los encadenamientos.

3) Programa de acceso de archivo - accesa todos los registros del archivo directo principal en el orden en que fueron dispersados en base a la llave principal, también accesa dicho archivo por la llave secundaria uno, en orden ascendente y por la llave secundaria dos, en el mismo orden.

4) Programa de acceso por registro - accesa un registro del archivo directo principal por su llave principal o accesa un conjunto de registros de dicho archivo por una de las dos llaves secundarias.

Descripción del algoritmo de dispersión

Paso uno. Obtener totint = $10^{numcarllv - 3}$ y dividir el intervalo de claves primarias en totint intervalos iguales, por ejemplo, si la llave principal es de 4 caracteres habrá 10 intervalos que son:

0-999, 1000-1999, 2000-2999, 3000-3999, 4000-4999, 5000-5999, 6000-6999, 7000-7999, 8000-8999, 9000-9999

Paso dos. obtener dirmáxprin, dirminsob, dirmáxsob, dirminsub1, dirmáxsub1, dirminsub2, dirmáxsub2, dirminsobesp y dirmáxsobesp mediante

dirmáxprin = dirminprin + totregent - 1

dirminsob = dirmáxprin + 1

dirmáxsob = dirminsob + totregent/10 - 1

dirminsub1 = dirmáxsob + 1

dirmáxsub1 = dirminsub1 + totregent - 1

dirminsub2 = dirmáxsub1 + 1

dirmáxsub2 = dirminsub2 + totregent - 1

dirminsobesp = dirmáxsub2 + 1

dirmáxsobesp = dirminsobesp + totregent - 1

Es decir el tamaño del área de sobreflujo del archivo directo principal es igual al 10 porciento del tamaño de dicho archivo.

Paso tres: leer en forma secuencial los intregistros del archivo y para cada intervalo de llaves principales obtener la llave mínima ($llamin_i$), la llave máxima ($llamax_i$), la frecuencia de llaves ($frella_i$) y la llave promedio ($llapro_i$), en donde

$$llapro_i = \sum_{i=1}^{frella_i} llave_i / frella_i$$

Paso cuatro: por cada intervalo de llaves principales y asignar a la llave mínima y máxima su dirección mínima y máxima respectivamente, mediante la suma de las frecuencias acumulativas, es decir

$$dirmin_1 = dirregprin$$

$$dirmax_1 = dirmin_1 + frella_1$$

$$dirmin_i = dirmax_{i-1} + 1$$

$$dirmax_i = dirmin_i + frella_i$$

para $i = 1, 2, \ldots, totint$

De tal forma que $dirmax_{totint} = diregprin$

Paso cinco: obtener la dirección ($dirint_i$) para una llave principal ($llaint_i$) que se encuentre en el intervalo i mediante una de las tres funciones de dispersión diseñadas: suma de bits, interpolación lineal o interpolación de Lagrange de segundo orden. La función de dispersión seleccionada se utilizará también para el acceso de registros.

Paso seis: si el registro que se encuentra en la dirección $dirint_i$ ya está ocupado, entonces revizar los registros que se encuentran en las siguientes direcciones: $dirint_i+1$, $dirint_i-1$, $dirint_i+2$, $dirint_i-2$

Paso 4 ete: si no se encontró un registro desocupado, entonces a la llave principal ($llaint_i$) se le asignará una dirección (dirsob) del área de sobreflujo del archivo directo principal mediante una interpolación lineal.

$$dirsob = dirsinsob + (dirmaxsob - dirsinsob)*(llaint_i - llasin)/(llamax - llasin)$$

Paso ocho: si el registro que se encuentra en la dirección dirsob ya está ocupado, entonces revisar los registros que se encuentran en las siguientes direcciones: dirsob+1, dirsob-1, dirsob+2, dirsob-2.

Paso nueve: si no se encontró un registro desocupado, entonces a la llave principal ($llaint_i$) se le asignará una dirección (dirsobesp) del archivo directo de sobreflujo especial mediante una interpolación lineal.

$$dirsobesp = dirsinsobesp + (dirmaxsobesp - dirsinsobesp)*(llaint_i - llasin)/(llamax - llasin)$$

Paso diez: si el registro que se encuentra en la dirección dirsobesp ya está ocupado, entonces revisar los registros que se encuentran en las siguientes direcciones: dirsobesp+1, dirsobesp-1, dirsobesp+2, dirsobesp-2.

Paso once: si no se ha podido encontrar una dirección con un registro desocupado entonces grabar el registro con llave principal ($llaint_i$) en el archivo secuencial externo archsecalt.

Paso doce: grabar las direcciones correspondientes en el archivo directo de apuntadores para la llave secundaria uno.

Paso trece: actualizar el campo de dirección siguiente del registro anterior de la cadena en el archivo directo de apuntadores para la llave secundaria uno. También se actualiza el campo de dirección final del registro de control para la llave secundaria uno.

Paso catorce: grabar las direcciones correspondientes en el archivo directo de apuntadores para la llave secundaria dos.

Paso quince: actualizar el campo de dirección siguiente del registro anterior de la cadena en el archivo directo de apuntadores para la llave secundaria dos. También se actualiza el campo de dirección final del registro de control para la llave secundaria dos.

Paso dieciseis: repetir los pasos del cinco al quince hasta que se terminen de dispersar los totregent registros del archent.

Descripción del algoritmo de acceso

Paso uno: obtener la dirección $(dir(int_i))$ para una llave primaria $(llaint_i)$ que se encuentre en el intervalo i mediante una de las tres funciones de dispersión diseñadas: suma-división, interpolación lineal o interpolación de Lagrange de segundo orden. La función de acceso seleccionada debe ser la misma que se utilizó para la dispersión de registros.

Paso dos: si el registro que se encuentra en la dirección $dirint_1$ no es el que se busca, entonces revizar los registros que se encuentran en las siguientes direcciones: $dirint_1+1$, $dirint_1-1$, $dirint_1+2$, $dirint_1-2$

Paso tres: si no se encontró el registro deseado, entonces a la llave principal ($llaint_1$) se le asignará una dirección (dirsob) del área de sobreflujo del archivo directo principal mediante una interpolación lineal:

$$dirsob = dirsobsob + (dirsobxsob-dirsobsob)*(llaint_1-llasob)/(llasobx-llasob)$$

Paso cuatro: si el registro que se encuentra en la dirección dirsob no es el que se busca, entonces revizar los registros que se encuentran en las siguientes direcciones: dirsob+1, dirsob-1, dirsob+2, dirsob-2

Paso cinco: si no se encontró el registro deseado, entonces a la llave principal ($llaint_1$) se le asignará una dirección (dirsobesp) del archivo directo de sobreflujo especial mediante una interpolación lineal:

$$dirsobesp = dirsobsobesp + (dirsobxsobesp-dirsobsobesp)*(llaint_1-llasob)/(llasobx-llasob)$$

Paso seis: si el registro que se encuentra en la dirección dirsobesp no es el que se busca, entonces revizar los registros que se encuentran en las siguientes direcciones: dirsobesp+1, dirsobesp-1, dirsobesp+2, dirsobesp-2

Paso siete: si no se ha podido encontrar la dirección con el registro deseado entonces buscar el registro con llave principal $llaint_1$ en el archivo secuencial alterno archsecalt

Paso ocho: repetir los pasos del uno al siete hasta que se terminen de localizar los totregent registros del archent.

Paso nueve: hacer el acceso para la llave secundaria uno siguiendo el encadenamiento que se encuentra en el archivo directo de apuntadores generado durante la dispersión. Este acceso por encadenamiento utiliza dos accesos: un primer acceso al archivo directo de apuntadores de la llave secundaria uno donde se obtiene la dirección del archivo directo principal en la que se encuentra el registro con dicha llave secundaria y la dirección del siguiente registro en la cadena. El segundo acceso se hace al archivo directo principal.

Paso diez: repetir el paso nueve hasta que se terminen de recorrer las cadenas generadas para la llave secundaria uno.

Paso once: hacer el acceso para la llave secundaria dos siguiendo el encadenamiento que se encuentra en el archivo directo de apuntadores generado durante la dispersión. Este acceso por encadenamiento utiliza dos accesos: un primer acceso al archivo directo de apuntadores de la llave secundaria dos donde se obtiene la dirección del archivo directo principal en la que se encuentra el registro con dicha llave secundaria y la dirección del siguiente registro en la cadena. El segundo acceso se hace al archivo directo principal.

Paso doce: repetir el paso once hasta que se terminen de recorrer las cadenas generadas para la llave secundaria dos.

4.2 DEMOSTRACION DE RESULTADOS

Los resultados generados se muestran a partir de las páginas siguientes en un total de 21 tablas, incluyendo los mensajes de error y las estadísticas obtenidas tanto para la dispersión de registros como para el acceso de los registros. Estos resultados se obtuvieron a partir de un archivo de 20 registros generados en forma aleatoria. Un registro se encuentra disponible cuando el campo de la llave principal tiene !!!!!.

Se explica el método 2 (interpolación de Lagrange) porque es con el que se obtuvieron mejores resultados.

El orden en que se muestran los resultados es de acuerdo a como se fueron generando.

Tabla - 1 - archivo secuencial de datos de entrada

Tabla - 2 - parámetros de control del archivo secuencial de datos de entrada

Tabla - 3 - rastreo de dispersión de los registros por su llave principal

Tabla 4.6 - rastreo del encadenamiento de los registros por su llave secundaria uno.

Tabla 4.5 - rastreo del encadenamiento de los registros por su llave secundaria dos.

Tabla 4.6 - parámetros de control del archivo directo de apuntadores con las direcciones de los encadenamientos para la llave secundaria uno.

Tabla 4.7 - parámetros de control del archivo directo de apuntadores con las direcciones de los encadenamientos para la llave secundaria dos.

Tabla 4.8 - archivo directo principal.

Tabla 4.9 - archivo directo de sobreflujo esquema.

Tabla 4.10 - archivo directo de apuntadores con las direcciones de los encadenamientos para la llave secundaria uno.

Tabla 4.11 - archivo directo de apuntadores con las direcciones de los encadenamientos para la llave secundaria dos.

Tabla 4.12 - rastreo del acceso completo del archivo directo principal por su llave principal.

Tabla 4.13 - rastreo del acceso completo del archivo directo principal por su llave secundaria uno.

Tabla 4.14 - rastreo del acceso completo del archivo directo principal por su llave secundaria dos.

Tablas 4.15 a 4.17 - rastreo del acceso interactivo con el usuario del archivo directo principal para cuatro llaves principales proporcionadas.

Tablas 4.18 a 4.19 - rastreo del acceso interactivo con el usuario del archivo directo principal para cuatro llaves secundarias del tipo uno proporcionadas.

Tablas 4.20 a 4.21 - rastreo del acceso interactivo con el usuario del archivo directo principal para cuatro llaves secundarias del tipo dos proporcionadas.

Finalmente, se muestran los mensajes emitidos por el manejador de archivos durante la corrida interactiva.

TABLA 4.1

EJEMPLO PARTICULAR CON UNA CLAVE PRINCIPAL Y

DOS LLAVES SECUNDARIAS

ARCHIVO DE DATOS DE ENTRADA

NUMERO REGISTRO	LLAVE PRINCIPAL	SUBLLAVE DOS	SUBLLAVE TRES	OTRO DATO
1	198	8	5	H
2	7624	33	8	H
3	2799	8	0	H
4	0221	24	6	H
5	8218	82	4	H
6	2620	35	0	H
7	870	67	8	H
8	4650	92	0	H
9	3414	20	1	H
10	5851	20	7	H
11	3670	84	2	H
12	7783	74	3	H
13	4837	13	3	H
14	469	97	5	H
15	313	30	3	H
16	6682	24	1	H
17	1378	42	7	H
18	2135	28	6	H
19	7239	76	4	H
20	1034	51	1	H

****

CLAVES PRINCIPALES REPETIDAS EN EL ARCHIVO DE ENTRADA ARCHIVO

****

CLAVE	DIRECCION	REPETICION

****

****

NUMERO DE REGISTROS REPETIDOS 0

****

TABLA A - 2

PARAMETROS DE CONTROL DEL ARCHIVO DIRECTORIO PRINCIPAL

INTER- VALO	LLAVE MINIMA	LLAVE MAXIMA	LLAVE PROMEDIO	FRECUEN- CIA	DIREC MINIMA	DIREC MAXIMA	DIRECCION PROMEDIO
1	148	870	4.625E+02	4	2001	2004	2.002E+03
2	1034	1378	1.206E+03	2	2005	2006	2.005E+03
3	2135	2795	2.517E+03	3	2007	2009	2.008E+03
4	3414	3630	3.542E+03	2	2010	2011	2.010E+03
5	4650	4837	4.745E+03	2	2012	2013	2.012E+03
6	5851	5851	5.851E+03	1	2014	2014	2.014E+03
7	6221	6682	6.451E+03	2	2015	2016	2.015E+03
8	7239	7783	7.509E+03	3	2017	2019	2.018E+03
9	8218	8218	8.218E+03	1	2020	2020	2.020E+03
10	11111	0	0.000E+00	0	0	0	0.000E+00

TABLA 3

MAPEO DE LA DISPERSION DE LA CLAVE PRINCIPAL

NÚ	CLAVE	ACCESO		EN	DIRECCION		CAMPO	TIPO
MIS	PRIN	NU	TI	TER	CALCU	DISPER	CLAVE	SOBRE
TRO	CIPAL	ME	RO	VALO	LADA	SADA	PRINCI	CLAVE
1	198	1	L	1	2001	2001	11111	
1	198	2	E	1	2001	2001	198	
2	7624	3	L	8	2018	2018	11111	
2	7624	4	E	8	2018	2018	7624	
3	2795	5	L	3	2009	2009	11111	
3	2795	6	E	3	2009	2009	2795	
4	6221	7	L	7	2015	2015	11111	
4	6221	8	E	7	2015	2015	6221	
5	8218	9	L	9	2020	2020	11111	
5	8218	10	E	9	2020	2020	8218	
6	2620	11	L	3	2008	2008	11111	
6	2620	12	E	3	2008	2008	2620	
7	870	13	L	1	2004	2004	11111	
7	870	14	E	1	2004	2004	870	
8	4650	15	L	5	2012	2012	11111	
8	4650	16	E	5	2012	2012	4650	
9	3414	17	L	4	2010	2010	11111	
9	3414	18	E	4	2010	2010	3414	
10	5851	19	L	6	2014	2014	11111	
10	5851	20	E	6	2014	2014	5851	
11	3670	21	L	4	2011	2011	11111	

TABLA 6.3

RASTREO DE LA DISPERSION DE LA LLAVE PRINCIPAL

RE	LLAVE	ACCESO		IN	DIRECCION		CAMPO	TIPO
GIS	PRIN	NU	TI	SER	CALCU	DISPER	LLAVE	SOBRE
TRO	(PRN.	RE	PO	VALO	LADA	SADA	PRINCI	FLUJO
11	3670	22	E	4	2011	2011	3670	
12	7783	23	L	8	2019	2019	11111	
12	7783	24	E	8	2019	2019	7783	
13	4837	25	L	5	2013	2013	11111	
13	4837	26	E	5	2013	2013	4837	
14	669	27	L	1	2002	2002	11111	
14	669	28	E	1	2002	2002	669	
15	313	29	L	1	2002	2002	669	
15	313	30	L	1	2002	2003	11111	
15	313	31	E	1	2002	2003	313	
16	6682	32	L	7	2016	2016	11111	
16	6682	33	E	7	2016	2016	6682	
17	1378	34	L	2	2006	2006	11111	
17	1378	35	E	2	2006	2006	1378	
18	2135	36	L	3	2007	2007	11111	
18	2135	37	E	3	2007	2007	2135	
19	7239	38	L	8	2017	2017	11111	
19	7239	39	E	8	2017	2017	7239	
20	1034	40	L	2	2005	2005	11111	
20	1034	41	E	2	2005	2005	1034	

TABLA 4.4

RASTREO DEL ENCADENAMIENTO DE DISPERSIÓN DE LAS

DIRECCIONES DE LA CLAVE SECUNDARIA UNO

REGISTRO	SUBCLAVE UNO	ACCESO NO.	ACCESO TIPO	PARAMETROS DE CONTROL DIR INI	PARAMETROS DE CONTROL DIR FIN	DIRECCIONES EN EL ARCHIVO DE APUNTADORES DIR ANT	DIR ACT	DIR SIG	DIR. DEL ARCH. PRIN.
1	8	1	E	2023	2023	2023	2023	2023	2001
2	33	2	E	2024	2024	2024	2024	2024	2018
3	8	3	E	2023	2023	2023	2023	2025	2001
5	8	4	E	2023	2025	2023	2025	2025	2009
4	24	5	E	2026	2026	2024	2026	2026	2018
5	82	6	E	2027	2027	2027	2027	2027	2020
6	35	7	E	2028	2028	2028	2028	2028	2008
7	67	8	E	2029	2029	2029	2029	2029	2004
8	92	9	E	2030	2030	2030	2030	2030	2012
9	20	10	E	2031	2031	2031	2031	2031	2010
10	20	11	E	2031	2031	2031	2031	2032	2010
10	20	12	E	2031	2032	2031	2032	2032	2014
11	84	13	E	2033	2033	2033	2033	2033	2011
12	74	14	E	2034	2034	2034	2034	2034	2019
13	13	15	E	2035	2035	2035	2035	2035	2013
14	97	16	E	2036	2036	2036	2036	2036	2002
15	30	17	E	2037	2037	2037	2037	2037	2003
16	24	18	E	2026	2026	2026	2026	2038	2015
16	24	19	E	2026	2038	2026	2038	2038	2016

TABLA 4.5

GASTNED DEL ENCADENAMIENTO DE DISPERSION DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA UNO

NE	SUB	ALCESO		Primary keys		DIRECCIONES EN EL			DIA
GIS	LLA	NU	TI	DE CONTROL		ARCHIVO DE APUNTADORES			DEL
TRO	VE	RE	PO	DIR	DIR	DIR	DIR	DIR	ARCH.
	UNO	RO		INI	FIN	ANT	ACT	SIG	PRIN.
17	42	20	E	2039	2039	2039	2039	2039	2006
18	28	21	E	2040	2040	2040	2040	2040	2007
19	76	22	E	2041	2041	2041	2041	2041	2017
20	51	23	E	2042	2042	2042	2042	2042	2005

TABLA A.5

MAPEO DEL ENCAMINAMIENTO DE DISPERSION DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA DNX

DF	SUB	ACCESO		PARAMETROS		DIRECCIONES EN EL			DIR
BIS	LIA	NO	TI	DE CONTROL		ARCHIVO DE APUNTADORES			DEL
TRO	VE	ME	PO	DIR	DIR	DIR	DIR	DIR	ARCH
	DOS	RO		INI	FIN	ANT	ACT	SIG	PRIN
1	5	1	E	2043	2043	2043	2043	2043	2001
2	8	2	E	2044	2044	2044	2044	2044	2018
3	0	3	E	2045	2045	2045	2045	2045	2009
4	4	4	E	2046	2046	2046	2046	2046	2015
5	4	5	E	2046	2046	2046	2046	2047	2015
5	4	6	E	2046	2047	2046	2047	2047	2020
6	0	7	E	2045	2045	2045	2045	2048	2009
6	0	8	E	2045	2048	2045	2048	2048	2008
7	8	9	E	2044	2044	2044	2044	2049	2018
7	8	10	E	2044	2049	2044	2049	2049	2006
8	0	11	E	2045	2048	2045	2048	2050	2008
8	0	12	E	2045	2050	2050	2050	2050	2012
9	1	13	F	2051	2051	2051	2051	2051	2010
10	7	14	E	2052	2052	2052	2052	2052	2014
11	2	15	F	2053	2053	2053	2053	2053	2011
12	3	16	E	2054	2054	2054	2054	2054	2019
13	3	17	E	2054	2054	2054	2054	2055	2019
13	5	18	E	2054	2055	2056	2055	2055	2013
14	5	19	E	2043	2043	2043	2043	2056	2001

TABLA N° 5

MAESTRO DEL ENTRELAZAMIENTO DE DISPERSION DE LAS

DIRECCIONES DE LA CLAVE SECUNDARIA DOS

RE	SUB	ACCESO		TABLA LINKS		DIRECCIONES EN EL			DIR
GIS	LLA	NU	TI	DE CONTROL		ARCHIVO DE APUNTADORES			DEL
TRO	VE	NE	RO	DIR.	DIR	DIR	DIR	DIR.	ARCH
	DOS	RO		INI	FIN	ANT	ACT	SIG	PRIN
14	5	20	E	2043	2056	2043	2056	2056	2002
15	3	21	E	2054	2045	2054	2055	2057	2013
15	3	22	E	2054	2057	2055	2057	2057	2003
16	1	23	E	2051	2051	2051	2051	2058	2010
16	1	24	E	2051	2058	2051	2058	2058	2016
17	7	25	E	2052	2052	2052	2052	2059	2016
17	7	26	E	2052	2059	2052	2059	2059	2006
18	6	27	E	2060	2060	2060	2060	2060	2007
19	4	28	E	2046	2047	2046	2047	2061	2020
19	4	29	E	2046	2061	2047	2061	2061	2017
20	1	30	E	2051	2058	2051	2058	2062	2016
20	1	31	E	2051	2062	2058	2062	2062	2005

****

ERREURS DÉTECTÉES DURANTE LA DISPERSION DE REGISTRES

****

****

1 Método de dispersión por interpolación lineal

2 Método de dispersión por multiplicación de ingreso

3 Método de dispersión por suma y por división

****

¿Qué método de dispersión desea ? 2

****

DISPERSIÓN EN ARCHIVO POR INTERPOLACIÓN DE CADENAS

DISPERSIÓN EN SOBREFLUJO POR INTERPOLACIÓN LINEAL

****

****

ESTADÍSTICAS DE LA DISPERSIÓN DE REGISTROS

****

REGISTROS LEÍDOS 20

REGISTROS DISPERSADOS 20

TOTAL COLISIONES 9

NÚMERO DE ACCESOS 47

NÚMERO DE ACCESOS PROMEDIO 2.350000

****

ESTADISTICAS DE LA DISPERSION DE REGISTROS

****

DISPERSADOS EN AREA NORMAL 20

EN PRIMER INTENTO 19

EN SEGUNDO INTENTO 1

EN TERCER INTENTO 0

EN CUARTO INTENTO 0

EN QUINTO INTENTO 0

DISPERSADOS EN AREA DE SOBREFLUJO 0

EN SEXTO INTENTO 0

EN SEPTIMO INTENTO 0

EN OCTAVO INTENTO 0

EN NOVENO INTENTO 0

EN DECIMO INTENTO 0

DISPERSADOS EN AREA ESPECIAL DE SOBREFLUJO 0

EN ONCEAVO INTENTO 0

EN DOCEAVO INTENTO 0

EN TRECEAVO INTENTO 0

EN CATORCEAVO INTENTO 0

EN QUINCEAVO INTENTO 0

```
**********************************************************

       ESTADISTICAS DE LA DISPERSION DE REGISTROS

**********************************************************

COLISIONES EN AREA NORMAL      1

        EN PRIMER INTENTO      1

       EN SEGUNDO INTENTO      0

        EN TERCER INTENTO      0

        EN CUARTO INTENTO      0

        EN QUINTO INTENTO      0

COLISIONES EN AREA DE SOBREFLUJO      0

                EN SEXTO INTENTO      0

              EN SEPTIMO INTENTO      0

               EN OCTAVO INTENTO      0

               EN NOVENO INTENTO      0

               EN DECIMO INTENTO      0

COLISIONES EN AREA ESPECIAL DE SOBREFLUJO      0

                      EN ONCEAVO INTENTO      0

                      EN DOCEAVO INTENTO      0

                     EN TRECEAVO INTENTO      0

                   EN CATORCEAVO INTENTO      0

                    EN QUINCEAVO INTENTO      0


**********************************************************

   ESO ES TODO EN CUANTO A DISPERSION DE REGISTROS

**********************************************************
```

TABLA n 6

PARAMETROS DE CONTROL DEL ARCHIVO DIRECTO DE APUNTADORES DE LA SUBLLAVE UNO

SUBLLAVE	DIRECCION INICIAL	DIRECCION FINAL
8	2025	2025
13	2035	2035
20	2031	2032
24	2026	2038
28	2040	2040
30	2037	2037
33	2024	2024
35	2028	2028
42	2039	2039
51	2042	2042
67	2029	2029
71	2034	2034
76	2041	2041
82	2027	2027
84	2033	2033
92	2030	2030
97	2036	2036

Tabla 7

PARÁMETROS DE CONFIGURACIÓN DE ARCHIVO DIRECTO DE APLICACIONES DE LA SUBCLASE 205

SUBCLASE	DIRECCIÓN INICIAL	DIRECCIÓN FINAL
0	2049	2050
1	2051	2052
2	2053	2053
3	2054	2057
4	2058	2061
5	2062	2056
6	2060	2060
7	2062	2064
8	2064	2069
9	11111	11111

TABLA 5.8

ARCHIVO DIRECTO PRINCIPAL

DIRECCION	CLAVE PRINCIPAL	SUBCLAVE UNO	SUBCLAVE DOS	DATO
2001	198	8	5	H
2002	469	97	5	H
2003	313	50	3	H
2004	870	67	8	H
2005	1034	51	3	H
2006	1378	42	7	H
2007	2133	28	6	H
2008	2670	11	0	H
2009	2795	8	0	H
2010	3414	20	3	H
2011	3670	84	2	H
2012	4850	92	0	H
2013	4837	13	3	H
2014	5851	20	2	H
2015	6221	24	4	H
2016	6682	24	3	H
2017	7232	28	4	H
2018	7664	33	8	H
2019	7781	74	3	H
2020	8219	82	4	H
2021	11111	11111	11111	
2022	11111	11111	11111	

TABLA 4.9

ARCHIVO DIRECTO PRINCIPAL

DIRECCION	LLAVE PRINCIPAL	SUBLLAVE UNO	SUBLLAVE DOS	OTRO DATO
2063	11111	11111	11111	
2064	11111	11111	11111	
2065	11111	11111	11111	
2066	11111	11111	11111	
2067	11111	11111	11111	
2068	11111	11111	11111	
2069	11111	11111	11111	
2070	11111	11111	11111	
2071	11111	11111	11111	
2072	11111	11111	11111	
2073	11111	11111	11111	
2074	11111	11111	11111	
2075	11111	11111	11111	
2076	11111	11111	11111	
2077	11111	11111	11111	
2078	11111	11111	11111	
2079	11111	11111	11111	
2080	11111	11111	11111	
2081	11111	11111	11111	
2082	11111	11111	11111	

TABLA 4.10

ARCHIVO DIRECTO CON LOS APUNTADORES DE LA SUBLLAVE UNO

DIRECCION ANTERIOR	DIRECCION ACTUAL	DIRECCION SIGUIENTE	SUBLLAVE	DIRECCION PRINCIPAL
2023	2023	2023	8	2001
2024	2024	2024	53	2018
2025	2025	2025	8	2009
2026	2026	2038	24	2015
2027	2027	2027	82	2020
2028	2028	2028	35	2008
2029	2029	2029	67	2006
2030	2030	2050	92	2012
2031	2031	2032	20	2010
2031	2032	2032	20	2014
2033	2033	2033	84	2011
2034	2034	2034	74	2019
2035	2035	2035	13	2013
2036	2036	2036	97	2002
2037	2037	2037	30	2003
2026	2038	2038	24	2016
2039	2039	2039	42	2006
2040	2040	2040	28	2007
2041	2041	2041	76	2017
2042	2042	2042	51	2005

TABLA 4.11

ARCHIVO DIRECTO CON LOS APUNTADORES DE LA SUBLLAVE: DOS

DIRECCION ANTERIOR	DIRECCION ACTUAL	DIRECCION SIGUIENTE	SUBLLAVE	DIRECCION PRINCIPAL
2043	2043	2056	5	2001
2044	2044	2049	8	2018
2045	2045	2048	0	2009
2046	2046	2047	4	2015
2046	2047	2061	4	2020
2045	2048	2050	0	2008
2044	2049	2049	8	2004
2048	2050	2050	C	2012
2051	2051	2058	1	2010
2052	2052	2059	7	2014
2053	2053	2053	2	2011
2054	2054	2055	3	2019
2054	2055	2057	3	2013
2043	2056	2056	5	2002
2055	2057	2057	3	2003
2051	2058	2062	1	2016
2052	2059	2059	7	2006
2060	2060	2060	6	2007
2047	2061	2061	4	2017
2058	2062	2062	1	2005

TABLA 4.12

RASTREO DEL ACCESO DE LA LLAVE PRINCIPAL

REGISTRO	LLAVE PRINCIPAL	ACCESO REQUERIDO		INTENTO VALIDO	DIRECCION CALCULADA	ACCESADA	CAMPO LLAVE PRINCI	TIPO SOBREFLUJO
1	198	1	L	1	2001	2001	198	
2	7624	1	L	8	2018	2018	7624	
3	2795	1	L	3	2009	2009	2795	
4	6221	1	L	7	2015	2015	6221	
5	8218	1	L	9	2020	2020	8218	
6	2620	1	L	3	2008	2008	2620	
7	870	1	L	1	2004	2004	870	
8	4650	1	L	5	2012	2012	4650	
9	3414	1	L	4	2010	2010	3414	
10	5851	1	L	6	2014	2014	5851	
11	3670	1	L	4	2011	2011	3670	
12	7783	1	L	8	2019	2019	7783	
13	4837	1	L	5	2013	2013	4837	
14	469	1	L	1	2002	2002	469	
15	313	1	L	1	2002	2002	469	
15	313	2	L	1	2002	2003	313	
16	6682	1	L	7	2016	2016	6682	
17	1378	1	L	2	2006	2006	1378	
18	2135	1	L	3	2007	2007	2135	
19	7239	1	L	8	2017	2017	7239	
20	1034	1	L	2	2005	2005	1034	

TABLA 6.13

RASTREO DEL ENCADENAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA UNO

REGISTRO	SUBLLAVE UNO	ACCESO NUMERO	TIPO	ARCHIVO DE APUNTADORES DIR ACT	DIR SIG	DIR ARCH PRIN	ARCHIVO PRINCIPAL LLAVE PRINCIPAL	SUBLLAVE DOS	OTRO DATO
1	8	2	L	2023	2025	2001	148	5	H
2	8	4	L	2025	2025	2009	2795	0	H
3	13	2	L	2035	2035	2013	4837	3	H
4	20	2	L	2031	2032	2010	3414	1	H
5	20	4	L	2032	2032	2014	5813	7	H
6	24	2	L	2026	2038	2015	6221	4	H
7	24	4	L	2038	2038	2019	6682	1	H
8	28	2	L	2040	2040	2017	2135	6	H
9	30	2	L	2037	2037	2003	913	3	H
10	33	2	L	2024	2024	2018	7624	8	H
11	35	2	L	2028	2028	2008	2420	0	H
12	42	2	L	2039	2039	2004	1378	7	H
13	51	2	L	2042	2042	2021	3034	1	H
14	67	2	L	2029	2029	2006	870	8	H
15	74	2	L	2034	2034	2012	7783	3	H
16	78	2	L	2043	2043	2022	7439	4	H
17	82	2	L	2027	2027	2020	8218	4	H
18	84	2	L	2033	2033	2011	3670	2	H
19	92	2	L	2030	2030	2002	4450	0	H

TABLA N 11

RASTREO DEL ENCAMINAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA UNO

REGISTRO	SUB LLAVE UNO	ACCESO MEDIO	TIPO	ARCHIVO DE APUNTADORES DIR ACCESO	DIR SIG	DIR AREA PRIN.	ARCHIVO PRINCIPAL LLAVE PRINCIPAL	SUB LLAVE DOS	OTRO DATO
20	97	2	I	2036	2036	3002	459	S	H

TABLA 4.14

RASTREO DEL ENCADENAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA: DOS

##	SUB	ACCESO	ARCHIVO DE APUNTADORES			ARCHIVO PRINCIPAL		
DIS	LLAVE	NU TI	DIR	DIR	DIR	LLAVE	SUB	OTRO
TRO	DOS	ME PO	ACT	SIG	ARCH	PRIN	LLAVE	DATO
		RO			PRIN	CIPAL	UNO	
1	0	2 L	2065	2068	2009	2795	8	N
2	0	6 L	2068	2050	2008	2620	35	N
3	0	6 L	2050	2050	2012	4650	92	N
4	1	2 L	2051	2058	2010	5414	20	N
5	1	6 L	2058	2062	2016	6682	24	N
6	1	6 L	2062	2062	2005	1034	51	N
7	2	2 L	2053	2053	2011	3670	84	N
8	3	2 L	2054	2055	2019	7783	74	N
9	3	4 L	2055	2057	2013	4837	13	N
10	3	6 L	2057	2057	2003	113	30	N
11	4	2 L	2066	2067	2015	6221	24	N
12	4	4 L	2067	2061	2020	8218	82	N
13	4	6 L	2061	2061	2017	7239	76	N
14	5	2 L	2063	2064	2000	198	8	N
15	5	4 L	2056	2056	2002	469	97	N
16	6	2 L	2060	2060	2007	2135	28	N
17	7	2 L	2052	2059	2014	5851	20	N
18	7	4 L	2059	2059	2006	1378	42	N
19	8	2 L	2044	2049	2018	7624	53	N

TABLA 4.14

REGISTRO DEL ENSAMBLAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA DOS

NO	SUB	ACCESO		ARCHIVO DE APUNTADORES			ARCHIVO PRINCIPAL		
DIS	LLAVE	NU	TI	DIR	DIR	DIR	LLAVE	SUB	OTRO
TRO	DOS	NE	PO	ACT	SIG	ARCH	PRIN	LLAVE	DATO
		RO				PRIN	CIPAL	UNO	
20	B	4	L	2049	2049	2004	870	67	H

****

ERRORES DETECTADOS DURANTE EL ACCESO DE REGISTROS

****

```
**************************************************

1 - Método de acceso por interpolación lineal

2 - Método de acceso por interpolación de lagrange

3 - Método de acceso por suma y por división

**************************************************


                Qué método de acceso desea ? 2


**********************************************************

  ACCESO EN ARCHIVO POR INTERPOLACION DE LAGRANGE

   ACCESO EN SOBREFLUJO POR INTERPOLACION LINEAL

**********************************************************


**************************************************************

          ESTADISTICAS DEL ACCESO DE REGISTROS

**************************************************************


     REGISTROS LEIDOS    20

REGISTROS LOCALIZADOS    20

     TOTAL COLISIONES:    1

    NUMERO DE ACCESOS:    21

NUMERO DE ACCESOS PROMEDIO:  1.0500E+00
```

****

ESTADISTICAS DEL ACCESO DE REGISTROS

****

ACCESADOS EN AREA NORMAL	20
EN PRIMER INTENTO	19
EN SEGUNDO INTENTO	1
EN TERCER INTENTO	0
EN CUARTO INTENTO	0
EN QUINTO INTENTO	0
ACCESADOS EN AREA DE SOBREFLUJO	0
EN SEXTO INTENTO	0
EN SEPTIMO INTENTO	0
EN OCTAVO INTENTO	0
EN NOVENO INTENTO	0
EN DECIMO INTENTO	0
ACCESADOS EN AREA ESPECIAL DE SOBREFLUJO	0
EN ONCEAVO INTENTO	0
EN DOCEAVO INTENTO	0
EN TRECEAVO INTENTO	0
EN CATORCEAVO INTENTO	0
EN QUINCEAVO INTENTO	0

```
**********************************************************

          ESTADISTICAS DEL ACCESO DE REGISTROS

**********************************************************


COLISIONES EN AREA NORMAL      1

        EN PRIMER INTENTO      1

       EN SEGUNDO INTENTO      0

        EN TERCER INTENTO      0

        EN CUARTO INTENTO      0

        EN QUINTO INTENTO      0

COLISIONES EN AREA DE SOBREFLUJO      0

                EN SEXTO INTENTO      0

              EN SEPTIMO INTENTO      0

               EN OCTAVO INTENTO      0

               EN NOVENO INTENTO      0

               EN DECIMO INTENTO      0

COLISIONES EN AREA ESPECIAL DE SOBREFLUJO      0

                    EN ONCEAVO INTENTO      0

                   EN DOCEAVO INTENTO      0

                  EN TRECEAVO INTENTO      0

                EN CATORCEAVO INTENTO      0

                 EN QUINCEAVO INTENTO      0



**********************************************************

     ESO ES TODO EN CUANTO A ACCESO DE REGISTROS

**********************************************************
```

TABLA N° 15

REGISTRO DEL ACCESO DE LA CLAVE PRINCIPAL

NE	LLAVE	ACCESO	IN	DIRECCION	CAMPO	TIPO
GIS	PRIN	NU TI	TER	FALSO	ACCE	LLAVE	SOBRE
TRO	CIPAL	ME PO	VALO	CADA	VACA	PRINCI	FLUJO
1	w69	1 L	1	2002	2002	w69	

TABLA N° 16

REGISTRO DEL ACCESO DE LA CLAVE PRINCIPAL 7624

NE	LLAVE	ACCESO	IN	DIRECCION	CAMPO	TIPO
GIS	PRIN	NU TI	TER	FALSO	ACCE	LLAVE	SOBRE
TRO	CIPAL	ME PO	VALO	CADA	VACA	PRINCI	FLUJO
1	7624	1 L	A	2018	2018	7624	

TABLA 4.17

RASTREO DEL ACCESO DE LA LLAVE PRINCIPAL 6666

REGISTRO	LLAVE PRIN LLPAL	ACCESO NUME	TIPO	INTERVALO	DIRECCION CALCULADA	DIRECCION ACCESADA	CAMPO LLAVE PRINCIP.	TIPO SOBREFLUJO
1	6666	1	L	7	2016	2016	6682	
1	6666	2	L	7	2021	2021	11111	1
1	6666	3	L	7	2021	2022	11111	1
1	6666	4	L	7	2075	2075	11111	2
1	6666	5	L	7	2075	2076	11111	2
1	6666	6	L	7	2075	2076	11111	2
1	6666	7	L	7	2075	2077	11111	2
1	6666	8	L	7	2075	2073	11111	2

TABLA 4.15

RASTREO DEL ENCADENAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA UNO 20

REGISTRO	SUBLLAVE UNO	ACCESO MEDIANTE	NO. DE PASOS	ARCHIVO DE APUNTADORES DIR ACT	DIR SIG	DIR ARCH PRINCIPAL	ARCHIVO PRINCIPAL LLAVE PRINCIPAL	SUBLLAVE DOS	OTRO DATO
1	20	R	2	2031	2032	2010	3416	1	H
2	20	A	4	2032	2032	2016	5851	7	H

TABLA 4.16

RASTREO DEL ENCADENAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA LLAVE SECUNDARIA UNO 24

REGISTRO	SUBLLAVE UNO	ACCESO MEDIANTE	NO. DE PASOS	ARCHIVO DE APUNTADORES DIR ACT	DIR SIG	DIR ARCH PRINCIPAL	ARCHIVO PRINCIPAL LLAVE PRINCIPAL	SUBLLAVE DOS	OTRO DATO
1	24	R	2	2026	2038	2015	6221	4	H
2	24	A	4	2038	2038	2016	6682	1	H

TABLA 6.20

RASTREO DEL ENCADENAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA CLAVE SECUNDARIA DOS = 4

REG	SUB	ACCESO		ARCHIVO DE APUNTADORES			ARCHIVO PRINCIPAL		
REG	LLAVE	NU	TI	DIR	DIR	DIR	LLAVE	SUB	OTRO
TRO	DOS	ME	PO	ACT	SIG	ARCH	PRIN	LLAVE	DATO
		RO				PRIN	CIPAL	UNO	
1	4	2	L	2046	2047	2015	6221	24	M
2	4	4	L	2047	2061	2020	8218	82	M
3	4	6	L	2061	2061	2017	7239	76	R

TABLA 6.21

RASTREO DEL ENCADENAMIENTO DE ACCESO DE LAS

DIRECCIONES DE LA CLAVE SECUNDARIA DOS = 6

REG	SUB	ACCESO		ARCHIVO DE APUNTADORES			ARCHIVO PRINCIPAL		
REG	LLAVE	NU	TI	DIR	DIR	DIR	LLAVE	SUB	OTRO
TRO	DOS	ME	PO	ACT	SIG	ARCH	PRIN	LLAVE	DATO
		RO				PRIN	CIPAL	UNO	
1	6	2	L	2060	2060	2007	2135	28	M

****

MÉTODO DE LA COMBINACIÓN INTEGRATIVA DE ACCESO SECUENCIAL

****

1 - Método de acceso por interpolación lineal

2 - Método de acceso por interpolación de Lagrange

3 - Método de acceso por suma y por división

****

¿Qué método de acceso desea ? 2

****

ACCESO EN SECUENCIA POR INTERPOLACIÓN DE LAGRANGE

ACCESO EN SECUENCIA POR INTERPOLACIÓN LINEAL

****

****

1 - LOCALIZAR CLAVE PRINCIPAL

2 - LOCALIZAR SUBCLAVE UNO

3 - LOCALIZAR SUBCLAVE DOS

****

QUE TIPO DE CLAVE DESEA LOCALIZAR ? 1

CLAVE PRINCIPAL A LOCALIZAR ? 449

```
**************************************

NO EXISTE EN EL ARCHIVO LA CLAVE  00000

**************************************


  DESEA CONTINUAR (S/N) ? N


*****************************

1  LOCALIZAR CLAVE PRINCIPAL

2  LOCALIZAR SUBCLAVE UNO

3  LOCALIZAR SUBCLAVE DOS

*****************************


  QUE TIPO DE CLAVE DESEA LOCALIZAR ? 2


          SUBCLAVE UNO A LOCALIZAR ? 20


  DESEA CONTINUAR (S/N) ? S
```

```
**********************************

1   LOCALIZAR CLAVE PRINCIPAL

2   LOCALIZAR SUBCLAVE UNO

3   LOCALIZAR SUBCLAVE DOS

**********************************


   QUE TIPO DE CLAVE DESEA LOCALIZAR ? 2


                  SUBCLAVE UNO A LOCALIZAR ? 24


   DESEA CONTINUAR (S/N) ? S


**********************************

1   LOCALIZAR CLAVE PRINCIPAL

2   LOCALIZAR SUBCLAVE UNO

3   LOCALIZAR SUBCLAVE DOS

**********************************


   QUE TIPO DE CLAVE DESEA LOCALIZAR ? 2


                  SUBCLAVE UNO A LOCALIZAR ? 66
```

```
**********************************************

NO EXISTE EN EL ARCHIVO LA SUBLLAVE UNO  00

**********************************************



  DESEA CONTINUAR (s/n) ? s



**********************************

1   LOCALIZAR LLAVE PRINCIPAL

2   LOCALIZAR SUBLLAVE UNO

3   LOCALIZAR SUBLLAVE DOS

**********************************



  QUE TIPO DE LLAVE DESEA LOCALIZAR ?  3


              SUBLLAVE DOS A LOCALIZAR ?  4



 DESEA CONTINUAR (s/n) ? s
```

```
**********************************

1 - LOCALIZAR LLAVE PRINCIPAL

2 - LOCALIZAR SUBLLAVE UNO

3 - LOCALIZAR SUBLLAVE DOS

**********************************



   QUE TIPO DE LLAVE DESEA LOCALIZAR ? 3



                SUBLLAVE DOS A LOCALIZAR ? 6



  DESEA CONTINUAR (s/n) ? s

**********************************

1 - LOCALIZAR LLAVE PRINCIPAL

2 - LOCALIZAR SUBLLAVE UNO

3 - LOCALIZAR SUBLLAVE DOS

**********************************



   QUE TIPO DE LLAVE DESEA LOCALIZAR ? 3



                SUBLLAVE DOS A LOCALIZAR ? 9
```

```
**********************************************

NO ESTA EN EL ARCHIVO LA SUBLLAVE DOS. B

**********************************************




 DESEA CONTINUAR (s/n) ? n


**************************************************

    ESO ES TODO EN CUANTO A ACCESO DE REGISTROS

**************************************************

**************************************************

ERRORES DETECTADOS DURANTE EL ACCESO INDIVIDUAL DE REGISTROS

**************************************************
```

CONCLUSIONES

Las primeras conclusiones se harán en base a las siguientes estadísticas generadas a partir de 20 archivos de 100 registros generados en forma aleatoria cada uno y dispersados y accesados con los tres algoritmos de dispersión que se diseñaron en el capítulo 3.2: interpolación lineal, interpolación de Lagrange de segundo orden y suma-división respectivamente.

Las estadísticas se presentan por grupos de 3 renglones, el primer renglón corresponde a la interpolación lineal, el segundo a la interpolación de Lagrange de segundo orden y el tercer renglon se refiere al algoritmo de suma-división.

La columna PAR representa el número promedio de accesos por registro.

La columna AA indica el número de registros accesados en el primer intento, es decir, los registros que se accesaron sin ninguna colisión.

La columna BB indica el número de accesos que se hicieron en el archivo directo principal.

La columna CC indica el número de accesos que se hicieron en el área de sobreflujo del archivo directo principal.

La columna DD indica el número de accesos que se hicieron en el archivo directo de sobreflujo especial.

La columna EE indica el número de colisiones que se presentaron en el archivo directo principal.

La columna FF indica el número de colisiones que se presentaron en el área de sobreflujo del archivo directo principal.

La columna GG indica el número de colisiones que se presentaron en el archivo directo principal.

ESTADISTICAS DEL ACCESO DE REGISTROS

ARCHI	NUM	NUM	NUM	PAR	AA	ACCESOS			COLISIONES		
VO	REG	COL	ACC			RR	CC	RB	EE	FF	OV
1	100	100	200	2.00	66	89	9	2	85	15	0
1	100	78	178	1.78	67	93	6	1	72	6	0
1	100	114	214	2.14	60	88	10	2	96	18	0
2	100	77	177	1.77	67	93	5	2	69	8	0
2	100	70	170	1.70	65	95	5	0	69	1	0
2	100	75	175	1.75	66	92	8	0	71	4	0
3	100	114	214	2.14	65	87	9	4	86	27	1
3	100	87	187	1.87	70	92	7	1	79	8	0
3	100	94	194	1.94	62	90	8	2	78	16	0
4	100	92	192	1.92	62	89	10	1	79	13	0
4	100	73	173	1.73	64	92	7	1	65	8	0
4	100	105	205	2.05	63	88	8	4	89	16	0
5	100	92	192	1.92	64	89	9	2	82	10	0
5	100	86	186	1.86	63	93	6	1	80	6	0
5	100	87	187	1.87	64	91	9	0	79	8	0
6	100	107	207	2.07	61	90	7	3	90	17	0
6	100	69	169	1.69	68	93	7	0	67	2	0
6	100	125	225	2.25	59	86	9	5	95	30	0

ARCHI	NUM	NUM	NUM	PAR	AA	ACCESS			COLISIONNES		
VO	AEG	10L	AFC			BB	CC	DD	EE	FF	GG
7	100	76	176	1.76	68	92	8	0	70	6	0
7	100	63	163	1.63	61	95	5	0	62	1	0
7	100	77	177	1.77	65	93	7	0	77	0	0
8	100	99	199	1.99	61	91	8	1	89	10	0
8	100	78	178	1.78	61	94	6	0	76	2	0
8	100	91	191	1.91	62	92	8	0	86	5	0
9	100	87	187	1.87	66	91	8	1	78	9	0
9	100	66	166	1.66	70	95	5	0	63	3	0
9	100	110	210	2.10	64	89	7	4	83	27	0
10	100	79	179	1.79	73	90	8	2	66	13	0
10	100	48	148	1.48	78	95	5	0	48	0	0
10	100	81	181	1.81	66	90	8	2	65	16	0
11	100	103	203	2.03	66	89	8	3	81	20	0
11	100	42	142	1.92	66	93	7	2	81	11	0
11	100	65	165	1.65	65	95	5	0	64	1	0
12	100	97	197	1.97	64	93	7	2	82	13	0
12	100	68	168	1.68	69	95	5	0	63	5	0
12	100	74	174	1.74	65	92	7	1	64	10	0

[illegible]	[illegible]	[illegible]	[illegible]	PAR	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]			[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
13	100	79	179	1 79	64	88	9	3	81	18	0
13	100	78	178	1 78	66	92	8	0	79	3	0
13	100	87	187	1 87	62	92	8	0	85	2	0
14	100	84	184	1 84	67	91	9	0	74	10	0
14	100	70	170	1 70	64	94	6	0	69	1	0
14	100	96	196	1 96	63	89	9	2	81	15	0
15	100	75	175	1 75	64	93	7	0	71	4	0
15	100	63	163	1 63	67	96	4	0	62	1	0
15	100	107	207	2 07	64	90	8	2	89	18	0
16	100	101	201	2 01	66	88	7	5	78	22	1
16	100	75	175	1 75	70	91	7	2	67	8	0
16	100	87	187	1 87	66	90	9	1	79	8	0
17	100	96	196	1 96	58	90	10	0	88	8	0
17	100	75	175	1 75	65	95	6	0	72	3	0
17	100	92	192	1 92	62	89	10	1	82	10	0
18	100	88	188	1 88	66	90	9	1	79	9	0
18	100	75	175	1 75	65	93	7	0	72	3	0
18	100	80	180	1 80	67	90	8	2	70	10	0

ESTADISTICAS DEL ACCESO DE REGISTROS

ARCHI	NUM	NUM	NUM	PAR	AA	ACCESOS			COLISIONES		
VO	REG	EOL	ACC			BB	CC	DD	EE	FF	GG
10	100	111	211	2 11	58	89	8	3	89	22	0
10	100	87	187	1 87	68	91	7	2	73	14	0
10	100	93	193	1 93	63	92	7	1	84	9	0
20	100	79	179	1 79	67	91	7	2	64	15	0
20	100	58	158	1 58	70	95	5	0	54	4	0
20	100	100	200	2 00	67	87	9	4	75	25	0

PROMEDIOS DE LAS ESTADISTICAS DEL ACCESO DE REGISTROS

NUM	NUM	NUM	PAR	AA	ACCESOS			COLISIONES		
REG	EOL	ACC			BB	CC	DD	EE	FF	GG
100	92 70	192 70	1 927	64 70	90 05	8 30	1 65	79 55	15 45	0 00
100	92 00	192 00	1 920	64 00	90 25	8 50	1 50	80 60	14 40	0 00
100	92 00	192 00	1 920	64 60	90 25	8 50	1 65	79 60	12 40	0 00

De las estadísticas anteriores se puede concluir que el mejor algoritmo de dispersión es el de interpolación de Lagrange de segundo orden debido a que se obtuvo

1) el promedio del número de colisiones más bajo: 72.95.

2) el promedio del número total de accesos más bajo: 172.95

3) el promedio de accesos por registro más bajo: PARA 1.729

4) el promedio del número de registros accesados en el primer intento más alto: 87.15.

5) el promedio del número de registros accesados en el archivo directo principal más alto: 93.65.

6) el promedio del número de registros accesados en el área de sobreflujo del archivo directo principal más bajo: 6.05.

7) el promedio del número de registros accesados en el archivo directo de sobreflujo especial más bajo: 0.50

8) el promedio del número de colisiones en el archivo directo principal más bajo: 68.55.

9) el promedio del número de colisiones en el área de sobreflujo del archivo directo principal más bajo: 4.40.

10) el promedio del número de colisiones en el archivo directo de sobreflujo especial igual a cero

De los valores de las tablas mostradas anteriormente podemos concluir que el algoritmo de dispersión por el método de interpolación de Lagrange de segundo orden resultó ser el mejor ya que se obtienen los valores mas bajos en los parámetros evaluados. Siendo estos los mas significativos en los archivos de acceso directo.

Entre menor sea el número de accesos para grabar ó recuperar un registro el algoritmo de dispersión es mejor ya que lo realiza en un tiempo menor, lo cual es muy importante en los sistemas interactivos de tiempo real.

Este aspecto se conoce como "tiempo de respuesta" siendo este el tiempo que transcurre desde que el usuario transmite su información y el procesador entrega la respuesta en la pantalla.

El mecanismo utilizado para la resolución de colisiones es el mismo mecanismo que utiliza la dispersión, solo que en este caso se utiliza un espacio de memoria mas pequeño sin embargo un número menor de registros tienen necesidad de este espacio.

Por último los registros que no fueron grabados en esta área y que son registros colisionados se graban en un espacio de memoria independiente el cual tiene la característica de ser secuencial. Sin embargo un número mínimo de registros requieren ser grabados en esta área.

Se debe hacer mención que los valores obtenidos en las tablas resultan de la muestra de 100 registros grabados en el archivo de datos el cual fue dimensionado de acuerdo al volúmen de registros de la muestra.

En un ambiente de operación real, el volúmen de registros que se graban en los archivos deben ser estimados dejandose un margen del holgura del 20 porciento de registros adicionales para un posible crecimiento.

Con este espacio se protege el archivo de una posible saturación y además la eficiencia del algoritmo de dispersión se mejora

En general podemos concluir que la herramienta desarrollada en este trabajo puede ser de gran utilidad en el desarrollo de sistemas que deban operar en un ambiente transaccional en donde el tiempo de respuesta es un factor determinante

El manejador de archivos facilita el desarrollo de los programas al proporcionar el medio para el acceso a los archivos y el control de los mismos, este puede ser utilizado desde cualquier programa escrito en cualquier lenguaje de programación , reduciendo significativamente los tiempos de desarrollo y estandarizando la forma y el medio de acceso a los archivos

APLICACIONES

La aplicación principal del manejador de archivos es facilitar al programador el acceso a archivos de tipo directo evitándole tener que definir una función de dispersión y la resolución de colisiones llamando de una manera simple desde cualquier lenguaje de programación a la rutina de acceso.

El manejador de Archivos puede ser utilizado en cualquier sistema que deba operar en un ambiente transaccional. El sistema puede ser tan simple o complicado como sea necesario ya que esto dependerá de la propia aplicación en sí y no del manejador de archivos ya que éste solo realiza la función de acceso y las operaciones de grabación y recuperación de datos y el acceso a los archivos.

El único requerimiento de la rutina es que debe correr en sistemas de cómputo en donde exista el recurso de archivos de acceso directo.

El manejador de archivos actua como una subrutina cuando es llamado desde otro programa por esto es necesario que exista un área común entre el programa del usuario y la subrutina, esta área se llama área de encadenamiento y contiene los parámetros y datos que se pasan entre el programa principal y la subrutina.

Ejemplo de la definición del área de entendimiento en un programa escrito en lenguaje COBOL.

```
LINKAGE-SECTION

        01 AREA-DE-DATOS          PICTURE X(90).   area para el dato principal

        01 PARAMETROS

           03 OPERACION           PICTURE X(02)

           03 NUMERO-DE-ARCHIVO PICTURE X(02)

           03 CODIGO-DE-RETORNO PICTURE X(04)
```

Ejemplo de una llamada a la subrutina desde un programa de usuario:

```
CALL "ACCESO" USING AREA-DATOS PARAMETROS
```

En el área de datos se graba el registro que se va a insertar o a recuperar, el area de parámetros está constituida por la operación, número de archivo a accesar y código de retorno.

Las operaciones que se pueden realizar son:

01 inserta registro

02 recupera registro

03 modifica registro

04 borra registro

El código de retorno al inicio de la llamada lleva el mismo valor de la operación, cuando la operación no se realiza correctamente este campo es modificado por la subrutina con el código de error que indica el sistema operativo

Inserción

Esta operación tiene como objetivo insertar un nuevo registro dentro del archivo de datos así como en sus archivos de apuntadores. Su operación incluye la ejecución de la función de dispersión así como la resolución de colisiones, estas funciones se describen en otro capítulo. La operación de inserción determina a través de la tabla de localización de archivos el espacio de direcciones en el cual va a realizar la función de dispersión para escribir la llave principal y sus apuntadores.

En el archivo principal se graba el registro completo de datos de acuerdo a como está definido en el programa del usuario conservando el valor de la dirección física en donde se graba el dato. Esta dirección será utilizada para adicionarla en el apuntador.

Recuperación

Esta operación de la rutina realiza la recuperación del registro que fue grabado con anterioridad. Cuando se desea recuperar el dato por la llave principal esta operación la rutina realiza la función de dispersión y recupera el dato en caso de existir, si el dato no existe la rutina devuelve un estado de operación que indica que el dato no fue localizado.

Borrado

Mediante esta operación se realiza la baja lógica en los espacios de direcciones y apuntadores, esta operación se realiza mediante cambiar el valor de situación del apuntador de un cero a un uno. El uno en el campo de situación indica que el registro esta dado de baja lógicamente, para realizar la baja física es necesario reorganizar el archivo leyendo todos los datos, clasificarlos y cargarlo nuevamente el archivo

Modificación

Esta estructura consiste en accesar el espacio de direcciones principal para cambiar datos que no forman parte de la llave. Debido a que en esta operación no hay cambio en el valor de la llave, la dirección física del registro no se ve modificada al realizar un cambio en los datos.

Criterios para la asignación de espacio de archivos de acceso directo.

Los archivos de acceso directo son estáticos en su crecimiento, es decir que el espacio debe asignarse desde el inicio de su creación considerando el volumen de datos a grabar más el crecimiento que tendrá el archivo

Es por esto que se debe ser muy cuidadoso para asignar el espacio de un archivo de acceso directo ya que este quedará fijo durante el uso del archivo o hasta que este sea reorganizado nuevamente

Así también es importante considerar que asignar espacio que no va a ser utilizado es costoso para cualquier sistema de cómputo, ya que se están reservando de manera exclusiva recursos que no se van a utilizar.

Debido a que la función de dispersión depende directamente del valor de la llave de acceso, la cual tiene un comportamiento de tipo aleatorio, es posible que el espacio de direcciones quede con espacios o huecos que no fueron asignados por la función de dispersión de tal manera que siempre se debe considerar dar holgura al espacio de direcciones

La asignación del espacio en disco para almacenar la información, es un aspecto importante cuando se diseña un sistema de acceso directo. Deben considerarse los siguientes criterios :

número de registros iniciales a almacenar

El analista deberá realizar un análisis de los datos que se van a grabar determinando el volumen inicial de registros de acuerdo al análisis de la información actual

Crecimiento del Archivo.

Hacer un análisis para determinar en base a la frecuencia de uso de la información el número de registros que se incrementan en un período determinado de tiempo, después de este período se debe realizar una reorganización del archivo

Longitud del registro físico

La longitud del registro físico ayuda a calcular el número de caracteres que se grabarán en el espacio de direcciones.

Ejemplo dimensionamiento de un archivo de datos principal con tres llaves.

Datos:

Registros iniciales 1,000

Crecimiento en un mes 100

Longitud del registro 120 caracteres

Cálculo

Archivo directo principal:

1100 registro x 120 caracteres = 132 000 caracteres

132 000 / 1024 = 128 90 Kbytes

Area de sobreflujo del archivo directo principal:

10 % del archivo directo principal = 128 90 Kbytes / 10 = 12.89 Kbytes

Archivo directo de apuntadores para la subllave 1

1100 registros x longitud apuntador = 1100 x 9 = 9 900 caracteres

9 900 / 1024 = 9.67 Kbytes

Archivo directo de apuntadores para la subllave 2

1100 registros x longitud apuntador = 1100 x 10 = 11 000 caracteres

11 000 / 1024 = 10.74 Kbytes

Espacio total requerido

Archivo directo principal	128.90 Kbytes
Area de sobreflujo del archivo directo principal	12.89 Kbytes
Archivo directo de apuntadores para la subllave 1	9.67 Kbytes
Archivo directo de apuntadores para la subllave 2	10.74 Kbytes
total requerido	162.20 Kbytes

BIBLIOGRAFIA

Computer data base organization

James Martin

Addison Wesley

QA76 M324

Advanced database techniques

Daniel Martin

Massachusetts Institute of Technology, 1986

QA76 9 D3M33

File organization for database design

Gio Wiederhold

McGraw-Hill, 1987

QA76 9F5 W54

An introduction to data structures with applications

Jean Paul Tremblay, Paul G Sorenson

McGraw Hill, 1984

QA76 9 T35I73