

35
24



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

DISEÑO E IMPLANTACION DE
S A D I N
SISTEMA AUXILIAR PARA EL
DIAGNOSTICO NEUROLOGICO

T E S I S

D U E P R E S E N T A N:
KATIA R. MONTERO FERNANDEZ
GEORGE GELMAN KIPNIS
PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACION

DIRECTORES DE TESIS:

ING. FEDERICO FERNANDEZ CANCINO +
DR. FRANCISCO CERVANTES PEREZ



México, D. F.

FALLA DE ORIGEN

1991



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

TESIS CON FALLA DE ORIGEN

ÍNDICE

I. INTRODUCCIÓN	1
II. REVISIÓN BIBLIOGRÁFICA EN EL ÁREA DE LA INTELIGENCIA ARTIFICIAL	7
1. - Antecedentes	8
2. - Áreas de Investigación	10
3. - Sistemas Expertos	16
3.1. - Clasificación	17
3.2. - Arquitectura	19
3.3. - Técnicas	21
3.4. - Algunos sistemas y sus características	22
3.5. - Requisitos de un sistema experto de calidad	32
III. ANÁLISIS Y DISEÑO DE SADIN: Sistema Auxiliar para el Diagnóstico Neurológico	35
1. - Análisis del Diagnóstico Neurológico	36
2. - Diseño de SADIN	40
2.1. - Arquitectura global	40
2.2. - Base de conocimiento	43
2.3. - Máquina de Inferencias	50
2.4. - Interfaz con el Experto	56
2.5. - Interfaz de Mantenimiento	57
2.6. - Interfaz con el Usuario	68
2.7. - Explicador	70
3. - Implantación	71
3.1. - Elección de los Lenguajes	72
3.2. - Desarrollo del Sistema	78
IV. DISCUSIÓN Y CONCLUSIONES	85
REFERENCIAS	93
BIBLIOGRAFÍA	97
FIGURAS	101
SUPLEMENTO I: INTERFAZ CON EL USUARIO Y MÁQUINA DE INFERENCIAS	109
SUPLEMENTO II: INTERFAZ DE MANTENIMIENTO	129

I. INTRODUCCIÓN

El presente escrito es la culminación de un trabajo que se inició como una inquietud por desarrollar algo práctico y con posibilidades comerciales dentro del vasto terreno de la Inteligencia Artificial. Esta inquietud concluyó no sólo con el diseño e implantación de un sistema útil en un campo específico del conocimiento humano, el diagnóstico neurológico, sino además, con posibilidades de utilización en una multitud de otras áreas de aplicación.

Al iniciar este trabajo, se detectó que la inquietud mencionada tenía un eco en numerosas empresas, organismos y personas, que necesitaban un apoyo eficiente y oportuno para su cotidiano proceso de toma de decisiones, bajo circunstancias de información incierta e incompleta, esto es, situaciones en las que frecuentemente no se posee la totalidad de los elementos de juicio necesarios.

1. INTRODUCCION

Al revisar las distintas áreas que integran la Inteligencia Artificial, se percato de que las características que presentaban estas situaciones eran precisamente las que pueden ser resueltas con un sistema experto: necesidades de una rapidez en la toma de decisiones, de contar con extensos conocimientos expertos y de evaluaciones eficaces en un amplio dominio de resultados, por mencionar solo algunas.

Estos sistemas, igual que la Inteligencia Artificial en su totalidad, nacieron en el siglo XX. Son la conjugación de los métodos de inferencia simbólica con la información particular de algún campo específico del conocimiento, todo ello implantado en un sistema computacional.

Casi inmediatamente a su despertar favorecidos por los alentadores resultados de los primeros sistemas, comenzaron a crearse una variedad de estos "consultores del futuro". Es así como aparecieron sistemas para el diagnóstico médico, exploración minera y configuración de computadoras entre muchos otros (*Duda, R.O. y Goshing, J.G., 1981*), (*The Handbook of Artificial Intelligence, 1982*).

Paulatinamente se evidenció que los campos de aplicación de los asesores electrónicos podían ser más amplios, y a partir de entonces es posible encontrar sistemas expertos en tareas tan diversas como decisiones financieras, autorizaciones de líneas de crédito, inteligencia civil, venta de seguros y muchas otras más (*Coxson, W.M., 1987*).

Este auge de los sistemas expertos llevó a la creación de los "shells" o estructuras de propósito general, es decir, sistemas expertos que podían ser ajustados por el usuario para funcionar en cualquier campo del conocimiento humano. La aparición a nivel comercial de

los "shells" permitió una mayor diversificación de las aplicaciones de los sistemas expertos y su familiarización por parte de usuarios con poco conocimiento de cómputo y de la Inteligencia Artificial.

Sin embargo, una mayor utilización de los "shells" demostró que presentaban algunas inconveniencias frente a los sistemas expertos "hechos a la medida", las cuales no habían sido tomadas en cuenta con anterioridad. Estas inconveniencias se basaban en que distintas áreas del conocimiento humano requieren de estructuras propias para recopilar y conservar el conocimiento, así como de diversas estrategias de búsqueda de solución dentro de este conocimiento. Es por ello que sistemas de propósito general, con una sola y rígida estructura de conocimiento y de búsqueda no permitían resultados óptimos.

El contemplar la situación de los sistemas expertos bajo esta perspectiva, motivó el objetivo general de desarrollar un sistema experto que utilizara como modelo de razonamiento aquel que usa en forma natural el experto en el área de interés. Además, al ser el presente trabajo una tesis de Ingeniería, se incluyeron en el proyecto los siguientes objetivos:

- una revisión bibliográfica sobre el tema de la Inteligencia Artificial, en el área de la Ingeniería de Conocimiento, especialmente la que se refiere a los Sistemas Expertos,
- el desarrollo e implantación de un sistema experto, con base en las técnicas y metodologías ligadas en forma directa con la Ingeniería en Computación, como son las de ingeniería de software, de programación estructurada y de los manejadores de bases de datos, y

- la cobertura de alguna necesidad real, reforzando con ello el objetivo de cualquier trabajo ingenieril, esto es, ser de utilidad para la sociedad

Investigando las distintas actividades que requerían de evaluaciones rápidas y eficientes, tradicionalmente basadas en las opiniones de los expertos, se planteó el objetivo de diseñar y construir un sistema experto para auxiliar el proceso de diagnóstico neurológico.

Al adentrarse en el tema, se vio que el proceso del diagnóstico neurológico consta de varias etapas consecutivas, cada una de las cuales va acercando al médico al conocimiento de las causas de la enfermedad.

Es por ello que se decidió hacer un diseño modular, en el que cada módulo representa una de las etapas del diagnóstico. En esta tesis se presenta el diseño y la construcción del primero de ellos, que consiste en identificar la región o regiones del sistema nervioso donde puede estar localizada la lesión que está provocando la enfermedad neurológica.

Para lograrlo, se tomó en cuenta para el diseño del sistema experto la aproximación del médico como la estrategia de búsqueda, evaluación y solución del problema. Este enfoque tiene la ventaja de permitir el diseño de estructuras de conocimiento adecuadas al tipo de información que se va a manejar, explotando al máximo sus relaciones y aprovechando en general las particularidades del diagnóstico neurológico.

El reporte del trabajo está organizado en la presente introducción y tres capítulos principales:

1. INTRODUCCION

El primero de los capítulos corresponde a la revisión bibliográfica en el área de la Inteligencia Artificial, en el que se hace una síntesis del desarrollo histórico del área y sus ramas primordiales de investigación. Posteriormente, se analiza con mayor detalle la literatura correspondiente a la Ingeniería de Conocimiento, en particular la de los Sistemas Expertos. Los aspectos que se describen se refieren a su clasificación, a las arquitecturas y técnicas usadas, así como la descripción de algunos sistemas expertos que se han desarrollado a lo largo de la historia y la explicación de sus principales particularidades. Por último, para terminar este capítulo, se ven las características que deben de poseer los sistemas expertos para ser considerados de calidad.

El segundo capítulo se refiere al diseño y la construcción del sistema experto que nos ocupa. Primero, se da una breve visión a la problemática del diagnóstico neurológico y de sus particularidades, posteriormente se presenta el diseño del sistema completo, así como de cada una de las partes que lo componen, y, por último, se describe el proceso de implantación del sistema.

En el último capítulo se presentan las conclusiones del trabajo, en donde se evalúa el sistema tanto del punto de vista de la Inteligencia Artificial y de la Ingeniería de Computación, como de su aplicabilidad en el área de la Neurología. Además, se mencionan las futuras etapas y posibles mejoras para el sistema actual.

Finalmente, se anexan las Referencias, la Bibliografía, las Figuras y un Suplemento en el que se presentan sesiones ejemplo que ilustran el funcionamiento de las diferentes partes que componen al sistema. En la Bibliografía se mencionan libros que pueden ampliar algunos temas descritos en esta tesis y que no fueron referenciados en el texto.

II. REVISIÓN BIBLIOGRÁFICA EN EL ÁREA DE LA INTELIGENCIA ARTIFICIAL

En este capítulo se presenta una breve revisión de los antecedentes de la Inteligencia Artificial y sus principales ramas de investigación. Posteriormente, se analiza con mayor profundidad la rama de los sistemas expertos, en donde se habla de su clasificación, las técnicas y las arquitecturas usadas, así mismo, se describen algunos de los sistemas expertos que se han desarrollado y se explican sus principales características. Por último y para terminar este capítulo, se revisan las características que debe de poseer un sistema experto para que pueda ser considerado de calidad.

1.-Antecedentes

A pesar de que existen indicios sobre el interés en la Inteligencia Artificial antes del siglo XX, tales como los mitos griegos sobre Talos y Galatea, los homunculos de Paracelso, el Golem del rabino Juda Ben Loew y los intentos de Charles Babbage de construir una máquina que pudiera realizar juegos como el ajedrez (*Credulid, H M y Gottesman, R. 1978*), podemos considerarla como un producto de este siglo. Esto, debido a que con anterioridad no se habían presentado resultados fehacientes, ni se habían desarrollado estudios sistemáticos.

La Inteligencia Artificial tiene sus raíces en la Cibernética, por lo que una fecha importante, que marca su inicio formal, es el año de 1943, en el que se publican tres artículos que influyeron notablemente en su posterior desarrollo. En el primero de ellos, Rosenblueth, Wiener y Bigelow (*Rosenblueth, A., et al., 1943*) sugirieron distintas formas de conferirles a las máquinas fines y propósitos, es decir, de volverlas teleológicas. El segundo escrito por McCulloch y Pitts (*McCulloch, W S y Pitts, W S., 1943*), manifiesta el modo en el que las máquinas pueden emplear los conceptos de la lógica y demuestra que cualquier ley de entrada-salida puede modelarse a través de una red neural. Finalmente, en el tercer artículo, Craik (*Craik, L. K.J.W., 1943*) propone el empleo de modelos y analogías en la resolución automática de problemas.

Las ideas contenidas en estos tres artículos, así como en el libro "Cibernética" publicado en 1948 por Wiener, siguieron siendo especulaciones teóricas hasta mediados de los años cincuenta, en los que el nivel del poder de cómputo de las máquinas existentes llegó a ser adecuado para poder implantar y emplear algunas de ellas.

1956 es considerado como el año en el que nace la Inteligencia Artificial como una rama independiente de la Cibernética, ya que éste es en el que se reúnen en Dartmouth College Samuel, McCarthy, Minsky, Selfridge, Newell, Shaw, Simon y Bernstein, para discutir la posibilidad de construir máquinas inteligentes, sin basarse en las ideas de la Cibernética.

A partir de este momento se separa el enfoque estructural, basado en los conceptos de la Cibernética, del funcional, basado en las ideas de Minsky y los otros asistentes a la reunión del Dartmouth, y se crean dos corrientes de investigación dentro de la Inteligencia Artificial.

El primer enfoque pretende emular el cerebro humano, incluyendo su estructura física. Este enfoque supone que el comportamiento animal, y por lo tanto el humano se rige por medio de la estructura física del cerebro y del sistema nervioso, por lo que trata de conocerla para poder simular cada una de sus actividades. El segundo busca crear sistemas que tengan un comportamiento similar al de una persona considerada inteligente. Éste no trata de reproducir la estructura física del cerebro, sino sus funciones. Considera que es suficiente conocer el proceso del comportamiento humano para poder simularlo, sin tener que copiar la estructura del cerebro o del sistema nervioso.

Otras ciencias también han tenido esta dualidad de opiniones. Como ejemplo se menciona la aeronáutica, la cual durante mucho tiempo trató de reproducir la estructura de las aves para el diseño de las máquinas voladoras. Este enfoque no tuvo muchos avances, ya que no existe a la fecha ningún aeroplano que se impulse batiendo las alas. Similarmente, el actual avión no se basó en ningún ser vivo que vuele con hélice o turbina. El otro enfoque fue más productivo, ya que con él se comprendió la función de volar, y

posteriormente, utilizando estructuras completamente nuevas, ésta se realizó. Otro campo que tuvo un desarrollo similar fue el automotriz, ya que tampoco se inspiró en criaturas vivas con ruedas ni se derivó en la invención de vehículos con patas. En la Inteligencia Artificial, por la brevedad de su desarrollo, todavía conviven las dos corrientes.

2.-Áreas de Investigación

Las principales líneas de investigación de la Inteligencia Artificial son (Lara Rosano, F. y Gelman Kipnis, G., 1989):

- *Solución heurística de problemas* - Los esfuerzos en este campo están dirigidos a investigar las formas de búsqueda de soluciones que permitan encontrar un estado final a un problema, en un espacio de búsqueda muy complejo o prácticamente infinito.

Esta área de investigación surge debido a la existencia de problemas reales cuya representación gráfica, en forma arbórescente, denota una complejidad y una extensión enormes, las cuales impiden llegar a la solución en un tiempo adecuado utilizando búsquedas exhaustivas.

La heurística, que significa "arte de inventar" (del griego "euristo" - inventar, hallar) (*Real Academia Española, 1970*) es la metodología que se usa para definir un camino más directo hacia la solución en base a algunos factores existentes, eliminando otros caminos donde la evaluación de los factores mencionados produce probabilidades más bajas.

- *Representación del conocimiento* - Tiene por objeto investigar las formas más apropiadas para representar el conocimiento, cierto e incierto, mediante bases de conocimiento, de manera que pueda utilizarse eficientemente en la solución de problemas

Dentro de este campo, la eficiencia está cuantificada según el espacio físico que ocupa el conocimiento, el cual debe tender al mínimo posible sin que esto interfiera negativamente con los mecanismos de búsqueda que se apliquen a él.

Esta área está ligada con muchas otras dentro de la Inteligencia Artificial, pero sobre todo, con la de los sistemas expertos, puesto que uno de los puntos medulares de estos últimos, determinante de su eficacia y rendimiento, es precisamente una adecuada representación del conocimiento. En algunos sistemas, las formas de representación no se limitan a sólo una, sino que se conjugan varias de ellas para aprovechar al máximo las peculiaridades de las estructuras dentro de las situaciones que se presentan.

- **Sistemas expertos** - Pretende crear sistemas computarizados que proporcionen soluciones a problemas en temas específicos, reproduciendo la pericia de un experto humano.

En la etapa actual, los sistemas expertos pretenden apoyar al experto, más que suplirlo. Su utilidad es especialmente apreciada en aquellos campos en los que la complejidad y extensión del dominio de trabajo dificultan llegar a un espacio final de soluciones, siendo además particularmente ventajoso en los casos en que la obtención de la solución óptima compete directamente con el tiempo a tomarla, creando períodos críticos muy cortos en los cuales se debe realizar una crucial toma de decisiones.

Es el área de la Inteligencia Artificial que más impacto ha tenido fuera del mundo académico, debido a sus abundantes posibilidades de aplicación. En la actualidad muchas ramas industriales están incorporando este tipo de sistemas a sus procesos productivos y de toma de decisiones. Por ejemplo, en los E. E. U. U., la autorización de una compra con la tarjeta American Express se realiza utilizando un sistema experto que evalúa la historia individual de un cliente dado y decide si le autoriza o no el monto solicitado.

- **Redes neuronales** - Su objetivo es descubrir la forma en que se organiza el cerebro de los organismos superiores y proponer mecanismos para la solución de problemas, utilizando procesadores elementales interconectados a la manera de redes nerviosas.

Este campo se inserta dentro del enfoque estructural de la Inteligencia Artificial, apoyándose en áreas de investigación tales como la biogenética, la fisiología celular y la bioquímica.

- *Recuperación inteligente de la información* - Se refiere a la obtención de información por medio del razonamiento deductivo a partir de los datos contenidos en bases de conocimiento y bases de datos.

Se diferencia de las búsquedas tradicionales en que sus entradas de información son incompletas o ambiguas, pretendiendo un espacio de resultados lo más completo posible a partir del mínimo de información. Sus principales aplicaciones se encuentran en la investigación criminalística, localización de datos bibliográficos, en investigaciones históricas y estereois.

- *Demostración automática de teoremas* - Se enfoca en la derivación y la demostración de todas las implicaciones relevantes a partir de una base axiomática dada.

Es particularmente útil en el trabajo de investigación en todos los campos de las matemáticas, en donde se empleó desde las etapas más tempranas de la Inteligencia Artificial debido a la estructura lógica que presentan este tipo de demostraciones.

- **Percepción y reconocimiento de patrones** - Al principio, la investigación en este campo estaba dedicada al diseño de sistemas que pudieran reconocer y clasificar imágenes bidimensionales, como por ejemplo, caracteres alfanuméricos.

Más adelante, el interés se ha concentrado en la creación de sistemas capaces de identificar patrones visuales tridimensionales y patrones auditivos.

Esta área está íntimamente relacionada con la robótica, ya que en la actualidad se está tratando de diseñar robots con una programación flexible que les permita interpretar estímulos visuales y auditivos para una oportuna modificación de su secuencia de actividades. Esto es muy importante en las líneas de producción modernas, donde cada día se está tratando de volverlas más flexibles, así como de disminuir el tiempo de programación entre un producto y otro.

- **Aprendizaje** - Investiga la forma en la que los organismos superiores aprenden de la experiencia y en la que este aprendizaje se estructura en sus sistemas nerviosos, con el fin de diseñar artefactos capaces de aprender de la experiencia.

Esta área se relaciona mucho con la robótica y con los sistemas expertos, ya que en ambas es muy importante el desarrollo de sistemas flexibles que puedan mejorar su desempeño a lo largo del tiempo, tomando en cuenta sus errores y mejorando con ello su programación inicial.

- *Procesamiento del lenguaje natural y la traducción automática.* - Este campo de la Inteligencia Artificial trata de analizar y sintetizar el lenguaje natural con el fin de crear sistemas capaces de comprender y expresarse en un lenguaje natural y hacer traducciones de un idioma a otros. Puede ser muy útil tanto en la interpretación simultánea como en la traducción de material impreso, disminuyendo los tiempos y los costos.
- *Robótica.* - Tiene como objetivo crear artefactos que puedan planear y ejecutar movimientos precisos de sus miembros y articulaciones, de acuerdo con la información del medio ambiente, captada mediante sensores apropiados, para cumplir con los objetivos dados. Los robots que no poseen sensores y son preprogramados, no se consideran parte de la Inteligencia Artificial. Conforme se ha desarrollado esta área, los robots se han vuelto más flexibles respecto a su programación y han logrado una mejor respuesta a los estímulos imprevistos del medio ambiente. Como ya se mencionó con anterioridad, la robótica está muy ligada con el aprendizaje, tratando de diseñar robots que puedan modificar su programación, no sólo a través de los estímulos externos, sino también a través del tiempo, aprovechando los errores cometidos.
- *Enseñanza inteligente.* - Este campo tiene como objetivo primordial el desarrollo de "tutores" automáticos inteligentes que apoyen la enseñanza, adaptándose a las necesidades individuales de cada uno de los estudiantes.

Esta área ha sido una de las más aplicadas después de la de los sistemas expertos, dadas algunas de sus ventajas frente a la enseñanza tradicional con un maestro humano, entre las que se cuentan: minimización de sentimientos de frustración por parte del alumno, atención personalizada, eliminación de situaciones de tensión y posibilidad de generar numerosas repeticiones para mayor comprensión. Algunas veces por su cercanía con el área de los sistemas expertos, no se hace la distinción y los "tutores" automáticos son llamados sistemas expertos en educación.

Como ya se mencionó al principio de este capítulo, a continuación se verá con mayor detalle el área de los sistemas expertos y el de la ingeniería de conocimiento.

3.-Sistemas Expertos

Para comprender a fondo el área de los sistemas expertos, en esta parte se analiza su clasificación dependiendo del tipo de tarea que realiza. Posteriormente, se habla de la arquitectura de lo que, al no existir un sistema estándar, se puede llamar un "sistema experto tipo", así como de las principales técnicas que se utilizan en su desarrollo. La mayoría de los sistemas expertos se basan en una u otra forma en este "sistema tipo", a pesar de que no existen dos sistemas iguales. Después se ven algunos sistemas existentes y se describen sus peculiaridades, diferencias y ventajas. Por último, se mencionan aquellas características que pueden definir a un sistema experto de calidad. Éstas se usarán en el capítulo de conclusiones para evaluar el sistema.

3.1.-Clasificación

La mayoría de los sistemas expertos pueden agruparse de acuerdo a la siguiente clasificación (Lara Rovano, F y Gelman Kipnis, G., 1989).

- *De interpretación* - Este tipo de sistemas infiere significados de estados y situaciones a partir de los datos. Esta categoría incluye la comprensión de mensajes, análisis de imágenes, interpretación de señales, etc.
- *De predicción* - Esta clase de sistemas infiere las probables consecuencias de una situación dada. Esta categoría incluye las predicciones de clima, del tráfico, las predicciones demográficas, etc.
- *De diagnóstico* - Son aquellos sistemas que infieren malfuncionamientos de un sistema a partir de signos y síntomas. Incluyen el diagnóstico médico, el electrónico, el mecánico, etc. La mayoría de los sistemas comerciales están dentro de este rubro.
- *De diseño* - Estos sistemas desarrollan las configuraciones que resuelven un problema satisfaciendo sus restricciones. Incluyen diseños de edificios, de circuitos, etc.

- **De planeación:** Esta categoría permite diseñar acciones interrelacionadas para lograr algún objetivo, utilizando ciertos recursos y sujetándose a ciertas restricciones. Incluye el diseño de rutas, el de proyectos, el de experimentos, etc.
- **De monitoreo:** Esta clase analiza observaciones del comportamiento de un sistema para detectar posibles desviaciones de la norma y tomar eventuales medidas correctivas con toda oportunidad. Estos sistemas se aplican en las plantas nucleares, en el tráfico aéreo, etc.
- **De depuración:** Son aquellos sistemas que prescriben los remedios para los casos en los que existen condiciones anómalas. Estos sistemas se aplican en la depuración de programas de computadora y en la detección de posibles causas de falla de equipo de toda índole.
- **De reparación:** Estos sistemas desarrollan y ejecutan planes para corregir fallas de toda índole.
- **De instrucción:** Son los sistemas que ayudan en el aprendizaje, diagnosticando la situación de cada estudiante y proporcionando a cada quien la orientación y la información pertinentes para avanzar óptimamente en el conocimiento. Estos sistemas algunas veces se clasifican dentro de la rama de enseñanza inteligente y no dentro de los sistemas expertos.

- *De control*. - Esta categoría tiene como tarea el cumplimiento de un objetivo dado, a través de interpretar la situación actual, compararla con la norma, detectar desviaciones, formular el plan de corrección y monitorear su ejecución.

3.2.-Arquitectura

A pesar de que no existe un sistema experto estándar, por la variedad de técnicas que se usan para su diseño, la mayoría de los sistemas expertos tienen una arquitectura parecida, basada primordialmente en el primer sistema experto exitoso -MYCIN (Shortliffe, E.H., 1976), que podría resumirse en los siguientes bloques principales.

- *La Base de Conocimiento*. - Es el módulo encargado de guardar el conocimiento del sistema. Este conocimiento, que usualmente está expresado en una serie de hechos conocidos, se plasma en una base de hechos y en un conjunto de relaciones cognoscitivas que configuran la base de relaciones. No podría existir un sistema experto que no tuviera esta parte, ya que el conocimiento es lo que lo distingue de un programa algorítmico normal.
- *La Máquina de Inferencias*. - Es la parte del sistema que, con fundamento en el conocimiento almacenado en la Base de Conocimiento, infiere nuevos hechos y relaciones que ayudan a resolver el problema planteado, en el caso de que exista una solución en este entorno. Existen diferentes algoritmos y

heurísticas para su construcción, ya que de ellos depende su eficiencia y eficacia. Una Máquina de Inferencias óptima tiene que encontrar el resultado en el menor tiempo posible.

- *El Sistema de Adquisición de Conocimiento* - Es el bloque del sistema experto que se encarga de captar y transferir el conocimiento de uno o varios expertos a la Base de Conocimiento. La facilidad de su uso ayuda a lograr bases de conocimiento más completas y con un menor número de inconsistencias.
- *La Interfaz con Usuario* - Es la parte encargada de permitir la comunicación entre el usuario y los demás bloques del sistema experto. Éste es el bloque que utiliza la Máquina de Inferencias para retroalimentarse durante la consulta, así como para proporcionarle al usuario los resultados, tanto los intermedios como los finales.
- *La Interfaz Explicativa* - Es el módulo que permite cuestionar al Sistema para que le pueda explicar al usuario la forma en la que obtuvo sus conclusiones o por qué hace determinadas preguntas al usuario. Esto es muy importante, ya que el usuario le tiene una mayor confianza a los resultados obtenidos por el sistema si comprende el procedimiento que guió al sistema hacia éstos. Otro aspecto a tomar en consideración, es la posible mejora que se obtiene al refinar la información contenida en la Base de Conocimiento, como resultado de la repetida evaluación por parte de los expertos, de las respuestas emitidas por esta interfaz.

3.3.-Técnicas

A continuación se mencionan las técnicas usadas en los diversos bloques de un sistema experto y sus divisiones. Estas técnicas pueden dividirse en cuatro niveles fundamentales:

- En el nivel básico están las técnicas que son fundamentales en todas las aplicaciones: cálculo proposicional, programación simbólica, búsquedas y heurística; es decir, todas aquellas que sirven para integrar la Máquina de Inferencias.
- El segundo nivel consiste de las diversas formas de representación de conocimiento: reglas lógicas y coeficientes de certeza, "frames" y objetos, redes semánticas, redes conceptuales, redes lógicas, pizarras, etc., que constituyen el fundamento de la Base de Conocimiento.
- En el tercer nivel se encuentran: el control y la organización, los resultados intermedios, la justificación y la optimación. Esta última es muy importante, ya que la mayoría de los sistemas expertos son interactivos con el usuario, por lo que la rapidez de procesamiento cobra un valor primordial.
- Por último, en el cuarto nivel se encuentran las capacidades de comunicación del sistema experto. Éste se comunica con los diseñadores del mismo, los expertos, los usuarios, la base de conocimiento, e inclusive en algunos casos

con otros sistemas en la misma o en otras computadoras. El lenguaje que el sistema usa para ello, debe ser diferente para cada uno de ellos. En este punto, el término "lenguaje" no hace referencia a uno de programación, sino más bien a una interfaz de comunicación.

3.4.-Algunos sistemas y sus características

En este subcapítulo se hablará de los principales sistemas expertos, describiendo sus arquitecturas y particularidades.

a.-MYCIN

Este sistema fue creado por Shortliffe y colaboradores en Stanford en 1976 (Shortliffe, E.H., 1976), (*The Handbook of Artificial Intelligence*, 1982). Fue diseñado para dar consultas en el diagnóstico y la terapia de las enfermedades infecciosas. Este sistema sirvió de base a muchos de los actuales sistemas expertos y en él está basado el "sistema tipo" cuya arquitectura se describió con anterioridad.

El conocimiento médico en este sistema está codificado en forma de reglas de producción. Esta forma de representación se fundamenta en dos estructuras lógicas cuya definición formal es la siguiente:

- *El juicio enunciativo*, que es una estructura lógica compuesta por un concepto sujeto y un concepto predicado, y que expresa un supuesto de partida. Este tipo de juicio recibe el nombre de "hecho". Los hechos son en general juicios particulares que se refieren a casos específicos y que, por lo tanto, varían de problema a problema. Ej: "La inflación es alta".
- *El juicio condicional*, que es una estructura lógica formada por una parte condicional o hipotética, constituida por la conjunción de uno o más juicios enunciativos, y una parte concluyente, formada por un juicio enunciativo que es implicado por la parte condicional. Este tipo de juicio también recibe el nombre de "reglas lógicas". Estas reglas expresan relaciones entre juicios y son de una naturaleza más permanente que los "hechos". Ej.: Si la inflación es alta y los salarios no aumentan, entonces el poder adquisitivo de la población disminuye".

El conjunto de reglas lógicas y de hechos forman la Base de Conocimiento.

Hay que hacer notar que tanto las reglas como los hechos pueden ser inexactos. Esta inexactitud puede ser representada en una escala de -1.0 a 1.0 , donde -1.0 corresponde a una seguridad total sobre la falsedad del juicio, mientras que 1.0 a la seguridad absoluta de la veracidad. Dependiendo de la mayor o menor seguridad, una regla puede tener cualquier valor entre estos dos extremos. A estos valores, Shortliffe y Buchanan los llaman coeficientes de certidumbre (CC). Shortliffe eligió los CC en vez de medidas estadísticas estándares debido a que las

experiencias con los médicos demostraron que su forma de razonar no se ajusta a estas últimas. El manejo de CC permite múltiples resultados con diferente certidumbre.

En la Máquina de Inferencias del MYCIN, las reglas son invocadas utilizando "backward-chaining", lo que resulta en una búsqueda exhaustiva de tipo "depth-first". (Rich, E., 1985)

El método de razonamiento "backward-chaining" es aquel en el que las búsquedas se realizan a partir de los estados meta hacia los estados iniciales.

En el método de búsqueda "depth-first", ésta se realiza siguiendo una rama del árbol de búsquedas, hasta la obtención del resultado o hasta haber llegado al fin de la rama. En este último caso, el método se regresa y continúa la búsqueda en una rama nueva. El proceso se repite hasta encontrar el resultado o haber recorrido todo el árbol.

Si después de revisar todas las reglas relevantes, el peso total de la evidencia de una hipótesis dada está localizado entre -0.2 y 0.2 (valores empíricos), ésta se sigue considerando desconocida.

Como ya se mencionó, este sistema fue de los primeros desarrollados, y por ende, una gran mayoría de los sistemas subsecuentes se basan en gran medida en su arquitectura.

b. CENTAUR

Este sistema también fue desarrollado en Stanford, pero por Aikins (*The Handbook of Artificial Intelligence*, 1982). Sirve para la interpretación de los datos provenientes de las pruebas de funciones pulmonares. La estructura de su base de conocimiento, igual que la de MYCIN, está formada por reglas de producción, pero además, contiene los llamados "frames".

La representación del conocimiento mediante "frames" (Minsky, M., 1975) se fundamenta en que las unidades del discurso lógico, en la lógica de predicados, no son juicios, sino conceptos, esto es, las representaciones lógicas de los objetos. De esta manera, agregando a una representación por reglas, un segundo tipo de representación por concepto, se obtienen formas muy eficientes y flexibles de representar el conocimiento de un experto.

Un "frame" se puede definir como un paquete de información que contiene la relevante de un objeto, estructurada en un formato preciso. En el caso de CENTAUR, cada "frame" representa alguna enfermedad pulmonar. Un "frame" está formado por un conjunto de "slots" (ranuras) que contienen las cualidades del objeto representado por el "frame", así como sus valores específicos concretos.

Un "frame", por ejemplo, el de "auto", consta de los siguientes elementos:

- El nombre del objeto (auto)

- El o los géneros próximos del objeto en una estructura lógica definida. Este género próximo del objeto se llama también "padre" del objeto (vehículo terrestre)
- Las cualidades del objeto o rasuras, y los valores que constituyen su connotación. (ruedas = 4, propulsión = motor de gasolina, cupo = 5 pasajeros, etc.)
- Condiciones indispensables para acceder o modificar la información del objeto.

Para la obtención del diagnóstico, la Máquina de Inferencias sugiere inicialmente, una hipótesis en base a los datos disponibles y los frames existentes. Después, conforme va adquiriendo más información y usando para ello las reglas de producción, selecciona una hipótesis más específica.

Como se puede observar, la arquitectura de este sistema está basada en MYCIN, pero al tener una Base de Conocimiento con estructuras adicionales como son los "frames", es más eficiente, desde el punto de vista de la computación, en su proceso de diagnóstico, ya que la búsqueda se enfoca a espacios más reducidos, disminuyendo con ello el tiempo de solución. Esto se logra haciendo primero una búsqueda rápida a través de los "frames" y una vez elegido el o los factibles se realiza una búsqueda más fina, usando para ello las reglas de producción, tal y como lo hace MYCIN.

c.-PROSPECTOR

Es un sistema experto de consulta entocado a la extracción de los minerales. Fue desarrollado en SRI Internacional por Duda, Garschtig y Hart en 1979 (*Duda, R.O., et al., 1979*), (*The Handbook of Artificial Intelligence, 1982*).

La estructura de datos para representar el conocimiento geológico que usa es la llamada *red de inferencia*. Una red de inferencia es un conjunto de proposiciones (nodos) interconectados mediante funciones lógicas que expresan las dependencias de verdad de los consecuentes en términos de sus antecedentes inmediatos. Según los autores del sistema, esta representación es la que mejor se ajusta a las particularidades del conocimiento geológico, por lo que con ello se disminuye el tamaño físico de la Base de Conocimiento y se aplica la heurística de búsqueda.

Una proposición "p" es un antecedente lógico de una proposición "q" (llamada consecuente lógico) en un discurso lógico dado, si cierto valor de verdad de "p" es una condición necesaria para que "q" sea verdadera.

Los antecedentes lógicos inmediatos, con respecto a un consecuente dado, son aquellos que lo determinan en forma directa, sin necesidad de ningún otro antecedente que sirva de intermediario.

Una función lógica es la que expresa la dependencia de verdad de un consecuente en términos de los valores de verdad de sus antecedentes lógicos inmediatos. Esta dependencia se expresa mediante el uso de los operadores lógicos como la conjunción "Y", la disyunción "O", la negación "NO", etc

Por ejemplo, supóngase las siguientes proposiciones

q_1 = "Este edificio está mal construido"

q_2 = "Próximamente habrá un sismo intenso"

q_3 = "Este edificio va a caerse"

Estas proposiciones pueden relacionarse a través de funciones lógicas, de tal forma que q_3 sea una conclusión de la conjunción de q_1 y q_2 . Esto es:

"Si este edificio está mal construido y próximamente habrá un sismo intenso, entonces este edificio va a caerse"

La Máquina de Inferencias de este sistema usa una forma de razonamiento mixto. Primeramente, le permite al usuario introducir todos los datos que éste posee. Posteriormente, en base a estos datos y la Base de Conocimiento, el sistema elige el mejor modelo para su posterior comprobación. A partir de este momento, el sistema comienza con "backward-chaining", similar al que utiliza MYCIN.

d.-INTERNIST

Es un sistema de consulta en el área de la medicina interna desarrollado por H.E. Pople, J.D. Myers y R.A. Miller en la Universidad de Pittsburgh (Miller, R.A., et al., 1982), (*The Handbook of Artificial Intelligence*, 1982). Se le presenta al sistema una lista de manifestaciones de la enfermedad (síntomas, signos, pruebas de laboratorio, etc.) y éste intenta formar un diagnóstico. Este diagnóstico consta de una lista de factibles enfermedades.

Una de las mayores metas en el desarrollo de INTERNIST fue la de poder modelar el razonamiento que siguen los médicos durante un diagnóstico.

La base de conocimiento de este sistema experto está organizada en forma de un "árbol de enfermedades". El primer nivel clasifica las enfermedades por órganos. Los siguientes niveles del árbol van refinando las enfermedades hasta llegar al último nivel que define enfermedades particulares.

Las enfermedades y sus manifestaciones se relacionan de dos formas principales: una manifestación evoca una enfermedad y una enfermedad puede manifestar algunos signos y síntomas. Cada una de las relaciones tiene asignada una probabilidad. Esta probabilidad puede tomar valores entre 0 y 5, donde 0 significa que no se puede obtener ninguna conclusión a partir de esta enfermedad y esta manifestación, mientras que 5 significa que la manifestación siempre está asociada con esta enfermedad. A pesar de que los autores le llaman probabilidad, nosotros

creemos que es más apropiado utilizar los mismos términos que Shortliffe, es decir, coeficientes de certidumbre. A diferencia de MYCIN, los coeficientes varían entre 0 y 5, en vez de -1 y 1.

La Máquina de Inferencias sigue el siguiente procedimiento para la obtención del resultado. Primeramente, se le introduce al sistema la lista de las manifestaciones que presenta el paciente. Cada manifestación evoca uno o más nodos del árbol. Por cada uno de los nodos evocados, la Máquina de Inferencias crea un modelo. Este modelo consiste de cuatro listas:

- Manifestaciones teóricas de la enfermedad.
- Manifestaciones existentes que no pueden ser explicadas por esta enfermedad.
- Manifestaciones que deberían de existir en el caso de esta enfermedad y que aún no han sido observadas en el paciente.
- Manifestaciones consistentes con la enfermedad que han sido observadas en el paciente.

Cada modelo es evaluado, y dependiendo del número de manifestaciones que logra explicar y del número que no logra explicar, le es asignado un valor. La estrategia posterior depende del número de hipótesis existentes en un momento dado.

En el caso de que existan más de cuatro hipótesis simultáneas, el sistema trata de eliminar el mayor número posible. Las preguntas que hace son aquellas cuyas respuestas tienen el mayor peso para cada uno de los modelos. En el caso de que no existan estas manifestaciones, el modelo dado es eliminado.

Si el número de hipótesis oscila entre dos y cuatro, la Máquina de Inferencias intenta discriminar entre ellas. Para ello se eligen preguntas sobre las manifestaciones que incrementen fuertemente las probabilidades de una enfermedad, disminuyendo al mismo tiempo las de la otra.

En el caso de que únicamente existe una sola hipótesis, el sistema elige las preguntas que comprueben o rechacen ésta.

Algunas veces, por insuficiencia de información, el sistema no puede determinar un nodo final (enfermedad concreta), y proporciona un nodo intermedio, es decir, una respuesta más general.

El principal problema de este sistema es el tiempo de respuesta, ya que inicialmente no se enfoca en ninguna parte del árbol en particular, sino trata de revisarlo todo. Es por esto que tarda un tiempo considerable en desechar algunas ramas no factibles. Si se utilizaran algunas estructuras lógicas, como por ejemplo los "frames", o si se diseñara alguna heurística para enfocar la búsqueda, este inconveniente podría reducirse.

Con esta breve lista de sistemas existentes, se trató de mostrar la riqueza de arquitecturas de las Bases de Conocimiento y de las heurísticas.

3.5.-Requisitos de un sistema experto de calidad

Por último, para concluir con este capítulo de revisión bibliográfica, se describen las características que debe poseer un sistema experto para ser considerado un buen sistema (Lara Rosario, F y Gelman Kipius, G., 1989):

- *Debe ser útil*: El sistema experto debe ser desarrollado para cubrir una necesidad específica. Esta característica es muy clara, ya que no existe ningún sentido desarrollar algo que no tiene alguna aplicación.
- *Debe ser fácil de usar*: El sistema experto debe ser diseñado de tal forma que hasta un novato en cómputo lo pueda utilizar. Esto se debe a que los sistemas expertos normalmente son utilizados por usuarios con pocos conocimientos en cómputo.
- *Debe ser educativo*: El sistema experto debe de enseñar e incrementar el conocimiento de los no expertos en el área.

- *Debe de poder explicar el resultado.* - El sistema experto debe explicar su razonamiento para convencer al usuario. Con ello, el usuario le tiene una mayor confianza a los resultados, y además puede detectar algunos errores en su funcionamiento y corregir la Base de Conocimiento.
- *Debe de poder responder preguntas.* - El sistema experto debe de poder explicar a un usuario no experto algunos puntos que no le fueran claros. Este inciso es complementario del anterior, ya que no es suficiente la explicación del resultado, sino se tiene una explicación del proceso que llevó a cabo el sistema para producirlo.
- *Debe de ser capaz de aprender.* - El sistema experto, además de responder a las preguntas del usuario, debe de ser capaz de generar las suyas propias para incrementar el conocimiento contenido en la Base de Conocimiento. Además, debe de ser capaz de ir modificando las probabilidades o los factores de certidumbre contenidos en la Base, a través de una retroalimentación. Es decir, cada diagnóstico realizado a lo largo del tiempo debe de ayudar a afinar los siguientes.
- *El conocimiento debe de poder ser fácilmente modificable.* - Es muy importante poder revisar la Base de Conocimiento del sistema para corregir algunos errores, así como para ampliar el conocimiento. Con esto, se le da suficiente flexibilidad al sistema para irse enriqueciendo, y se evita que éste quede obsoleto debido a los vertiginosos avances de la ciencia.

III. ANÁLISIS Y DISEÑO DE SADIN: Sistema Auxiliar para el Diagnóstico Neurológico

Al principio de este capítulo se presenta una breve descripción de la problemática del diagnóstico neurológico, así como de sus particularidades, para definir los requerimientos que deben contemplarse en el diseño de la estructura lógica del sistema experto que nos ocupa. Posteriormente, se presenta la arquitectura global del sistema, y a continuación, se explican detalladamente cada uno de los bloques que la componen. La descripción de estos bloques se presenta en términos de las características funcionales de sus principales módulos. Para finalizar el capítulo, se especifica el proceso de implantación de SADIN en una computadora personal (PC).

I.- Análisis del Diagnóstico Neurológico

El diagnóstico médico sobre cualquier patología de algún subsistema del cuerpo humano requiere de una metodología específica (Le Myer, W., 1980), (Martín Abreu, L., 1978), así como de conocimientos básicos acerca de su integración funcional con el resto del cuerpo. En el caso particular del sistema nervioso se encuentra el ejemplo clínico más claro donde las características anatómicas y fisiológicas se relacionan de manera lógica y directa con funciones de salud y posibles alteraciones durante el padecimiento de alguna enfermedad específica. Es importante señalar que el sistema nervioso se encarga de acoplar funcionalmente a diversos sistemas corporales (e.g., sensoriales y músculo-esquelético), así como de coordinar espacio-temporalmente las interacciones del organismo con el medio ambiente complejo donde se desenvuelve.

La realización del diagnóstico neurológico requiere de una entrevista inicial con el paciente para conocer sus antecedentes y las condiciones de su padecimiento. Entre los principales antecedentes que deben ser explorados durante la entrevista están:

- *Los hereditarios* (e.g., psicosis, enfermedades convulsivantes, fenómenos conversivos, consanguinidad, etc.);
- *los relacionados con las condiciones del nacimiento* (i.e., el parto);
- *los relativos a padecimientos infectocontagiosos* (e.g., fiebre tifoidea, sarampión, paperas, varicela, viruela, rubéola, amibiasis, cisticercosis, etc.);

- *las de los contactos con sustancias neurotóxicas* (e.g., tetracloruro de carbono, DDT, gasolina, plomo, alcohol, drogas alucinógenas, etc.)
- *terapéuticas* (e.g., ácido acetilsalicílico, quinina, anfetaminas, estreptomicina, barbitúricos, etc.).

Otro grupo de antecedentes importantes son los de tipo *iatrogénico* (e.g., neuritis por inyección, bloqueos ganglionares, anestesia lumbar, mielografía, etc.), así como que los *traumatismos* (e.g., craneoencefálicos, del raquis, de los nervios periféricos, etc.) representan un grupo de condiciones fortuitas que pueden contribuir al desarrollo de alteraciones neurológicas.

Una vez que se recopilan los datos referentes a los antecedentes del paciente, el médico debe investigar las posibles condiciones bajo las cuales se haya desarrollado el padecimiento actual. A diferencia de lo que ocurre con otros especialistas, el neurólogo frecuentemente se ve ante la necesidad de valorar vivencias y opiniones, pues los pacientes pueden no haber detectado la manera en que se desarrolló su enfermedad.

La *entrevista clínica* permite identificar el conjunto de síntomas o afecciones que aquejan al paciente, cuya valoración es utilizada por el neurólogo para establecer las primeras hipótesis sobre la posible localización en el sistema nervioso, y el tipo de daño que está padeciendo el enfermo.

Los resultados de la entrevista permiten al médico llevar a cabo una exploración clínica dirigida, durante la cual determina los signos que presenta el paciente. Los neurólogos han desarrollado un plan de procedimiento lógico que debe seguirse rutinariamente, pues es la única manera para establecer la disciplina exploratoria indispensable, y no pasar por alto detalles que pueden resultar de gran importancia en el momento de fundamentar su diagnóstico final.

La primera etapa en el diagnóstico del neurólogo consiste en tratar de identificar la región del sistema nervioso donde se localiza la afección. Los aspectos fundamentales que contribuyen a determinar la región afectada de manera exitosa son los aspectos tróficos y el desempeño del paciente al realizar algunas funciones motoras.

La capacidad motora es la manifestación integrada de aspectos voluntarios y condiciones músculo-esqueléticas que permiten realizar movimientos específicos. Cuando esta capacidad se encuentra limitada se le conoce como *parésia*, mientras que cuando está impedida se le denomina *parálisis*. Ambas condiciones ofrecen características distintivas para diferenciarlas y para localizar la probable ubicación de la lesión que las produce. Algunas de estas afecciones se clasifican, en relación al tono muscular, en:

- *flaccidas*, si el tono muscular está disminuido;
- *espásticas* si está aumentado.

De acuerdo a la cantidad de grupos musculares que se ven afectados, se les denomina como:

- *monoplejias* ;
- *hemiplejias* ;
- *paraplejias* ;
- *cuadriplejias* .

Otro aspecto importante que se toma en cuenta para localizar la lesión, son las alteraciones sensoriales. Estas pueden ser de origen táctil, térmico, de movimiento y de dolor; además, según el sitio en que se originan son superficiales o profundas. Si la reacción al estímulo es exagerada, se le denomina hiperalgesia y si lo que se produce es la disminución de la sensibilidad, se le denomina hiposiestesia.

Con base en todos los signos que presenta un paciente durante la exploración y los síntomas identificados durante la entrevista, el neurólogo determina las regiones del sistema nervioso que tienen más alta probabilidad de estar lesionadas. En algunos casos, todos los signos y síntomas apuntan a una sola región; sin embargo, en otras ocasiones, por diversas razones relacionadas con el tipo de enfermedad o con el propio paciente, no se pueden identificar los datos suficientes para determinar la ubicación exacta de la afección, y sólo se puede generar una lista de probables regiones.

Por último, una vez que se genera una lista de las regiones involucradas, el neurólogo está en posición de elegir pruebas más específicas que le permitan determinar la patología específica que está produciendo la lesión. En cuanto se identifica la enfermedad, el neurólogo puede establecer un tratamiento médico adecuado a la enfermedad, la región afectada y los antecedentes del paciente.

2.- Diseño de SADIN

A continuación se presenta la arquitectura global del sistema y la descripción detallada de cada uno de los bloques que lo componen. Para finalizar el capítulo se describe el proceso de la implantación de SADIN en una computadora personal (PC).

2.1.-Arquitectura global

La arquitectura global de SADIN (Fig. 1) se diseñó considerando los bloques principales de sistemas expertos tradicionales, mencionados en el capítulo anterior. Consecuentemente, la arquitectura global de SADIN es similar a la de MYCIN, sobre algunas particularidades que se explicarán más adelante.

La primera función que debe realizar el sistema experto es la de capturar, de alguna manera, el conocimiento indispensable para su labor. Para ello, se contempla un módulo denominado Interfaz con el Experto (IE), que permite a los expertos en el campo (neurología) introducir la información relevante al sistema para su

incorporación posterior a las bases de conocimiento. Este módulo captura la información en forma estructurada y la guarda en archivos externos para su posterior procesamiento e integración al sistema. Esta forma de proceso "fuera de línea" se eligió para facilitar la recopilación del conocimiento proporcionado por distintos expertos, los cuales generan un archivo independiente que contiene su aportación. De esta manera, la información proporcionada por varios expertos puede ser fácilmente integrada e incorporada a la Base de Conocimiento (BC) de SADIN.

Una vez que se cuenta con información confiable acerca de las posibles lesiones que pueden presentarse en el campo de la neurología, la siguiente función de SADIN es, precisamente, la del procesamiento e integración de la información capturada por el módulo IE a la Base de Conocimiento. Esto se lleva a cabo a través del módulo denominado como Intelecto de Mantenimiento (IM) (Fig. 2), en donde esta información se almacena en estructuras lógicas iguales a las de la BC. La IM también se encarga de verificar que no haya inconsistencias entre la nueva información y la ya almacenada en la BC, permitiendo al encargado de mantenimiento resolver dichas inconsistencias en favor de la información que involucre conocimiento más reciente. La actualización de las reglas de la BC implica que en la IM se generen o recalculen los factores de certidumbre que deben asignarse a cada una de las reglas.

La separación de los dos bloques anteriores es una particularidad de SADIN. Esto se hizo con el propósito de preservar la integridad de la BC, ya que se prevé la figura de un encargado de mantenimiento, quien será responsable de recopilar la información proveniente de los diferentes expertos y de integrarla a la BC. Este especialista debe resolver cualquier inconsistencia que se presente y es la única persona que tiene

acceso directo a la BC para poder modificarla. Es obvio que el papel del mantenedor puede ser realizado por el mismo usuario, quien a la vez puede ser el experto. La separación entre las diferentes interfaces (i.e., IE, IM y IU) es únicamente de índole funcional.

La estructura de datos donde se almacena el conocimiento obtenido a través de la IE, y procesada por la IM, es la denominada como Base de Conocimiento (BC) (Fig. 3). En esta estructura se encuentran las reglas de conocimiento, es decir, los elementos y sus relaciones, así como los factores de certidumbre correspondientes.

El módulo más importante del sistema es el marcado como Máquina de Inferencias (MI) (Fig. 4). Este módulo se encarga de llevar a cabo el proceso de consulta y de proporcionar la lista de las posibles regiones involucradas en el padecimiento del paciente. Además, de la MI depende la rapidez y la exactitud del diagnóstico. Como parte de este bloque se definió una Memoria Temporal (MT), que es una estructura similar a la Base de Conocimiento, en la que se guardan los datos y los resultados que se obtienen en el transcurso de la consulta de un paciente determinado.

Además de estos módulos, SADDN cuenta con otros dos, que son básicamente de apoyo. El primero de ellos es la Interfaz con el Usuario (IU), que se encarga de comunicar a éste con la MI, así como de proporcionar al usuario algunas facilidades extras, como la posibilidad de continuar una consulta interrumpida por fallas en el suministro de la energía eléctrica, la de consultar la historia clínica del paciente, etc.

El segundo bloque es el Explicador (EX), que le proporciona al usuario una explicación detallada sobre la forma de trabajo del sistema, el por qué de las preguntas que SADIN va haciendo al usuario y el cómo se llegó al resultado diagnóstico.

2.2.-Base de Conocimiento (BC)

La BC es el módulo del sistema que contiene el conocimiento proporcionado por los expertos para ser usado por la Máquina de Inferencias durante el proceso de diagnóstico. Este conocimiento consta de elementos que se relacionan entre ellos en forma explícita. En esta sección, se describen estos elementos y sus respectivas relaciones, así como las estructuras lógicas que los albergan.

a.-Elementos

Una vez conocido el método y los procedimientos que utiliza el neurólogo para la realización del diagnóstico, descritos con anterioridad, se identificaron los elementos indispensables para este proceso, que son los siguientes:

- **Nervio:** Es la parte medular del sistema nervioso. Cada nervio está dividido en segmentos y es sujeto a posibles lesiones, esto es, un nervio puede tener lesionado uno o más segmentos.

- **Síntomas:** Un síntoma es la descripción, en palabras del propio paciente, de un malestar físico. La utilidad de un síntoma consiste en ser una guía para dirigir la búsqueda hacia las probables regiones del sistema nervioso lesionadas. Para esta etapa de diagnóstico, las regiones mencionadas corresponden a los nervios y sus segmentos.
- **Signos:** Un signo es la evidencia de un mal funcionamiento de los órganos, aparatos o sistemas del cuerpo humano que el médico encuentra en el paciente después de efectuarle un exámen clínico. La utilidad de un signo reside en que su presencia o ausencia incrementa o disminuye la probabilidad de la existencia de la lesión en una región corporal dada (segmento de nervio). Además, permite guiar la búsqueda de otras probables regiones lesionadas.
- **Maniobra:** Es el procedimiento que debe realizar el médico para determinar la existencia de un signo. La maniobra sirve para guiar al médico cuando desconoce la forma en la que debe proceder para averiguar la presencia o no de un signo dado.

Con base en esta información, para realizar sus funciones, el sistema debe poseer en su BC:

- el conjunto de los segmentos de nervio, cada uno asociado con su respectivo nervio,

- el conjunto de los posibles síntomas,
- el conjunto de los posibles signos, y
- el conjunto de las maniobras

b.-Relaciones

Sin embargo, estos elementos, por sí solos, no son suficientes para generar un diagnóstico neurológico. Es necesario, además, definir las relaciones que guardan entre sí los conjuntos anteriores.

La primera se define como la relación entre el conjunto de los segmentos y el de los síntomas. A cada segmento de cada nervio se le puede asociar un subconjunto de síntomas, ya que en caso de lesión de un segmento específico se ha podido determinar la probabilidad de que se presente uno o más síntomas de este subconjunto. Esto es, se puede establecer que cuando el paciente X tiene lesionado el Segmento J del Nervio K, se presenta uno o más síntomas, del subconjunto:

Síntoma₁, Síntoma₂, ..., Síntoma_M

De la misma forma, cada síntoma apunta a un subconjunto del conjunto de segmentos, lo que significa que si el paciente presenta el Síntoma_i, es factible que tenga una lesión en uno o más segmentos del subconjunto correspondiente:

Segmento₁, Segmento₂, ..., Segmento_N

La segunda relación es la que existe entre el conjunto de los segmentos de nervio y el de los signos. Este tipo de relación se establece de manera análoga a la relación anterior. Esto es, cada segmento está relacionado con un subconjunto del conjunto de signos, de tal forma que para el Segmento_i del Nervio se puede definir un subconjunto de signos correspondientes.

Signo₁, Signo₂, ..., Signo_k

Asimismo, cada signo está relacionado con un subconjunto del conjunto de segmentos, de tal manera que cuando se presenta un Signo_i se puede identificar el subconjunto correspondiente de los segmentos de nervio que pueden estar lesionados.

Segmento₁, Segmento₂, ..., Segmento_i

Para estimar la probabilidad de la lesión de un nervio dado, es importante definir el valor con el que contribuye cada uno de los signos manifestados. Es por ello que la relación anterior se concretiza a través de un conjunto de factores de certidumbre correspondientes a las parejas formadas por un signo y un segmento de nervio. Es decir, la existencia del Signo₁ aporta un valor probabilístico P₁₁ a la probabilidad de existencia de la lesión del Segmento₁ y un valor P₁₂ a la del Segmento₂; el Signo₂ aporta un valor P₂₁ y P₂₂ a las probabilidades de las lesiones de los Segmento₁ y Segmento₂, y así sucesivamente para todos los signos y segmentos. Un aspecto fundamental en el cálculo de los factores de certidumbre es que la suma de los valores de todos los factores de certidumbre del subconjunto de signos asociado a

un segmento de nervio en particular debe ser igual a 100. Esto se debe a que en el diagnóstico neurológico, si el paciente presenta todos los signos que caracterizan la lesión de un segmento de un nervio dado, ésta puede concluirse con el 100% de seguridad.

La tercera relación es la que se establece entre el conjunto de los signos y el de las maniobras. Ésta es una relación uno a uno, ya que a cada signo le corresponde una y sólo una maniobra. De la misma forma, a cada maniobra le corresponde uno y sólo un signo.

c.-Reglas

Después de considerar las distintas formas de representación de conocimiento con que se cuenta en inteligencia artificial (Gevarter, W.M., 1987), (Lara Rosano, F. y Gelman Kipnis, G., 1989), mencionadas en el capítulo anterior, se llegó a la conclusión de utilizar la estructura de reglas de producción para estructurar la BC de SADIN, debido a que era ésta la que mejor se ajustaba al tipo de información existente. La cantidad de elementos y sus relaciones no ameritaban la utilización de estructuras más complejas, como son, por ejemplo, los "frames". Utilizando los elementos y las relaciones anteriormente descritas, se definieron los siguientes tipos de reglas:

- Es factible que el paciente tenga una lesión en el SegmentoJ del NervioK si presenta:

Síntoma1 0

Síntoma2 0

SíntomaM

- El paciente tiene un P_k de probabilidad de tener una lesión en el Segmento del Nervio si presenta el SignoK
- Si el médico no sabe si el paciente presenta o no el SignoL debe ejecutar la ManobraL.

Estos tres tipos de reglas cubren todas las posibilidades en las que los elementos de conocimiento pueden estar integrados.

d.-Archivos

Una vez obtenida la estructura lógica de las reglas, se procedió a diseñar su estructura física. Inicialmente, se contempló la posibilidad de que fuera sólo un archivo el que albergara todas las reglas. Este diseño se rechazó, debido a que producía un archivo muy grande en el que la búsqueda de reglas era demasiado lenta. La alternativa más eficiente resultó ser el uso de bases de datos relacionales (Evenet, C. C., 1986). Esta solución permite la interacción de archivos de un tamaño mínimo necesario en los que se pueden realizar búsquedas rápidas y eficientes, ya que éstas se realizan a través de índices.

Aplicando las tres formas de normalización a las reglas de producción y a los elementos de conocimiento, lo que constituye un procedimiento indicado para el diseño de bases de datos relacionales, se llegó a las siguientes bases de datos (Fig. 3):

Base1: (Nervio, Segmento, ClaveNervio)

Base2: (Signo, Maniobra, ClaveSigno)

Base3: (Síntoma, ClaveSíntoma)

Base4: (ClaveNervio, ClaveSíntoma)

Base5: (ClaveNervio, ClaveSigno, Peso)

En la lista anterior se utilizó una forma común de representación de las bases de datos relacionales, en donde cada elemento contenido entre los paréntesis constituye un campo del registro. La primera de las bases contiene el conjunto de nervios con sus respectivos segmentos, donde cada ClaveNervio identifica una única pareja Nervio-Segmento; la segunda contiene el conjunto de signos y sus respectivas maniobras, donde cada ClaveSigno identifica un único signo y a cada signo le corresponde una única maniobra, la tercera contiene el conjunto de síntomas, donde cada ClaveSíntoma identifica un único síntoma, la cuarta contiene las relaciones entre las parejas Nervio-Segmento y los Síntomas, donde cada ClaveNervio está relacionado con varias ClaveSíntoma y viceversa, además de que

no existen dos parejas Clave-Neuro-Clase Síntoma iguales, y, finalmente, la quinta base contiene las relaciones entre las parejas Neuro-Segmento y los Signos, así como los valores de los factores de certidumbre de cada una de las relaciones. Este archivo tiene la misma restricción observada en la base anterior.

El uso de las claves se adoptó para evitar la utilización de nombres completos en las bases de datos y evitar, de esta manera, el aumento del tamaño de los archivos y agilizar los procesos de búsqueda.

2.3. Máquina de Diferencias (MD)

La Máquina de Diferencias (MD) es la parte del sistema experto que permite resolver el problema planteado, es decir, realizar el diagnóstico. Para lograrlo, se apoya en la información almacenada en la BD, así como en nuevos hechos y relaciones que enriquecen las hipótesis iniciales generadas a partir de los datos que el usuario proporciona.

A través de la MD se obtiene la información referente a casos particulares de diagnóstico neurológico, así como para tomar decisiones y rechazar o consolidar las hipótesis que se van formando durante el proceso de diagnóstico. La MD necesita a la BD, para sustrair las reglas nuevas que se permiten consultar con las hipótesis con el tiempo las subsecuentes, y a la MD para almacenar y consultar los resultados parciales que se obtienen durante el avance del diagnóstico y que permiten intentar nuevas relaciones que conlleven finalmente a la identificación de la lista de las regiones del sistema nervioso que podrían estar involucradas en el padecimiento del paciente.

A continuación se describe detalladamente el proceso que sigue la MI para la realización del diagnóstico, así como la estructura de la MI, que puede considerarse como una parte esencial de la MI.

a.-Proceso

El proceso que sigue la MI de este sistema (Fig. 4), siguiendo los lineamientos planteados desde el inicio del trabajo, es muy parecido al que utiliza el neurólogo. Al principio, trata de generar algunas hipótesis iniciales, lográndolo gracias a los síntomas que apuntan a los nervios que son factibles de estar lesionados. Con esto se evita realizar unas búsquedas extensas en todo el espacio de conocimiento, como en el caso de INTERNIST. Una vez definida el Área inicial de búsqueda, se procede a estimar las probabilidades de lesión de cada uno de los segmentos de los nervios ubicados en ella, a través de la verificación de las manifestaciones de los signos correspondientes. Ya que algunos de los signos apuntan además, a nuevos nervios, el espacio de búsqueda se amplía en forma heurística. Esto permite enfocarse a revisar sólo los probables nervios lesionados, evitando así una revisión total.

Es por ello que inicialmente la MI, a través de la IU, recopila los síntomas de los que se queja el paciente y los guarda en la memoria temporal. La MI, utilizando las relaciones contenidas en la Base 4 de la BC y los síntomas guardados en la memoria temporal, genera entonces el espacio de búsqueda. Éste es una parte de la memoria temporal en la que se guardan los segmentos de los nervios que son factibles de estar lesionados.

A partir de este momento la MI trata de estimar la probabilidad de lesión en cada uno de los nervios localizados en el espacio de búsqueda. El proceso consiste en preguntar al usuario si el paciente presenta los signos relacionados con cada uno de los nervios. Las posibles respuestas a la existencia de un signo dado pueden ser "SI", en cuyo caso el factor de credibilidad asociado a este signo es tomado en cuenta, y "NO" o "NO SE", en cuyo caso éste no es tomado en cuenta.

En ese momento, la MI le presenta al usuario el conjunto de los signos cuya respuesta ha sido "NO SE", y le sugiere revisar las maniobras de estos signos. Una vez presentadas las maniobras, se permite que el usuario pueda modificar su respuesta "NO SE", por cualquiera de las otras dos posibles ("SI" o "NO"), reestimando por ende las probabilidades de lesión.

Cuando la totalidad de elementos del subconjunto de signos relacionados con un segmento de nervio ya ha sido revisada, se traslada este segmento, con su probabilidad, del espacio de búsqueda al espacio de resultados. Este último espacio es la parte de la memoria temporal donde se van guardando los resultados parciales.

Cada vez que se ha revisado el espacio de búsqueda completo, la MI vuelve a preguntar al usuario si el paciente se ha quejado de otros síntomas, tratando de generar nuevas hipótesis. Sin embargo para no complicar el proceso, se revisa cada una de las nuevas hipótesis con las comprobadas anteriormente que se guardan en

el espacio de resultados. Con esto, se evita repetir el proceso con hipótesis ya verificadas, así como solicitar la identificación de los signos manifestados anteriormente.

Conforme a las respuestas y con base en los signos que ha presentado el paciente hasta el momento, la MI vuelve a llenar el espacio de búsqueda con nuevos factibles segmentos. Los signos cuya respuesta ha sido "NO SÍ", también son considerados para la generación de los nuevos segmentos a ocupar el espacio de búsqueda. Es decir, el espacio de búsqueda se llena con segmentos de nervio que están relacionados con los signos que presenta el paciente, o aquellos que aún no se han descartado después de la auscultación. El proceso se repite hasta que la MI ya no puede llenar el espacio de búsqueda por falta de relaciones. Para los procesos anteriores se utilizan las bases cuarta y quinta de la BC.

Una vez concluido este procedimiento, la MI pregunta al usuario si ha detectado en el paciente otros signos que no hayan sido preguntados por el sistema con anterioridad. Esto se incluyó debido a que existe la posibilidad de que el paciente tuviera algunos signos, a los que ni sus síntomas, ni los otros signos apuntaran. Con esto se afina el diagnóstico, ya que el médico tiene la posibilidad de proporcionarle al sistema toda la información que obtiene durante el proceso de auscultación. Si la respuesta es positiva, la MI presenta la lista de los signos almacenados en la BC que aún no se han presentado, e introduce los nuevos signos elegidos por el usuario en el espacio de búsqueda, reiniciando el proceso. Estas acciones se repiten hasta que el usuario no tenga más información que proporcionar, o bien, hasta que se hayan revisado todos los segmentos de nervios contenidos en la BC.

h.- Información contenida en la Memoria Temporal (MT)

De acuerdo con los requisitos del proceso descrito, la información útil a la MI, que debe contener la MT, es la siguiente:

- El conjunto de los síntomas de los que se queja el paciente. Esta información es necesaria para que la MI pueda generar las hipótesis sobre los factibles segmentos de nervios lesionados.
- El conjunto de los segmentos de nervios factibles de estar lesionados (e.g., espacio de búsqueda).
- El conjunto de los segmentos de nervios cuya probabilidad de estar lesionados ya se conoce (e.g., espacio de resultados).
- El conjunto de todos los signos relacionados con los segmentos de nervios contenidos en el espacio de búsqueda cuya respuesta es "SI" o "NO SI". Esta información sirve para generar los nuevos factibles segmentos de nervios.
- El conjunto de los signos relacionados con los segmentos de nervios contenidos tanto en el espacio de búsqueda como en el de resultados. Cada uno de los signos tiene asignada la respuesta "SI", "NO" o "NO SI" correspondiente.

La diferencia entre el conjunto del penúltimo inciso y el del último es que en el del último se localizan todos los signos que han sido verificados durante el proceso de la consulta, mientras que en el del penúltimo, únicamente están los de la última iteración. Una vez que se han utilizado para generar los nuevos segmentos a ocupar el espacio de búsqueda, estos signos se borran

c.-Archivos de la Memoria Temporal

La estructura física de la MI también se concibió como una base de datos relacional. Con esto se agiliza y se unifica el proceso de diagnóstico. Aplicando las reglas de normalización a la información anterior, se llegó a las siguientes bases de datos:

Base temporal 1. (Clave Síntoma)

Base temporal 2. (Clave Nervio)

Base temporal 3. (Clave Nervio, Factor de Certidumbre)

Base temporal 4. (Clave Signo)

Base temporal 5. (Clave Signo, Respuesta)

Igual que en el caso de la BC, el uso de las claves en las estructuras de la MT se adoptó para evitar la utilización de nombres completos. Esta acción se tomó para disminuir el tamaño de los archivos y hacerlos compatibles con los de la BC, ya que los signos son idénticos.

2.4-Interfaz con el Experto (IE)

Es el módulo que permite la comunicación entre el experto y el sistema. A través de ella, en un ambiente amigable, el experto suministra relaciones y hechos del conocimiento que conforman las reglas de la BC.

El programa se compone de un módulo de edición y uno de mantenimiento. El módulo de edición permite generar nuevos archivos y modificar los ya existentes. Por un lado, en el primer caso, la información se recibe dentro de un formato de líneas, siguiendo una secuencia de elementos definida (Fig. 5), esta secuencia tiene como objetivo establecer las relaciones de los elementos entre sí. Durante este proceso, el programa va presentando al usuario elementos que pueden o no ser parte de la estructura cognoscitiva que se está definiendo. El experto entonces decide que elemento, presentado por el programa, elimina de la secuencia, y cuál otro admite en ella. Esto es debido a que el número de síntomas y signos relacionados con un segmento de nervio dado es variable. Por otro lado, el segundo caso permite modificaciones tanto en el contenido de la información como en sus relaciones, insertando elementos

dentro de la secuencia. Siguiendo la filosofía anterior, el programa presenta para insertar sólo aquellos elementos que sean válidos dentro de la sucesión definida, permitiendo después actualizar el archivo.

El módulo de mantenimiento permite acciones de borrado de archivos y despliegue de directorios. Estos procesos se ejecutan únicamente sobre archivos generados por el programa, filtrando documentos externos.

La información que la IE obtiene es guardada en archivos tipo texto que alimentan a la IM para su procesamiento. El diseño de esta interfaz fue realizado a través de numerosas entrevistas con los especialistas (en este caso neurólogos), los cuales determinaron la secuencia de los elementos (estableciendo así sus relaciones), tipo de información a recabar en cada elemento, y número de relaciones asociado a cada elemento, así como las características inherentes al proceso. Esto permitió calcular aproximadamente las dimensiones necesarias de memoria para contener la información.

2.5.-Interfaz de Mantenimiento (IM)

La IM es el módulo del sistema que se encarga de integrar y conservar adecuadamente la información de la BC. Las funciones que realiza son: agregar a la BC los nuevos nervios contenidos en los archivos generados por la IE, modificar y visualizar la BC. A continuación se describirá a detalle cada uno de los módulos de la IM encargados de estas funciones (Fig. 2)

a.- Módulo de adición de nueva información

Dentro del proceso de adición de los nuevos nervios a la BC, el primer paso de la IM es revisar la secuencia de los elementos contenidos en los archivos proporcionados por el experto. Estos elementos son

- Nervio
- Segmento
- Sintoma
- Signo
- Maniobra

y las secuencias permitidas (Fig. 5) son:

- El primer elemento del archivo tiene que ser el Nervio
- Después del Nervio debe existir un Segmento
- Después de un Segmento debe encontrarse un Sintoma
- Después de un Sintoma puede encontrarse otro Sintoma o un Signo

- Después de un Signo puede haber otro Signo, una línea de Manobra o un Segmento.
- Después de una línea de Manobra puede encontrarse otra línea de Manobra, un Signo o un Segmento.

Una vez aceptada la secuencia, la MM verifica si existen inconsistencias entre el archivo a agregar y la BC. Los resultados de este análisis son:

- El nervio a agregar no existe en la BC, por lo que no pueden haber inconsistencias. En este caso el archivo pasa a la siguiente etapa.
- El nervio a agregar existe en la BC, pero el segmento es nuevo. En este caso toda la información de este segmento pasa a la siguiente etapa.
- El nervio a agregar existe en la BC, así como los segmentos correspondientes. Además, sus elementos (síntomas y signos) son idénticos a los existentes en la Base de Conocimiento. En este caso no existen inconsistencias, ya que la información es repetida y el archivo no es agregado a la Base de Conocimiento.
- El nervio a agregar existe en la Base de Conocimiento, así como los segmentos correspondientes, y sus elementos (síntomas y signos) son diferentes a los existentes en la BC. En este caso existen inconsistencias,

por lo que el archivo, junto con la información localizada en la BC correspondiente a este nervio, es guardado en la Bitácora y no es agregado a la BC.

La Bitácora es un archivo en el cual se van guardando los resultados del proceso de adición. En caso de haber inconsistencias, el mantenedor puede comparar la información existente en la BC con la del archivo inconsistente y tomar la decisión.

Una vez terminado el análisis, los archivos que no presentaron inconsistencias ni repeticiones están listos para ser agregados. El siguiente paso es la asignación de los factores de certidumbre a cada una de las parejas Signo-Segmento de cada uno de los nervios. Para la asignación de estos se diseñó el algoritmo que a continuación se describe.

De acuerdo con la teoría de la comunicación (Schwartz, M., 1983) entre menor es la certidumbre de un mensaje, mayor contenido de información lleva. Trasladando esta idea a nuestra caso, podemos decir que entre menos probable es la aparición de un signo, mayor información le proporciona al sistema experto. Esto quiere decir que un signo que está relacionado con pocos nervios le proporciona más información al sistema que uno que está relacionados con muchos. Por ende, el factor de certidumbre es directamente proporcional a la información que aporta, y por lo tanto, inversamente proporcional al número de relaciones dentro de la BC (Papoulis, A., 1984). De aquí se desprende la base del algoritmo que dice:

- Entre menos relaciones tenga un signo, mayor será el valor de su factor de certidumbre.

No se debe de olvidar que la suma de los factores de certidumbre de todos los signos correspondientes a un segmento dado debe ser igual a 100.

Traduciendo lo anterior al lenguaje matemático, podemos decir que:

$$FC_{jk} = C_j \cdot I/N_k \quad F.1.$$

donde

- FC_{jk} es el Factor de Certidumbre de la relación del Signo k con el Segmento j del Nervio i
- C_j es el Coeficiente de normalización del Segmento j del Nervio i
- N_k es el número de relaciones del Signo k dentro de la BC (información)

La normalización, en este caso, se refiere a que la suma de todos los factores de certidumbre de un segmento dado es igual a 100. Esto es,

$$\sum FC_{jk} = 100 \quad F.2.$$

De aquí se puede obtener el valor del coeficiente C_{ij} :

$$\sum FC_{ij} = \sum (C_{ij} \cdot I \cdot N_k)$$

$$\sum FC_{ij} = C_{ij} \cdot \sum (I \cdot N_k)$$

Utilizando la fórmula F.2, se obtiene:

$$C_{ij} \cdot \sum (I \cdot N_k) = 100$$

$$C_{ij} = \frac{100}{\sum (I \cdot N_k)} \quad \text{F.3}$$

Por lo tanto, substituyendo F.3 en F.1, se obtiene:

$$FC_{ij} = \left(\frac{100}{\sum (I \cdot N_k)} \right) \cdot I \cdot N_k$$

de donde reacomodando los términos se logra la fórmula definitiva:

$$FC_{ij} = \frac{100 \cdot I \cdot N_k}{N_k \cdot \sum (I \cdot N_k)}$$

Para ilustrar lo anterior usaremos la fórmula en el siguiente ejemplo.

En la BC existen, entre otros, dos segmentos SI G1 y SI G2 del Nervio 1 y estos están caracterizados por los signos

SEG₁ : SIGNO₁
SIGNO₂
SIGNO₃

SEG₂ : SIGNO₂
SIGNO₄

El SIGNO₁ se repite 5 veces en la BC, es decir, está relacionado con cinco distintos segmentos

El SIGNO₂ se repite 2 veces en la BC

El SIGNO₃ es único

El SIGNO₄ se repite 150 veces en la BC

Por lo tanto, los pesos de cada uno de ellos serán:

$$N_1 = 5$$

$$N_2 = 2$$

$$N_3 = 1$$

$$N_4 = 150$$

$$\sum 1/N_k \text{ del SEG}_1 = 1/5 + 1/2 + 1/1 = 1.7$$

$$\sum 1/N_k \text{ del SEG}_2 = 1/2 + 1/150 = 0.506$$

Aplicando la fórmula se obtiene

$$FC_{111} = \frac{100}{5 * 1,7} = 11,76$$

$$FC_{112} = \frac{100}{2 * 1,7} = 29,41$$

$$FC_{113} = \frac{100}{1 * 1,7} = 58,83$$

$$FC_{111} + FC_{112} + FC_{113} = 100$$

$$FC_{121} = \frac{100}{2 * 0,506} = 98,68$$

$$FC_{121} = \frac{100}{150 * 0,506} = 1,32$$

$$FC_{122} + FC_{124} = 100$$

Con este ejemplo se ve que, efectivamente, los signos con pocas relaciones (SIGNO₃) tienen los mayores factores de certidumbre. Además, se aprecia que el factor de certidumbre es el de la relación Signo-Segmento y no únicamente del Signo, ya que el mismo signo (SIGNO₂) tiene dos distintos factores de certidumbre (FC₁₁₂ y FC₁₂₂).

De aquí se observa que es importante conocer el número de relaciones de cada uno de los signos con los segmentos de nervio. Para no desperdiciar el tiempo de máquina contando el número de repeticiones, durante el proceso de cálculo de los factores de certidumbre se decidió agregar un campo extra a la Base 2 de la BC para guardar el número de relaciones de cada uno de los signos.

Una vez calculados y asignados los pesos, la IM recalcula los pesos de todos los signos que están relacionados con los segmentos de nervios que se vieron afectados por la adición de nuevos signos. Una vez realizado esto, la IM agrega los elementos contenidos en los archivos generados por la II a las bases correspondientes de la BC. Durante este proceso se evalúa si los síntomas o los signos a agregar ya existen en las Bases 3 y 2. En el caso de que sean síntomas o signos nuevos, se les agrega a las bases correspondientes y se les asigna una clave. En caso contrario, lo único que se hace es incrementar en uno el valor del campo que contiene el número de relaciones. También se asigna una clave a la pareja Nervio-Segmento, la que se guarda en la Base 1. Las relaciones entre los Nervios, los Signos y los Síntomas se guardan en las Bases 4 y 5. Más adelante se explicará el procedimiento de asignación de claves.

h.-Módulo de modificación de la información

El módulo de Modificación de la BC es el encargado del manejo de la información contenida en ella. Permite borrar parejas Nervio-Segmento, Signos y Síntomas, así como agregar Signos y Síntomas nuevos. Como ya se vio con anterioridad, la adición de nuevas Signos y Síntomas se realiza a través del Módulo de Adición.

Para borrar información se cuenta con dos submódulos. El primero es el de las parejas Nervio-Segmento, y se encarga de borrar todas las relaciones de una pareja dada, de recalcular todos los factores de certidumbre de los signos relacionados, así como de disminuir el valor del campo que guarda el número de relaciones de los signos y síntomas correspondientes. Mientras que el segundo, el de Signos y Síntomas, permite borrar uno o más signos o síntomas de una pareja Nervio-Segmento dada.

Cuando un signo o un síntoma es dado de baja, además de borrarse su relación de las Bases 5 o 4, también es necesario, para poder recalcular los pesos, disminuir el contador de la Base 2. En el caso de que el contador haya sido igual a 1, el signo también es dado de baja de la Base 5. En el caso de los síntomas, se puede presentar el problema de no saber si el síntoma que se da de baja está aun relacionado con otras parejas Nervio-Segmento, en cuyo caso no se le puede dar de baja de la Base 4. Para resolver este problema, se decidió agregar un campo extra similar al que existe en la Base 5. Por lo tanto, cuando un síntoma es agregado, se sigue el mismo

procedimiento que en el caso de los signos, es decir, si el síntoma es nuevo, se le asigna al campo de contador el valor de 1, y si no, este valor se incrementa en 1. El procedimiento de borrado también es análogo.

Para agregar Signos y Síntomas se cuenta con un submódulo que se encarga de producir nuevas relaciones entre éstos y las parejas Nervio-Segmento correspondientes. El signo o síntoma a agregar puede ser nuevo o ya existente

En el caso de que el usuario lo proponga como nuevo, la IM verifica si en realidad el signo o síntoma lo es. En caso afirmativo, el sistema le asigna una clave y lo guarda junto con el valor de contador de 1 en la base correspondiente. Posteriormente, el sistema pregunta al usuario cuales son las parejas Nervio-Segmento con las que estará relacionado. El resto del proceso es igual al ya descrito, es decir, el sistema genera las relaciones que se guardan en la Base 5 o en la Base 4; incrementa el contador por cada una de las relaciones, y finalmente, recalcula los pesos. En el caso de que el signo o el síntoma ya exista en la BC, el proceso sigue un curso similar, excepto en el momento de asignación de la clave.

Para la asignación de claves, tanto de las parejas Nervio-Segmento como de los Signos y de los Síntomas, se decidió utilizar un método que aprovechara las claves vacantes, evitando así su crecimiento desmedido, la generación de números cada vez más grandes y la eventual interrupción en la secuencia de asignación. El algoritmo ideado se describe a continuación:

Cada vez que una pareja, un signo o un sintoma es dado de baja, su clave es guardada en un archivo especial. Posteriormente, cuando es necesario asignar una clave a un elemento nuevo, se toma la primera de las claves de estos archivos. Si el archivo correspondiente está vacío, la clave asignada es la siguiente a la clave más grande existente.

c.-Módulo de visualización de la información

El último módulo de la IM es el más sencillo, aunque no por ello menos importante, y permite visualizar la BC, haciéndola un despliegue de todo el contenido de ella, o bien, únicamente una o varias parejas. Servio segmento. En ambos casos, el usuario puede observar todas las ramificaciones de cada una de las parejas advirtiendo los síntomas, signos y pesos.

2.6.-Interfaz con el Usuario (IU)

La IU es el módulo encargado de comunicar al usuario con la IM y de ayudarle a realizar algunas funciones complementarias, que se describirán a continuación.

La primera función es permitir al usuario definir el "drive" y el subdirectorio en el que está ubicada la BC, así como el "drive" y subdirectorios en el que van a estar las bases temporales (de trabajo). Con esto se facilita la movilidad de la BC y el ajuste del sistema experto a diferentes configuraciones computacionales.

La siguiente función consiste en permitirle al usuario a definir si se va a trabajar en memoria o en disco. En el caso de que la computadora tuviera suficiente memoria, así como un "no-break" para evitar pérdidas de información al momento de presentarse interrupciones en el suministro de energía, es aconsejable usar la modalidad de memoria, ya que el tiempo de respuesta es más corto debido a que se evitan múltiples lecturas y escrituras a disco.

Asimismo, la IU se encarga de manejar el acceso al sistema para llevar a cabo consultas relacionadas con un paciente nuevo o con uno anteriormente atendido. En el primer caso permite llenar una breve historia clínica que se guarda junto con los resultados de las consultas del paciente; mientras que en el segundo caso, el usuario puede realizar una nueva consulta de un paciente ya atendido, tomando la opción de revisar o modificar la historia clínica de este. Además, el sistema permite examinar las consultas anteriores de un paciente dado, tanto de algunas fechas en particular o de todas en global.

La IU, también le permite al usuario borrar los resultados de las consultas de los pacientes. En esta modalidad, el sistema puede borrar todo el archivo de un paciente dado, o bien, únicamente consultas de algunas fechas en especial.

Por último, la IU se encarga de continuar una consulta que se estaba realizando en disco, y que se hubiese interrumpido por algún error del sistema o por falta de suministro de energía. En este caso, el sistema trata de continuar la sesión con la

información contenida en las bases temporales. La recuperación no siempre es posible, ya que si la falla ocurrió en el momento preciso de la cerradura de las bases, estas se invalidan.

Además de las funciones complementarias anteriores, la IU es el vínculo entre la MI y el usuario. Es a través de esta interfaz que la MI obtiene la información indispensable para el diagnóstico, como son los síntomas de los que se queja el paciente. Es también a través de esta que la MI le solicita al usuario la revisión de determinados signos, y recibe a continuación los resultados de la auscultación. Por último, mediante esta interfaz el sistema le proporciona al usuario el resultado de la consulta. Al final de la misma, el sistema le permite al usuario guardar los resultados obtenidos en su transcurso.

2.7.-Explicador

Este es el módulo encargado de proporcionarle al usuario explicaciones tanto sobre las preguntas que le va haciendo la MI, como sobre los resultados que esta le proporciona. Esta dividido en dos submódulos: el de Cómo y el de Por qué.

a.-Módulo de Cómo

El módulo de Cómo le dice al usuario, al final de la consulta, la forma en la que se llegó a las conclusiones. Le presenta la lista de los síntomas de los que se quejó el paciente, así como la de los signos, invalida esta última en los signos presentados

por el paciente, los no presentados y los desconocidos. También le presenta la lista de las reglas usadas por la MI para que el usuario pueda ver el por qué de los resultados, y en caso de no estar de acuerdo, modificar posteriormente las reglas en la BC. Con base en esto, se espera lograr una mayor confianza del usuario a los resultados proporcionados por el sistema experto, así como tener un mecanismo de retroalimentación para optimar la BC.

h. Módulo de Por qué

El módulo de Por qué le explica al usuario la razón de una u otra pregunta que le hace la MI durante la consulta. Cuando la MI le pregunta al usuario si el paciente presenta o no un signo dado, éste puede inquirir el por qué de la pregunta. El explicador le dará la lista de la o las parejas Nervio-Segmento que están siendo revisadas y que se relacionan con el signo en cuestión. En el caso de que el usuario aún tuviese dudas del por qué de estas parejas, el sistema le proporciona la lista de los síntomas y la de los signos que originaron que estas parejas estuvieran en el espacio de búsqueda. Igual que en el módulo de Cómo, se espera que con ello el usuario le tenga una mayor confianza al proceso de diagnóstico.

3.- Implantación

En esta parte se hablará del proceso de la implantación del sistema. Se verá la forma en la que se eligieron los lenguajes de programación y de las modificaciones que se le tuvo que hacer al sistema para ajustar el diseño teórico a las exigencias de los lenguajes.

3.1.-Elección de los Lenguajes

Finalizado el diseño del proyecto se comienzan a plantear las características de su implantación. Una de las principales acciones fue la evaluación de las necesidades del sistema para la elección del lenguaje de programación.

La siguiente lista muestra las características que se establecieron *requisitos* o deseables dentro del lenguaje para ser usadas en el proyecto:

- *Computación de código*: Permitíannos así la particularidad del lenguaje de generar poco volumen de código objeto y código ejecutable a partir de las instrucciones en código fuente. Del diseño obtenido se estimó que la cantidad de código sería elevada, por lo que se requería un ambiente de programación que permitiera la edición y la codificación de grandes módulos, así como un generador de código objeto eficiente.
- *Programación modular*: Posibilidad de crear estructuras de programación modulares que puedan ser usadas independientemente a su contexto o al resto de todo el proyecto. Esa forma de comunicación entre módulos es particularmente deseable cuando se pretende utilizar rutinas y funciones externas en otros lenguajes. Gracias a la comunicación entre módulos que postula la programación modular, la repetición innecesaria de rutinas en el programa se minimiza, lo cual se visualiza como una necesidad en el proyecto, cuyo diseño plantea módulos iterativos y de rutinas generales.

- **Base de Datos** - Se refiere a la posibilidad de utilizar estructuras de datos de una forma ordenada y eficiente, a través de un manejador de bases de datos. Uno de los puntos básicos del diseño del sistema experto contempla el manejo de información mediante las técnicas de bases de datos, utilizando archivos externos administrados por una máquina.
- **Legibilidad** - Es el conjunto de características que hacen de un ambiente de desarrollo de programas un escenario cómodo y accesible para el usuario. Aunque un ambiente dado puede ser muy conveniente para el desarrollo de una aplicación, puede no serlo para la ejecución del resultado. Deben ser considerados, entonces, ambas facetas del lenguaje para su evaluación. El proyecto requirió de un buen ambiente en ambos aspectos, el de desarrollo y el de ejecución. Atendiendo a este último, se consideró deseable un compilador con independencia de ligador externo, con el fin de poder ligar diversos códigos objeto, desarrollados en cualesquiera lenguajes, en sólo un programa ejecutable.
- **Puqutería** - La existencia de programas de utilidad en un lenguaje reduce la cantidad de código generada, ya que en la mayoría de los casos dichas rutinas se encuentran minimizadas. Algunos lenguajes, sobre todo aquellos de bajo nivel, no poseen comandos o funciones de uso generalizado, sino que es necesario definir rutinas tan básicas como escrituras a elementos de entrada salida, ordenamientos o búsquedas, lo cual provoca retrasos en tiempo de programación. El vasto procesamiento de información que realiza

el programa requiere de numerosas técnicas y rutinas utilizadas en la mayoría de las aplicaciones por computadora, por lo cual es necesario que se encuentren disponibles en el lenguaje elegido.

- **Módulo de gráficos y ventanas.** Son elementos claves en el desarrollo de programas enfocados a usuarios no conocedores de computación, ya que permiten una comunicación directa y más flexible. Los diálogos en diálogo por medio de esos elementos suelen ser más claros y efectivos en sus propósitos, a la vez que permiten un mayor control de respuestas. Dado que el programa se diseña para ser usado por personas no necesariamente involucradas con la computadora, el sistema se concibe para ser lo más transparente posible al usuario. La utilización de ventanas estuvo incorporada al diseño desde el primer momento como una herramienta básica del diálogo.

Puesto que el proyecto se orienta dentro del campo de problemas que requieren de búsquedas extensas, coordinadas por un mecanismo de control, se determinó que un lenguaje declarativo era más conveniente para su implementación que uno procedimental, atendiendo a lo siguiente (Cuadrado, C. Y Cuadrado, J.L., 1985; Eisenbach, S. y Sailer, C., 1985; Kowalski, R., 1985).

- La naturaleza de los lenguajes declarativos permite una abstracción del código en problemas con mucha de grandes volúmenes de datos y posibilidades de solución, pues en ellos no es necesario especificar el flujo de control del problema, sino el flujo de datos.

- Todos los lenguajes declarativos poseen, implícitamente, un algoritmo de búsqueda de todas las soluciones posibles. Si no se desea utilizarlo como método principal, puede ser convenientemente dirigido para realizar búsquedas exhaustivas secundarias que formen parte de la meta principal.
- En los lenguajes procedurales las variables van tomando diversos valores conforme el programa va realizando procesos, teniendo entonces una dependencia posicional. Esto provoca que las verificaciones y los chequeos de los valores de las variables a lo largo del desenvolvimiento del programa se vuelvan difíciles y tediosos. En los lenguajes declarativos, las variables no son significativas por su posición, sino por el papel que representan, siendo entonces elementos conectivos que permiten mayor claridad y transparencia en programas extensos. Las transformaciones, las conversiones y el paso de parámetros son entonces fácilmente realizados y verificados.
- El control del flujo de datos en los lenguajes declarativos es más relajado, ya que el comportamiento de las variables no tiene que ser exhaustivamente explicado al compilador. Esto ayuda a minimizar tiempos en programas extensos, favoreciendo el desempeño de algoritmos largos.

El principal inconveniente de los lenguajes declarativos estriba, básicamente, en su naturaleza lógica. No es fácil acostumbrarse al flujo de control que compone el esqueleto del programa, el cual también es, frecuentemente, una forma de conocimiento del problema que se está resolviendo. En no pocas ocasiones, situaciones laterales no contempladas pueden sorprender al programador, es usual

tener que restringir las posibilidades de solución para evitar resultados imposibles que tratan de encontrar el programa, y en numerosos casos, el flujo de acciones puede salirse de control. La depuración del código llega a ser notablemente complejo y laboriosa, sobre todo atendiendo al aspecto lógico de desenvolvimiento del programa.

Los lenguajes procedurales son, en este aspecto, más controlables, pero también exigen otras sacrificios (Robinson, 1987), como son:

- una mayor cantidad de código,
- mayor especificidad de variables y procedimientos,
- en algunos casos, excesiva complejidad o extensión de programación de determinados problemas.

Por todo lo anterior se decidió hacer la implementación en un lenguaje declarativo. Entre todos los lenguajes declarativos existentes en el mercado para una computadora personal, se eligió Turbo Prolog 2.0, por ser éste el que mejor se ajustaba a las necesidades. Esto es debido a las siguientes características existentes en Turbo Prolog 2.0 (Robinson, 1987), (*Turbo Prolog User's Guide Version 2.0*, 1988), (*Turbo Prolog Toolbox User's Guide and Reference Guide*, 1987)

- Posibilidad de llamar programas objeto y funciones externas escritos en otros lenguajes, está particularmente enfocada a C, Pascal y Ensamblador. También es posible llamar, desde otros lenguajes, rutinas escritas en Turbo Prolog.
- Además de poder trasladar elementos de datos, como 'strings', variables ó constantes, Turbo Prolog permite trasladar estructuras de datos complejas, como registros y listas entre bloques escritos en diferentes lenguajes.
- El entorno de Turbo Prolog comprende dos bases de datos: interna y externa, las cuales poseen cada una, un manejador de datos eficiente. El manejador de la base de datos externa es particularmente apropiado para procesos complejos.
- El código fuente es modulable, contando además con la posibilidad de crear sistemas "project", los cuales se componen de hasta nueve módulos que interactúan en un mismo programa. Esto permite el desarrollo de sistemas de gran escala.
- El lenguaje maneja dos herramientas útiles en la manipulación de mecanismos de Inteligencia Artificial: unificación y "backtracking".
- Diferenciándose del Prolog estándar, Turbo Prolog es naturalmente declarativo en un entorno procedural. Esto es, el lenguaje no es totalmente declarativo, pudiéndose utilizar también algoritmos procedurales en él.

- Posee un depurador de instrucciones o "Debugger", llamado "Trace", que está instalado como un seguidor del programa.
- Posee una gran variedad de rutinas, agrupadas en una biblioteca, la cual contempla manejo de ventanas, manejo de 'strings', procedimientos de búsqueda, ordenamientos, etc.

3.2. Desarrollo del Sistema

Para la implantación del sistema, se decidió crear tres programas independientes, cada uno de los cuales cubre una función especial:

El primer programa realiza la función de la Interfaz con el Experto. Para implantar esta función se decidió utilizar el lenguaje Turbo Pascal 4.0. La H. no requiere de las características de un lenguaje declarativo, por lo que se decidió aprovechar las ventajas de uno procedural.

El segundo programa es el que cubre las funciones de la IM. Este programa, igual que el tercero, que realiza las funciones de la MU de la H. y del Explicador, se desarrolló en Turbo Prolog 2.0. A pesar de que a diferencia del tercero, no requiere muchas de las características de un lenguaje declarativo, la programación se facilitaba, ya que este programa tenía que interactuar con la BC, cuyo diseño se había concebido a partir del manejador de bases de datos de Turbo Prolog 2.0.

Para la Base de Conocimientos se utilizó el manejador de bases de datos externos de Turbo Prolog 2.0. El manejador permite definir una base de datos, ubicarla en disco, en memoria principal o en memoria expandida. Además permite realizar lecturas, escrituras, modificaciones y búsquedas en la base de datos. Para agilizar el proceso, posee una indexación muy eficiente para la cual utiliza los árboles B+. La indexación puede realizarse sobre sólo un campo o un conjunto de ellos. Una misma base de datos puede tener más de un árbol (índice) asignados.

Inicialmente, se pensó utilizar el número de registro como clave de las Bases 1, 2 y 3, aborrrando con ello un campo. Desgraciadamente, el lenguaje no permite realizar ninguna operación, incluyendo la indexación, sobre un campo de tipo "referencia" (*Turbo Prolog Reference Guide Version 2.0, 1988*), que es el del número de registro. La indexación únicamente es permitida sobre un campo o un conjunto de campos de tipo "string". El lenguaje permite las conversiones entre los distintos tipos de campos, pero no entre el tipo "referencia" y cualesquiera otros. Es por ellos que las claves tuvieron que implantarse en forma de un campo tipo "string".

La estructura definitiva de las bases de datos que integran la Base de Conocimiento (Fig.6), es la siguiente:

- Base1: (Nervio, Segmento, ClaveNervio)
- Nervio: tipo "string"
- Segmento: tipo "string"
- ClaveNervio: tipo "string"
- Árbol1: indexación por ClaveNervio
- Árbol2: indexación por Nervio



Base 2 (Signo, Contador de Signo, Maniobra, Clave Signo)

- Signo: tipo "string"
- Contador de Signo: tipo "integer"
- Maniobra: tipo "string"
- Clave Signo: tipo "string"
- Arbol1: indexación por Clave Signo
- Arbol2: indexación por Signo

Base 3 (Síntoma, Contador de Síntoma, Clave Síntoma)

- Síntoma: tipo "string"
- Contador de Síntoma: tipo "integer"
- Clave Síntoma: tipo "string"
- Arbol1: indexación por Clave Síntoma
- Arbol2: indexación por Síntoma

Base 4 (Clave Nervio, Clave Síntoma)

- Clave Nervio: tipo "string"
- Clave Síntoma: tipo "string"
- Arbol1: indexación por Clave Nervio
- Arbol2: indexación por Clave Síntoma

Base 5 (Clave Nervio, Clave Signo, Factor de Certidumbre)

- Clave Nervio: tipo "string"
- Clave Signo: tipo "string"
- Factor de Certidumbre: tipo "real"
- Arbol1: indexación por Clave Nervio
- Arbol2: indexación por Clave Signo

Las Bases Temporales también se implantaron en forma de bases de datos relacionales, igual que las de la Base de Conocimiento. Esto se hizo debido a las siguientes razones:

- Las búsquedas dentro de estas bases son más rápidas que en las bases de datos internas de TurboProlog. Esto debido a la indexación a través de árboles B +.
- La ventaja de que el manejador pudiera ubicar las bases tanto en el disco como en la memoria (principal o expandida) fue considerada importante, ya que el usuario, dependiendo de la configuración de su equipo y de sus necesidades, puede elegir el lugar en el que desea mantenerlas.
- El hecho de tener estructuras similares entre la HC y las Bases Temporales facilitó la transferencia de la información.

La estructura definitiva de las bases de datos que integran las Bases Temporales, incluyendo los árboles de indexación es la siguiente:

Base Temporal1: (ClaveSintoma)

ClaveSintoma: tipo "string".

Árbol1: indexación por ClaveSintoma

Base Temporal2: (ClaveNervio, Status)

ClaveNervio: tipo "string"

Status: tipo "string"

Árbol1: indexación por ClaveNervio

Árbol2: indexación por Status

Base Temporal3: (ClaveNervio, Probabilidad)

ClaveNervio: tipo "string"

Probabilidad: tipo "real"

Base Temporal4 (ClaveSigno)

ClaveSigno: tipo string
 Arbol1: indexación por ClaveSigno

Base Temporal5 (ClaveSigno, Respuesta)

ClaveSigno: tipo string*
 Respuesta: tipo string*
 Arbol1: indexación por ClaveSigno
 Arbol2: indexación por Respuesta

El Status de la Base Temporal2 es la variable que permite saber si la ClaveNervio correspondiente está localizada en el Espacio de Búsqueda o en el de Resultados. Se piensa en esta variable para evitar tener que barrer las ClaveNervio cuando pasamos del Espacio de Búsqueda al de Resultados. Lo único que se hace es cambiar el Status de la función a *desactivado*.

La Base Temporal3 no está indexada debido a que las características del lenguaje, como ya se mencionó con anterioridad, no permiten indexar ningún campo que no sea de tipo string. Se hubiera podido transformar el valor real de Probabilidad a su equivalente string, pero el orden de indexación de los string es distinto al de los números. En una lista ordenada de strings el 10' segunda a 1, antes del 2. Si hubiera 11, 12, etc., y otros strings que coincidan con 1, irían antes de los que comienzan con 2. Es por esto que se dejó el valor de Probabilidad como real, se imprimió una lista de ordenación utilizando para ello el método de la biblioteca *Knuth, DL, 1973*. No se quiso utilizar ningún método más complejo, ya que normalmente el Espacio de Resultados no contiene muchos elementos, por lo que la ganancia de velocidad no sería relevante.

Un problema que se manifestó durante la etapa de verificación del sistema con la información real, es el de insuficiencia de memoria que surge durante consultas complicadas que involucran un elevado número de síntomas y signos. El problema reside en el "stack" del sistema, cuyo manejo es transparente para el programador, y que es usado por Turbo Prolog para los apuntadores. Como Turbo Prolog corre únicamente bajo el ambiente del MS-DOS que no permite un manejo superior a los 640K de memoria, el sistema, en el momento de compilar, fija el "stack" en un tamaño, y aunque el sistema posteriormente se use en una máquina con memoria expandida, el tamaño no varía. Se espera que la versión 3.0 de Turbo Prolog permita ubicar el "stack" en la memoria expandida, o que las nuevas versiones de MS-DOS manejen una cantidad de memoria superior a los 640K.

IV. DISCUSIÓN Y CONCLUSIONES

Uno de los aspectos importantes del trabajo presentado en esta tesis es su naturaleza multidisciplinaria, esto es, requirió de la participación de un grupo de personas con diferente formación académica (e.g., neurólogos e ingenieros), todos unificados bajo la inquietud de **crear un sistema útil e interesante**.

El esfuerzo desarrollado para resolver las dificultades de comunicación entre personas con metas y campos de acción tan dispares, enriqueció la experiencia y guió el trabajo y expectativas hacia el producto que ahora se reporta en esta tesis.

Particularmente, la parte de diseño representó una de las etapas más difíciles del proyecto. La comprensión de la forma de resolución del médico fue esencial para elaborar los algoritmos que se programaron para que SAHIS simulara esa conducta. La falta de información para poder diseñar la Base de Conocimiento, así como la interfaz con el experto, resultó crítica, sobre todo por la diversidad de conocimiento especializado que obstaculizaba la identificación de situaciones y relaciones importantes, haciendo parecer más atractivos otros elementos.

Si se evalúa el sistema de acuerdo a los criterios enunciados en el capítulo II (sección 3.5.3), podemos señalar que SAHIS:

- es un sistema útil que cubre una necesidad real, y además, que ya está siendo utilizado;
- es un sistema fácil de usar, ya que posee interfaces que guían al usuario evitando resolviendo sus dudas y ayudándolo a que se concentre en los aspectos relevantes para el diagnóstico. Los resultados hasta ahora arrojados por el sistema dentro de la actividad normal son bastante halagadores, aunque aún falta someterlo a otras pruebas para evaluar completamente su eficiencia y desempeño;
- utilizando SAHIS, a pesar de que esto puede ser discutible, el usuario puede enriquecer su conocimiento de neurología a través del uso constante del sistema, el explicador facilita esta tarea de aprendizaje ya que guía al usuario a través del proceso de diagnóstico y es capaz de explicar el razonamiento que se sigue para alcanzar el resultado;

- con la inclusión de un módulo Explicador y de las maniobras explicativas, en SADIN se logra resolver y explicar la mayoría de las dudas que el usuario pueda tener;
- la Interfaz de Mantenimiento permite visualizar y modificar la información contenida en la Base de Conocimiento; y
- finalmente, cabe señalar que debido a su complejidad, la condición de que el sistema debe de ser capaz de aprender no fue cubierta en esta etapa.

A pesar de que los resultados preliminares obtenidos con el sistema dentro de la actividad normal son bastante buenos, hace falta aún someterlo a otras pruebas para evaluar completamente su eficiencia y desempeño. Estas pruebas no se realizaron por falta de interacción con médicos no especialistas. Una de esas pruebas tendría que comparar los resultados arrojados por un médico no especialista auxiliando su diagnóstico con el uso de SADIN, contra los arrojados por un médico experto (i.e., neurólogo). Otra evaluación consistiría en alimentar el Sistema Experto con datos de casos cuyo diagnóstico haya sido comprobado y comparar los resultados que genere para medir su exactitud. Los registros estadísticos de pacientes y de desempeño médico, si los hubiera, serían particularmente útiles en la estimación de la posición del Sistema Experto dentro de un marco de desenvolvimiento médico.

Si evaluamos el sistema desde el punto de vista de ingeniería de software (Shooman, M.L., 1985), se debe señalar que la programación se hizo modular y estructurada. Las Bases de Conocimiento y las Temporales fueron implantadas a través de bases de datos relacionales,

que pasaron por el proceso de normalización. Las búsquedas a través de estas bases se hacen utilizando para ello los árboles B+. Todo ello hizo como objetivo una optimización tanto del tiempo de procesamiento, como del tamaño de los archivos.

Una de las particularidades de SADIN es la forma en la que se eligen las reglas para formar el Espacio de Búsqueda. A diferencia de otros sistemas que revisan todas las reglas que tienen en su Base de Conocimiento, SADIN utiliza los Síntomas, los que le permiten determinar el primer subconjunto de Segmentos de Nervio que son factibles de estar lesionados. Posteriormente, el uso de los Signos permite corroborar la existencia de la lesión en cada uno de los segmentos, según a la Máquina de Inferencias hacia ciertos probables. Con esto se evita que el sistema tenga que revisar todas y cada una de las reglas que están contenidas en su Base de Conocimiento. Se puede decir que el número de segmentos que se verifican es el mínimo, ya que cada uno de ellos, salvo los primeros que fueron utilizados en base a los síntomas, tiene una probabilidad mayor que cero. Aun los primeros suelen tener una probabilidad superior a cero ya que no fueron obtenidos aleatoriamente, sino con base en los síntomas, los cuales, al llegar a ser determinísticos, indican los segmentos más probables.

Otro aspecto novedoso es el algoritmo que utiliza la *Intertax de Mantenimiento* para la asignación de las probabilidades a cada uno de los signos asociados con los segmentos de nervio. En contraposición a otros sistemas en los que el experto es el encargado de proporcionar la probabilidad asociada a cada una de las reglas, SADIN los calcula en forma automática. Esto debido al hecho de que la mayoría de los neurólogos no pueden concretar el valor exacto que aporta cada uno de los signos a cada uno de los segmentos de nervio. Para simplificar el cálculo, el sistema supone que la probabilidad de que el paciente

manifieste cualquier signo de los que están asociados con el segmento lesionado es la misma. Basándose en este supuesto, el sistema calcula la aportación de cada uno de los signos dependiendo de la cantidad de segmentos con los que están relacionados.

A través de simulaciones se ha comprobado que las probabilidades que asigna el sistema se acercan mucho a los diagnósticos hecho por expertos humanos. El algoritmo puede ser afinado en un futuro si se le agrega la probabilidad de la manifestación de cada uno de los signos, misma que puede ser calculada a partir de estudios estadísticos sobre una gran cantidad de pacientes reales.

Una particularidad de SADDN que no se ve en la mayoría de los sistemas es la diferenciación funcional de las figuras del experto, del mantenedor y del usuario. Aunque en la práctica una misma persona puede desempeñar las tres funciones, el hecho de diferenciarlas permite formalizar las actividades y englobarlas en interfaces distintas. SADDN también prevé la posibilidad de recibir aportaciones de distintos expertos, las que en un momento dado pueden llegar a ser contradictorias o inconsistentes. Es por ello que todos los archivos producidos por la interfaz con el experto pasan a través de la de Mantenimiento antes de ser incorporados definitivamente a la Base de Conocimiento. La figura del mantenedor es la única autorizada para modificar la Base de Conocimiento.

Por último, se considera muy importante el señalar un problema que se ha observado en la utilización del sistema, el cual es independiente de SADDN y puede ocasionar errores en el diagnóstico. El problema se presenta cuando la información que se alimenta al sistema es incorrecta. Si el usuario (e.g., médico no especialista) cree que el paciente presenta un signo que en realidad no presenta, el sistema dará un diagnóstico completamente acorde a los

datos que le fueron proporcionados pero desajuste a la realidad. Es decir, la exactitud del diagnóstico del sistema depende de la experiencia que tenga el usuario en interpretar los signos que presenta el paciente. La incorporación, en un futuro, de un módulo que reciba en tiempo real los resultados de algunas pruebas médicas, como por ejemplo, de un electroencefalograma, podría disminuir la gravedad de este problema y hacer el proceso de diagnóstico mucho más sencillo.

No se debe de olvidar al paciente y su familia durante la etapa de verificación del sistema con la información real, que se mencionó en el capítulo anterior, que es el de insuficiencia de la memoria que surge durante consultas complicadas que involucren un elevado número de síntomas y signos. Se espera que la versión 3.0 de *Turno Prolog* o las nuevas versiones de *MS-DOS* permitan resolverlo.

Se considera útil volver a mencionar el hecho de que este trabajo es la primera etapa de un proyecto global más amplio. Esto es, el diagnóstico neurológico está formado por varias fases, de las cuales la primera es la de la identificación de la localización de la región en la que está ubicada la lesión. Esta es la etapa que se ha cubierto en este proyecto.

Una futura etapa sería la de identificar la enfermedad que causó o está causando dicha lesión. Para ello se tiene que incrementar la Base de Conocimiento para que además de contener las reglas referentes a las probables ubicaciones de las lesiones, abarque las reglas necesarias para poder hacer un diagnóstico completo. Es más probable que en esta etapa además de los datos que se han utilizado en la anterior (Nervios, Signos, Síntomas, etc.) se tenga que usar algunos otros provenientes de la historia clínica del paciente (e.g. la edad, el sexo, las enfermedades padecidas con anterioridad, etc.) entre otros.

Otra etapa futura podría ser aquella que con base en las dos anteriores, es decir, en la localización de la ubicación de la lesión y en la identificación del mal que la ha provocado, diseñe un tratamiento terapéutico completo. Esto es, el sistema podría diseñar un tratamiento adecuado que incluya los distintos medicamentos y su dosis apropiada. Para esta etapa se tendría que incluir en la Base de Conocimiento la lista de las medicinas, las enfermedades que éstas curan y la forma de su dosificación. Otros datos importantes podrían ser los relacionados con la medicación como por ejemplo, el peso del paciente, algunas otras medicinas que esté tomando en ese momento, las alergias y los rechazos a algunas medicinas en especial, por mencionar sólo algunos.

Además de estas posibles etapas futuras, es posible prever una serie de mejoras al mismo programa. Entre estas posibles mejoras están:

- **Aprendizaje:** Sería un módulo que registrara los resultados reales de un paciente, y con base en ellos, fuera modificando los factores de certidumbre de su Base de Conocimiento. Es decir, entre más pacientes tratara el sistema, mayor exactitud tendría en sus pronósticos. Además, actualmente el sistema supone que la probabilidad de la aparición de cualquiera de los signos del subconjunto relacionado con un segmento de un nervio dado es la misma. El diagnóstico podría afinarse introduciendo en el sistema una base de datos conteniendo la estadísticas de la aparición real de los signos para cada uno de los segmentos. Inclusive, estas estadísticas podrían variar dependiendo de la edad del paciente, de su sexo, de la zona geográfica de la que proviene, etc.

- **Gráficas:** Sería un módulo que tuviera digitalizado el cuerpo humano y pudiera mostrar la localización de la lesión en forma gráfica.
- **Captura Automática:** Sería un módulo que recibiera en tiempo real los resultados de algunas pruebas médicas, como por ejemplo, de un electroencefalograma, y los aprovechara para afinar el diagnóstico.

REFERENCIAS

- Atkinson, I. V.**, *Pascal Programming*, John Wiley & Sons, Great Britain, 1983.
- Craik, K.J.W.**, *The Nature of Explanation*, Cambridge University Press, London, 1943.
- Cuadrado, C.Y. y Cuadrado, J.L.**, "Prolog Goes to Work", *Byte*, vol.10, no 8, Agosto, 1985, pp. 151-160.
- DeMyer, W.**, *Technique of the Neurological Examination*, McGraw Hill, USA, 1980.
- Duda, R.O., Garsching, J.G. y Hart, P.F.**, "Model Design in PROSPECTOR Consultant System for Mineral Exploration", *Expert Systems in the Microelectronic Age*, Edinburgh University Press, 1979, pp. 153-167.

REFERENCIAS

- Duda, R.O. y Garsching, J.G., "Knowledge-Based Expert Systems Coming of Age", *Byte*, vol.6, no.9, Septiembre, 1981, pp. 238-281.
- Eisenbach, S y Sadler, C., "Declarative Languages: An Overview", *Byte*, vol.10, no.8, Agosto, 1985, pp. 181-200.
- Everest, G.C., *Database Management*, McGraw-Hill, Singapore, 1980.
- Geduld, H.M. y Gottesman, R. (Eds.), *Robots, robots, robots...*, New York Graphic Society, Boston, 1978.
- Gevarter, W.M., *Maquinas Inteligentes*, Diaz de Santos, Madrid, 1987.
- Knuth, D.F., *The Art of Computer Programming*, Addison-Wesley, USA, 1973.
- Kowalski, R., "Logic Programming", *Byte*, vol.10, no.8, Agosto, 1985, pp. 161-180.
- Lara Rosano, F. y Gelman Kipnis G., *Metodos y Modelos de la Ingenieria del Conocimiento para Sistemas Expertos*, Instituto de Ingenieria, UNAM, 1980.
- Martin-Abreu, I., *Fundamentos del Diagnóstico*, Francisco Méndez Cervantes, México, 1978.
- McCulloch, W.S. y Pitts, W.S., "A Logical Calculus of the Ideas Immanent in Neural Nets", *Bulletin of Mathematical Biophysics*, vol.5, 1943.
- Miller, R.A., Pople, H.F. y Myers, J.D., "Internist-I: An Experimental Computer-Based Diagnostic Consultant for General Internal Medicine", *The New England Journal of Medicine*, vol.307, no.8, 1982, pp. 428-436.
- Minsky, M., *A Framework for Representing Knowledge*, The Psychology of Computer Vision, McGraw-Hill, New York, 1975.

REFERENCIAS

- Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, McGraw Hill, Japan, 1984.
- Real Academia Española, *Diccionario de la Lengua Española*, Espasa-Calpe, Madrid, 1970.
- Rich, F., *Artificial Intelligence*, McGraw Hill, Singapore, 1985.
- Robinson, P.R. *Aplicación Turbo Prolog*, McGraw Hill, Madrid, 1987.
- Rosenbluth, A., Wiener, N. y Bigelow, J., "Behavior, Purpose, and Teleology", *Philosophy of Science*, vol.10, 1943, pp 18-24
- Schwartz, M., *Transmisión de Información, Modulación y Ruido*, McGraw Hill, México, 1983
- Shoeman, M.L., *Software Engineering*, McGraw Hill, Singapore, 1985.
- Shortliffe, F.H., *Computer Based Medical Consultations: MYCIN*, American Elsevier, New York, 1976.
- The Handbook of Artificial Intelligence*, W. Kaufmann, California, 1982.
- Turbo Prolog Reference Guide Version 2.0*, Borland, USA, 1988
- Turbo Prolog-User's Guide Version 2.0*, Borland, USA, 1988
- Turbo Prolog Toolbox-User's Guide and Reference Guide*, Borland, USA, 1987.

BIBLIOGRAFÍA

Arden, B.W. (Ed). *What Can Be Automated*. MIT Press, Cambridge, 1980.

Brady, M. y Gerhardt, L. (Eds). *Artificial Intelligence*, Springer-Verlag, New York, 1984.

Brown, J.S. y Hurton, R.R., "Multiple Representation of Knowledge for Tutorial Reasoning", *Representation and Understanding: Studies in Cognitive Science*, Bobrow, D.G., y Collins, A. (Eds.), Academic Press, New York, 1975, pp. 311-349.

Buchanan, B.G. y Mitchell, T.M., *Model Directed Learning of Production Rules*, Rept. STAN-CS 77-597, Computer Science Department, Stanford University, California, 1977.

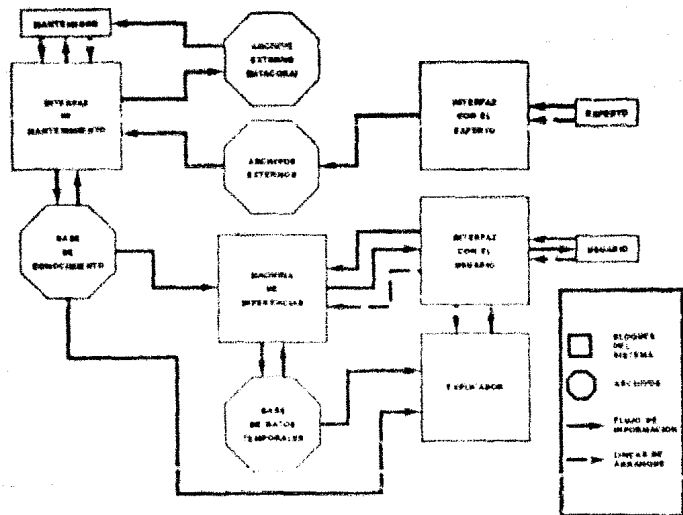
Buchanan, B.G. y Feigenbaum, E.A., "DENDRAL and METADENDRAL: Their Applications Dimension", *Artificial Intelligence*, vol.11, 1978, pp. 5-24.

- Davis, R. v Lenat, D.B.**, *Knowledge-Based Systems in Artificial Intelligence*, McGraw Hill, USA, 1982.
- Duda, R.O.**, 'State of Technology in Artificial Intelligence', *Research Directions in Software Technology*, MIT Press, Cambridge, 1979.
- Erman, J.D., Hayes-Roth, F., Lesser, V.R. y Reddy, D.R.**, 'The HEARSAY-II Speech Understanding System: Integrating Knowledge to Resolve Uncertainty', *Computing Science*, vol.12, no.2, 1980, pp. 273-283.
- Feigenbaum, E.A., Buchanan, B.G., y Lederberg, J.** 'On Generality and Problem Solving: A Case Study Using the DD-NDRM Program', *Machine Intelligence*, vol.6, 1971.
- Hayes-Roth, F.** (Ed.), *Building Expert Systems*, Addison-Wesley, 1983.
- Lindsay, R.K., Buchanan, B.G., Feigenbaum, E.A. y Lederberg, J.**, *Applications of Artificial Intelligence to Organic Chemistry: The DD-NDRM Program*, McGraw Hill, New York, 1980.
- Martin, W.A. y Fateman, R.J.**, 'The MACSYMA System', *Second Symposium on Symbolic and Algebraic Manipulation*, Los Angeles, 1971, pp. 59-75.
- McDermott, J.**, R1: An Expert in the Computer Systems Domain, *AAAI*, vol.1, 1980, pp. 269-274.
- McDermott, J.**, R1: A Rule-Based Configurer of Computer Systems, *Artificial Intelligence*, vol. 19, 1982, pp. 39-88.
- Newell, A. y Simon, H.A.**, *GPS: A Program That Simulates Human Thought*, *Computers and Thought*, McGraw Hill, New York, 1963.

BIBLIOGRAFÍA

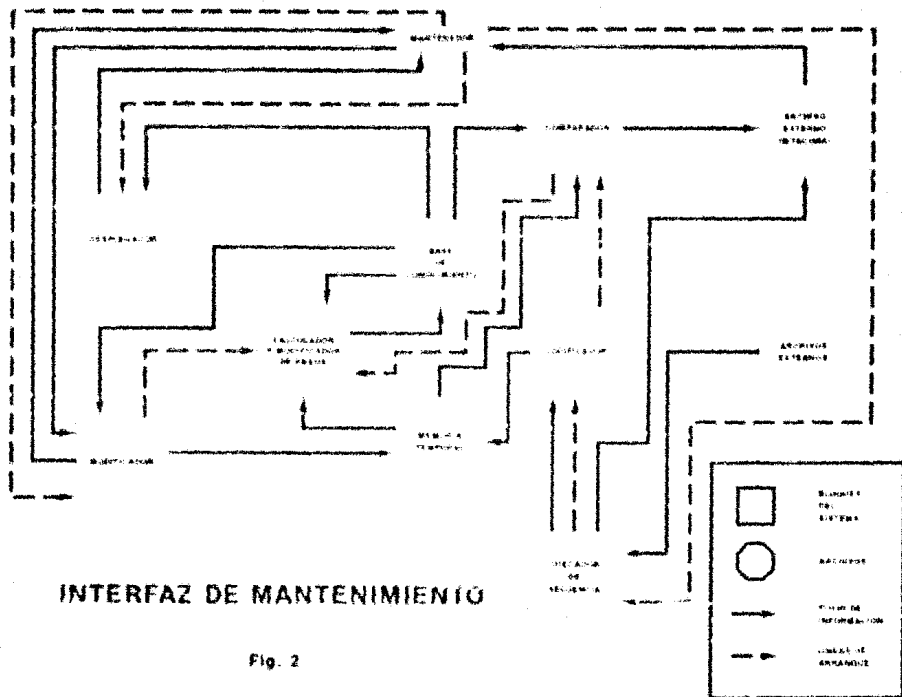
- Newell, A., "Heuristic Programming - Ill-structures Problems", *Progress in Operations Research*, vol.3, John Wiley & Sons, New York, 1969, pp. 360-414
- Nilsson, N.J., *Principles of Artificial Intelligence*, Tioga, Palo Alto, 1980
- Nilsson, N.J., "Artificial Intelligence - Engineering, Science or Sorcery", *AI Magazine*, vol.3, no.1, Winter, 1981/1982, pp. 2-9
- Pople, H.E., Myers, J.D. y Miller, K.A., "DIALOG: A Model of Diagnostic Logic for Internal Medicine", *IJCAI*, vol.4, 1975, pp. 848-855
- Pople, H.E., "The Formation of Composite Hypotheses in Diagnostic Problem Solving: An Exercise in Synthetic Reasoning", *IJCAI* vol.5, 1977, pp. 1030-1037
- Pople, H.E., "Heuristic Methods for Imposing Structure on Ill-structured Problems: The Structuring of Medical Diagnostics", *Artificial Intelligence in Medicine*, Westview Press, Colorado, 1981, pp. 119-185.
- Schank, R.C., "The Current State of AI: One Man's Opinion", *AI Magazine*, vol.4, no.1, Winter-Spring, 1983, pp. 3-8.
- Weiss, S.M., Kulikowski, C.A. y Saffir, A., "A Model- Based Consultation System for the Long-term Management of Glaucoma", *IJCAI*, vol.5, 1977, pp. 826-832.
- Weiss, S.M., Kulikowski, C.A. y Saffir, A., "A Model- Based Method for Computer Aided Medical Decision-making", *Artificial Intelligence*, vol.11, 1979, pp. 145-172.
- 1^{er}. *Symposium Internacional de Inteligencia Artificial*, McGraw Hill, México, 1988
- 2^{do}. *Symposium Internacional de Inteligencia Artificial*, McGraw Hill, México, 1989

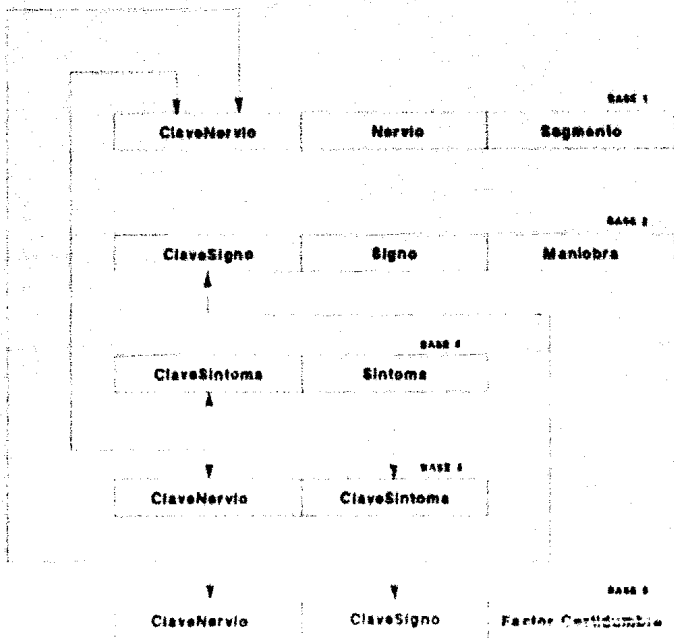
FIGURAS



SISTEMA GLOBAL

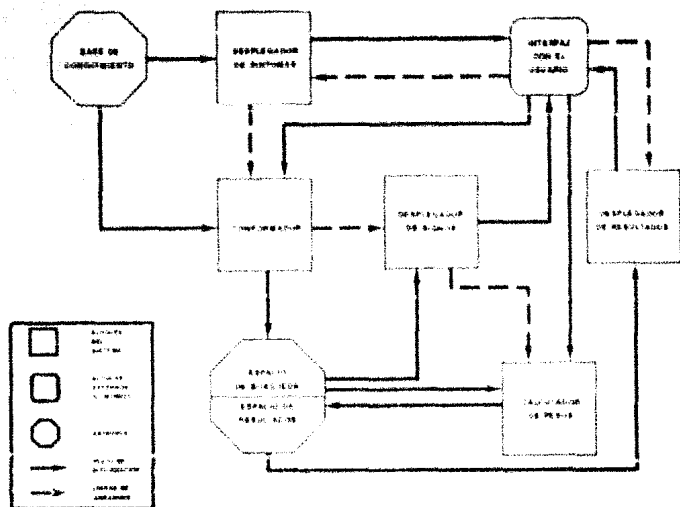
Fig. 1





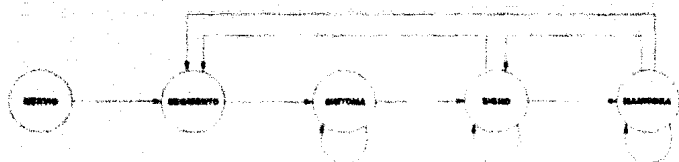
BASE DE CONOCIMIENTO (Estructura Teórica)

Fig. 3



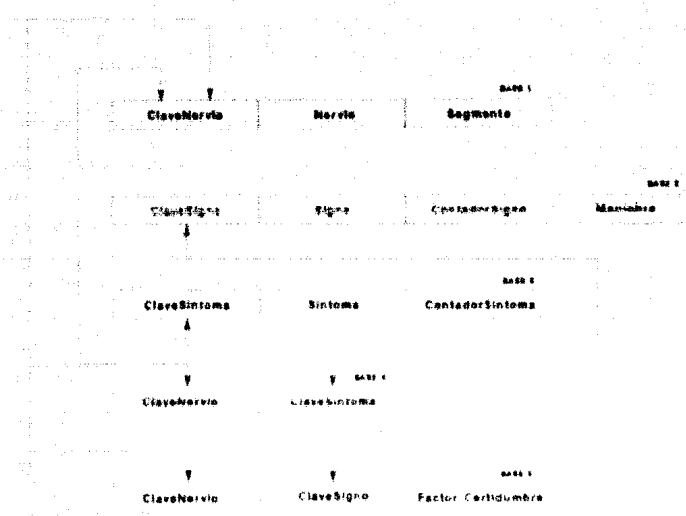
MAQUINA DE INFERENCIAS

Fig. 4



SECUENCIA PERMITIDA

Fig. 5



BASE DE CONOCIMIENTO (Estructura Real)

Fig. 6

SUPLEMENTO I

INTERFAZ CON EL USUARIO Y MÁQUINA DE INFERENCIAS

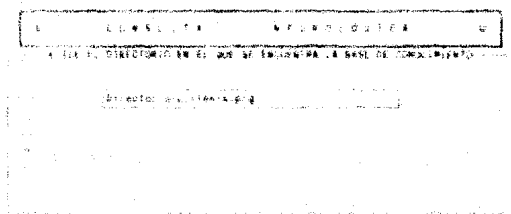
Con el objeto de mostrar una sesión típica del sistema SADIN, se presenta a continuación una sucesión de pantallas que ilustran el desarrollo de una consulta neurológica tal como se le ofrece al usuario.

El número de pantallas mostrado no comprende la totalidad de las que es capaz de generar el sistema, así como tampoco es único el orden en que se exponen, sino que en este espacio se exhibe una de tantas posibilidades de comportamiento que ostenta SADIN.

Por supuesto, se trató de cubrir una gran parte de las habilidades del sistema en esta demostración, pero hubiera sido necesario realizar numerosas y diferentes consultas para poder contemplar el conjunto completo de las bondades que muestra el sistema.

Se eligió en esta ocasión un conjunto pequeño de síntomas y signos, con el propósito de no extender demasiado la serie de pantallas que se producen.

El primer resultado que se muestra al usuario es aquel en el que se le pide proporcionar el directorio en el que se encuentra la Base de Conocimiento:



Este direccionamiento es tanto indispensable como conveniente, pues permite al usuario tener la Base de Conocimiento en diversos dispositivos de almacenamiento, e incluso manipular la información según sus necesidades y propósitos.

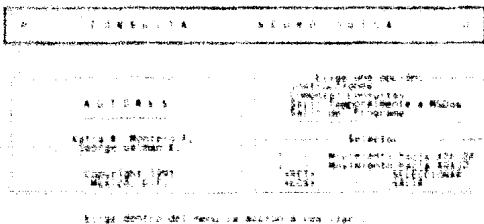
Seguida a esta pantalla, y después de aprobar el directorio introducido por el usuario, el sistema pregunta por el directorio en el que se va a trabajar, en una pantalla similar a la anteriormente mostrada.

Enseguida, previa verificación del directorio anterior, SADIN muestra una pantalla de elección en la que se pide al usuario que elija dónde va a hacer residir la información, si en memoria principal o secundaria.

D	CONSULTA	EUFOROLÓGICA	U
¿DÓNDE DEBE RESIDIR LA INFORMACIÓN?			
<p>1. En memoria principal (se obtendrán respuestas más rápidas, pero el sistema quedará expuesto a no poder recuperarse de una falla de suministro de energía antes del término de la sesión)</p> <p>2. En memoria secundaria (se mantendrá la integridad de los archivos que en ese momento conforman el estado del problema. El costo de esta acción es el sacrificio que en tiempo de acceso debe realizar SADIN para desempeñarse)</p>			
<input type="checkbox"/> Opción 1 <input type="checkbox"/> Opción 2			

Esta disyuntiva se plantea al usuario como una selección frente a sus prioridades: rapidez de consulta o seguridad de ésta. Bajo el primer aspecto, en el cual se le da la mayor importancia al tiempo de desempeño del sistema, se deberá optar por hacer residente la información en memoria principal, en la cual se obtendrán respuestas más rápidas, pero el sistema quedará expuesto a no poder recuperarse de una falla de suministro de energía antes del término de la sesión. Por el contrario, bajo la segunda opción, el hecho de optar por hacer residir la información en memoria secundaria permitirá que ésta se recobre de una interrupción por falta de suministro de energía y se mantenga la integridad de los archivos que en ese momento conforman el estado del problema. El costo de esta acción es el sacrificio que en tiempo de acceso debe realizar SADIN para desempeñarse.

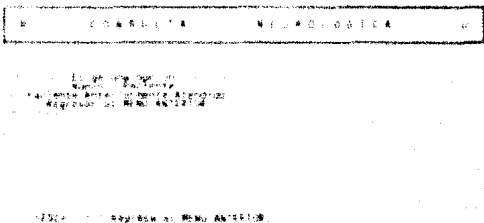
Una vez elegido el lugar donde residirá la información, se muestra al usuario la pantalla de bienvenida, en la cual se le invita a entrar al sistema.



Es posible, en este punto, seleccionar la opción de "INSTRUCCIONES", con lo cual se le presentará al usuario un ambiente de visualización de textos en los que se muestra una breve introducción al sistema y los términos básicos de su utilización.

De igual forma es factible iniciar la interacción con MS-DOS y su conjunto de comandos bajo la tercera opción.

Al elegir la segunda de las opciones (i.e. "COMENZAR CONSULTAS"), se despliega ante el usuario el primer (de una serie de) menús. Este en particular le permitirá identificar un paciente, ya sea nuevo o conocido, para comenzar su consulta.



En el caso de que el paciente no fuese nuevo, el programa presentará un menú en el que el usuario podrá señalar el tipo de acción que desea realizar con las consultas (*generar una nueva, continuar una interrumpida, revisar consultas anteriores, o bien, borrarlas*). Mención especial requiere la opción de continuar una consulta interrumpida, pues es mediante esta opción que el sistema puede reparar las estructuras que se hubiesen dañado por causa de alguna interrupción del sistema, particularmente por fallas de suministro de energía.

Inmediatamente después de seleccionar "NUEVO PACIENTE", SADIN ofrece para su captura un escenario con la Historia Clínica del nuevo paciente, permitiéndole aportar los datos que el sistema requiere para identificar un nuevo individuo:

```

Fecha : 18 ago 1991                               hora: 12:52:40
                                     HISTORIA CLINICA
Nombre del paciente:                               edad:      sexo:
Fecha de nacimiento:
Enfermedades padecidas:
Observaciones 1:
Observaciones 2:
Observaciones 3:
Observaciones 4:
Observaciones 5:
Comentarios :
                                     *fin* para terminar
  
```

Este módulo de captura presenta numerosas utilerías que facilitan la tarea del usuario, como es la inclusión de ventanas de selección de *opciones* que aparecen al situar el *cursor* sobre el campo. Por ejemplo, la que permite elegir el sexo del paciente:

D	CONSULTA NEUROLÓGICA	U
---	----------------------	---

Responde SI, NO o DESCONOCIDO acerca del signo que se muestra

HIPOTENSIÓN EN CUARTO Y QUINTO DEDO DE LA MANO

DESCONOCIDO

SI

NO

<F> por que... <RÉTERNA> para marcar y desmarcar... <S>, <N> o <P> para moverse

En este momento, SADIN comienza a tratar de probar las hipótesis que ya ha formado con la información que se le suministró (i.e., síntomas y signos), y continúa cuestionando con la siguiente serie de signos, los cuales están relacionados con posibles lesiones:

D	CONSULTA NEUROLÓGICA	U
---	----------------------	---

Responde SI, NO o DESCONOCIDO acerca del signo que se muestra

MANO EN GARZA

DESCONOCIDO

SI

NO

<F> por que... <RÉTERNA> para marcar y desmarcar... <S>, <N> o <P> para moverse

D	CONSULTA NEUROLÓGICA	U
---	----------------------	---

Responde SI, NO o DESCONOCIDO acerca del signo que se muestra

ABRIR DE LA ALCELA ROTULIANA

DESCONOCIDO

SI

NO

<F> por que... <RÉTERNA> para marcar y desmarcar... <S>, <N> o <P> para moverse

P	C	CONSULTA	NEUROLOGICA	Q
---	---	----------	-------------	---

INVESTIGAR EN CUANTO Y CUANTO DEBO DE LA MANO
EXAMINE LA SUSTITUCION EN CUANTO Y CUANTO DEBO

F10 - Terminar la inspeccion. INSPECCIONA CON LAS FLECHAS DE MOVIMIENTO

En este punto, cabe mencionar que todas las ventanas de despliegue de textos, ya sea de presentación o explicación, son factibles de ser inspeccionadas en forma interactiva, esto último con el fin de poder incluir en tales ventanas textos de información de extensión superior a la superficie alojada en pantalla.

Nuevamente SADIN invita a la recapitulación y pregunta si el usuario ya ha podido probar la presencia o ausencia en algunos de los signos que había marcado con la etiqueta de "DESCONOCIDO" y así cambiarlos.

2	C	CONSULTA	NEUROLOGICA	Q
---	---	----------	-------------	---

Deseg. MODIFICAR algunos de los signos cuya existencia se
desconoce?
Responda SI O NO

Contestando afirmativamente, se verá desplegar la lista de los signos:

P	CONSULTA	NEUROLOGICA	U
---	----------	-------------	---

ESCRIBIR LOS SIGNOS QUE RESPONDA DEREGE MODIFICANDO
 LOS SIGNOS EN CUANTO Y CUANTO DEBE DE LA MANO

OPCIÓN TERMINAR: RETORNO MENÚ Y DESMARCAR * * * * * PARA OBTENERLO

Eligiendo uno o más signos, se le volverán a mostrar para su regrabación, pero ahora con la posibilidad de selección restringida a "SI" y "NO":

P	CONSULTA	NEUROLOGICA	U
---	----------	-------------	---

RESPONDER SI O NO ALGUNO DEL SIGUENTE GRUPO DE SIGNOS

RESPONDER SI O NO ALGUNO DEL SIGUENTE GRUPO DE SIGNOS

SELECCIÓN PARA MARCAR Y DEMARCAR * * * * * PARA INVERSER

Llegado a este estado, SADES pregunta al usuario si el paciente presenta otros síntomas que aun no hayan sido declarados

P	CONSULTA	NEUROLOGICA	U
---	----------	-------------	---

Presenta o responde SIGUENTES SIGNOS * * * * *

Al contestar negativamente, esta vez le preguntará por otros signos que no hubiesen sido advertidos con anterioridad.

```

P CONSULTA NEUROLOGICA Q
  
```

```

Presenta el paciente otros SIGNOS ?
Responde SI O NO ?
  
```

Si en este punto se responde "SI", entonces el sistema desplegará la lista de los signos para que el usuario pueda elegir entre ellos:

```

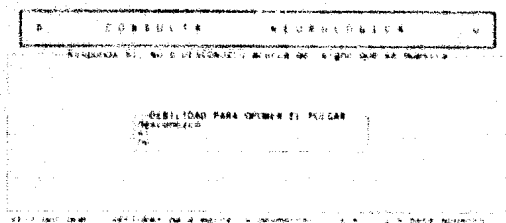
P CONSULTA NEUROLOGICA Q
  
```

```

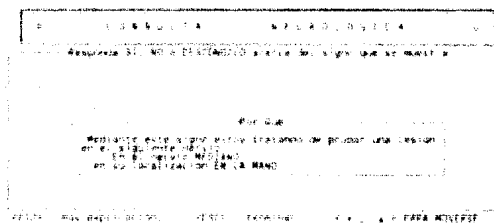
PREGUNTA: ¿Presenta el paciente:
1. DOLOR EN EL CODO
2. DOLOR EN EL CODO
3. DOLOR EN EL CODO
4. DOLOR EN EL CODO
5. DOLOR EN EL CODO
6. DOLOR EN EL CODO
7. DOLOR EN EL CODO
8. DOLOR EN EL CODO
9. DOLOR EN EL CODO
10. DOLOR EN EL CODO
11. DOLOR EN EL CODO
12. DOLOR EN EL CODO
13. DOLOR EN EL CODO
14. DOLOR EN EL CODO
15. DOLOR EN EL CODO
16. DOLOR EN EL CODO
17. DOLOR EN EL CODO
18. DOLOR EN EL CODO
19. DOLOR EN EL CODO
20. DOLOR EN EL CODO
21. DOLOR EN EL CODO
22. DOLOR EN EL CODO
23. DOLOR EN EL CODO
24. DOLOR EN EL CODO
25. DOLOR EN EL CODO
26. DOLOR EN EL CODO
27. DOLOR EN EL CODO
28. DOLOR EN EL CODO
29. DOLOR EN EL CODO
30. DOLOR EN EL CODO
31. DOLOR EN EL CODO
32. DOLOR EN EL CODO
33. DOLOR EN EL CODO
34. DOLOR EN EL CODO
35. DOLOR EN EL CODO
36. DOLOR EN EL CODO
37. DOLOR EN EL CODO
38. DOLOR EN EL CODO
39. DOLOR EN EL CODO
40. DOLOR EN EL CODO
41. DOLOR EN EL CODO
42. DOLOR EN EL CODO
43. DOLOR EN EL CODO
44. DOLOR EN EL CODO
45. DOLOR EN EL CODO
46. DOLOR EN EL CODO
47. DOLOR EN EL CODO
48. DOLOR EN EL CODO
49. DOLOR EN EL CODO
50. DOLOR EN EL CODO
51. DOLOR EN EL CODO
52. DOLOR EN EL CODO
53. DOLOR EN EL CODO
54. DOLOR EN EL CODO
55. DOLOR EN EL CODO
56. DOLOR EN EL CODO
57. DOLOR EN EL CODO
58. DOLOR EN EL CODO
59. DOLOR EN EL CODO
60. DOLOR EN EL CODO
61. DOLOR EN EL CODO
62. DOLOR EN EL CODO
63. DOLOR EN EL CODO
64. DOLOR EN EL CODO
65. DOLOR EN EL CODO
66. DOLOR EN EL CODO
67. DOLOR EN EL CODO
68. DOLOR EN EL CODO
69. DOLOR EN EL CODO
70. DOLOR EN EL CODO
71. DOLOR EN EL CODO
72. DOLOR EN EL CODO
73. DOLOR EN EL CODO
74. DOLOR EN EL CODO
75. DOLOR EN EL CODO
76. DOLOR EN EL CODO
77. DOLOR EN EL CODO
78. DOLOR EN EL CODO
79. DOLOR EN EL CODO
80. DOLOR EN EL CODO
81. DOLOR EN EL CODO
82. DOLOR EN EL CODO
83. DOLOR EN EL CODO
84. DOLOR EN EL CODO
85. DOLOR EN EL CODO
86. DOLOR EN EL CODO
87. DOLOR EN EL CODO
88. DOLOR EN EL CODO
89. DOLOR EN EL CODO
90. DOLOR EN EL CODO
91. DOLOR EN EL CODO
92. DOLOR EN EL CODO
93. DOLOR EN EL CODO
94. DOLOR EN EL CODO
95. DOLOR EN EL CODO
96. DOLOR EN EL CODO
97. DOLOR EN EL CODO
98. DOLOR EN EL CODO
99. DOLOR EN EL CODO
100. DOLOR EN EL CODO
  
```

Use las teclas F10 para terminar, F11 para salir y desmenujar, y las teclas de movimiento.

Marcando el signo "SENSIBILIDAD NORMAL DE LA MANO", se provocará que SADIN comience otro interrogatorio sobre la presencia o ausencia de signos tratando de probar nuevas hipótesis.



Una de las habilidades de SADDN es la capacidad que tiene de explicación de sus acciones a lo largo del programa. Si se oprime la tecla de *POR QUÉ* en este momento (e. i. la tecla de función < F10 >), la respuesta del explicador será la que a continuación se ofrece:



Como se puede apreciar, aún es posible pedir otra explicación al sistema, invocando con ello un segundo nivel de profundidad en el *POR QUÉ*:

CONSULTA	NEUROLÓGICA
Responda SI, NO o DESCONOCIDO acerca del signo que se muestra	
<p>Por que</p> <p>La respuesta al siguiente signo fue SI o DESCONOCIDO: DEBILIDAD MANUAL DE LA MANO</p>	

Finalmente, al oprimir la tecla de función < F10 >, el sistema vuelve al interrogatorio, y después de obtener una respuesta sobre el signo anteriormente mostrado, despliega la ventana de preguntas con un nuevo signo

CONSULTA	NEUROLÓGICA
Responda SI, NO o DESCONOCIDO acerca del signo que se muestra	
<p>DEBILIDAD PARA AGRADE EL PULGAR</p> <p>DESCONOCIDO</p> <p>SI</p> <p>NO</p>	

El cual es marcado con "NO" en este ejemplo particular. Con esta acción, el sistema satisface todas las hipótesis que generó en el ejemplo, y procede a emitir sus resultados:

Guardando la consulta:

```
Directorio:CIYTES10.PRG
```

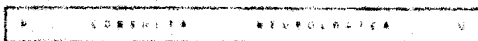
y a preguntar al usuario por el nombre del archivo:

Guardando la consulta:

```
Nombre del archivo: JUANPEZ
```

A partir de este momento, **SADIN** devuelve el control al sistema de menús, desde los cuales es posible llegar a aquel donde se elige el tipo de consulta a realizar.

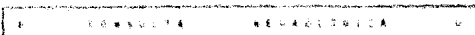
Con el objeto de revisar la consulta previamente generada, se selecciona la segunda opción (i.e. **PREVISAR CONSULTAS ANTERIORES**), comenzando así una nueva sesión en **SADIN**:



Borrar los datos
 Continuar Consulta Nueva
 Borrar Consulta Anterior
 Nueva Consulta
 Borrar Consulta Anterior
 Borrar el Menú

SECO - Regresa al MENU AVANZADA

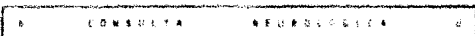
Enseguida, el sistema contesta con la siguiente pantalla, donde pide al usuario el directorio donde se encuentran los archivos de los pacientes



Darle la ruta donde se encuentra el arch. de los pacientes
 Borrar el arch. de los pacientes correspondiente en la ventana

Directorio de Archivos: /

Al verificar y localizar el directorio, SADIN despliega en un menú de selección los nombres de los archivos.



Darle la ruta donde se encuentra el arch. de los pacientes
 Borrar el arch. de los pacientes correspondiente en la ventana

Archivos:
 RAMIREZ LIS - PEREZ LIS

Es inmediatamente después a la elección de uno de ellos, pregunta al usuario por el dispositivo de salida al cual va a enviar los resultados

D	CONSULTA	NEUROLÓGICA	U
ELIJA DONDE DESEA VER LOS RESULTADOS			
Elige una opción en el menú de opciones siguientes o bien oprímelo (o presiona la tecla) de la palabra a elegir. Movimiento hacia ABAJO			
<input type="checkbox"/> Monitor <input type="checkbox"/> Impresora			

Acto seguido, SADIN muestra los datos del paciente (e. *nombre*, *fecha de nacimiento* y *sexo*), y la lista de las consultas que tiene, cada una de ellas encabezada por la fecha en que fue creada:

D	CONSULTA	NEUROLÓGICA	U
Consultas			
Paciente: JUAN BARRIZ		Sexo: M	
Fecha de nacimiento: 13 de SEP de 1966			
Consulta del 18 de ago de 1991			
Síntomas indicados en la consulta			
ADUEZAMIENTO DE LA MANO			
Signos indicados en la consulta			
SIGNO DE JUNE WERNICKE DE LA MANO			
SIGNO DE SABA			
DISTONIA PARA ABducir o cerrar			
FIN Terminar la Inspección		INSPECCION CON LAS FLECHAS DE MOVIMIENTO	

Como puede observarse en la secuencia, los archivos de consulta tienen consignada, como primer bloque, la lista de síntomas que el paciente presentó en esa consulta. Como segundo bloque se encuentra la lista de los signos arregiada en dos columnas: la de la izquierda tiene los signos que estuvieron involucrados en el proceso de diagnóstico, y en la de la derecha se indican si fueron presentados o no por el individuo.

Es inmediatamente después a la elección de uno de ellos, pregunta al usuario por el dispositivo de salida al cual va a enviar los resultados:

CONSULTA NEUROLÓGICA	
ELIGE DONDE DESEAS VER LOS RESULTADOS	
<p> <input type="checkbox"/> En el que estás en el menú principal «ESTADOS» <input type="checkbox"/> o bien, presionando la primera letra de la palabra a elegir. <input type="checkbox"/> Movimiento hacia ARRIBA <input type="checkbox"/> Movimiento hacia ABAJO </p>	
<p> <input type="checkbox"/> pantalla <input type="checkbox"/> impresora </p>	

Acto segundo, SADIN muestra los datos del paciente (i.e., nombre, fecha de nacimiento y sexo), y la lista de las consultas que tiene, cada una de ellas encabezada por la fecha en que fue creada:

CONSULTA NEUROLÓGICA	
Consultas	
<p> Paciente: JUAN BARRIOZ Sexo: M Fecha de nacimiento: 15 de SEP de 1966 </p>	
Consulta del 18 de mayo de 1991	
<p> Síntomas indicados en la consulta: ABLEGAZAMIENTO DE LA MANO SIGNOS INDICADOS EN LA CONSULTA: SIGNOS DE LA MANO DE LA MANO MANO EN CARRA DIBUJOS PARA ABOLUCION EL PUNJAR </p>	
<p> <input type="checkbox"/> TERMINAR LA INSPECCION </p>	<p> <input type="checkbox"/> PRESENTADO por el paciente <input type="checkbox"/> NO PRESENTADO por el paciente <input type="checkbox"/> NO PRESENTADO por el paciente </p>
<p> FIG Terminar la inspección INSPECCION CON LAS FLECHAS DE MOVIMIENTO </p>	

Como puede observarse en la secuencia, los archivos de consulta tienen consignada, como primer bloque, la lista de síntomas que el paciente presentó en esa consulta. Como segundo bloque se encuentra la lista de los signos arreglada en dos columnas: la de la izquierda tiene los signos que estuvieron involucrados en el proceso de diagnóstico, y en la de la derecha se indican si fueron presentados o no por el individuo.

Examinando esta serie de pantallas, se puede advertir la evolución del padecimiento. Las tres consultas se generaron el mismo día, pero los síntomas y los signos fueron cambiando (esto último para poner en relieve las bondades del sistema), de tal forma que el porcentaje estimado en los resultados cambió dependiendo de la información suministrada.

Nótese que al final de cada archivo se proporciona el dato de la cantidad de consultas que lo componen.

Concluida la inspección del archivo, el usuario oprime la tecla de función <F10> para regresar al sistema de menues, donde la pantalla principal le permite salir de SADIN.

```

> CONSULTA NEUROLÓGICA <
  
```

AUTORES

Esteban Montero F.
George Gelman E.

Copyright 1991
Montero, E. F.

Debes salir?

SI
NO

Selector

0. Salir al sistema de menues
 1. Movimiento de Archivos
 2. Salir de SADIN

Elige dentro del menu la accion a realizar .

SUPLEMENTO II

INTERFAZ DE MANTENIMIENTO

La primera pantalla que visualiza el usuario es aquella en la que se le pide proporcionar el directorio en el que reside la Base de Conocimiento. Obtenido el dato, el sistema le pide *con una segunda pantalla similar a la anterior*, el directorio en el cual van a residir los archivos temporales.

Hecho esto, SADIN da la bienvenida al usuario mostrando el siguiente escenario:

M A N T E N I M I E N T O

```

Elegir una opción:
1) Agregar datos
2) Borrar datos
3) Modificar base de conocimiento
4) Revisar base de conocimiento
5) Regresar al menú anterior

Escriba una opción:
1) Ingresar datos
2) Borrar datos
3) Modificar base de conocimiento
4) Revisar base de conocimiento
5) Regresar al menú anterior

Escriba una opción:
1) Ingresar datos
2) Borrar datos
3) Modificar base de conocimiento
4) Revisar base de conocimiento
5) Regresar al menú anterior
    
```

Escriba dentro del menú la acción a realizar:

Al igual que en la Máquina de Inferencias, este programa cuenta con un archivo de instrucciones en el que se muestra una breve introducción a su manejo. La segunda de las opciones, por su parte, nos permite entrar al sistema.

Nuevamente, un sistema de menús es el encargado de guiar al usuario, y el primero de estos nos muestra el siguiente panorama:

M A N T E N I M I E N T O

```

Elegir una opción:
1) Agregar datos
2) Borrar datos
3) Modificar base de conocimiento
4) Revisar base de conocimiento
5) Regresar al menú anterior
    
```

*Flecha Cancelar *RETORNO: Elegir *Flecha: Para mover el cursor

Si se elige la opción **REVISAR BASE DE CONOCIMIENTO**, SADIN continuará la ejecución presentando un menú donde pregunta al usuario si desea ver la totalidad de la Base de Conocimiento, o bien, sólo algunos netivos.

Cabe mencionar que en todas las pantallas de inspección, es posible, para mayor comodidad del usuario, generar un "zoom" o agrandamiento de la ventana, con el fin de que ocupe toda la superficie del monitor. Esto, como se muestra en la figura de abajo, es particularmente útil al examinar archivos extensos.

```

PANTALLA DE EL DEDO
-----
* Opciones para inspeccionar los dedos de la mano
* DESPLAZAR PARA MUESTRAR LA MANO
* DESPLAZAR PARA MUESTRAR EL DEDO

* Opciones para inspeccionar la mano
* DESPLAZAR PARA MUESTRAR LA MANO
* DESPLAZAR PARA MUESTRAR EL DEDO

* Opciones para inspeccionar el cuarto y quinto dedo de la mano
* DESPLAZAR PARA MUESTRAR EL CUARTO Y QUINTO DEDO
* DESPLAZAR PARA MUESTRAR EL CUARTO DEDO
* DESPLAZAR PARA MUESTRAR EL QUINTO DEDO

* Opciones para inspeccionar la falange
* DESPLAZAR PARA MUESTRAR LA FALANGE
* DESPLAZAR PARA MUESTRAR LA FALANGE DEL CUARTO Y QUINTO DEDO

* Opciones para inspeccionar la flexión
* DESPLAZAR PARA MUESTRAR LA FLEXION
* DESPLAZAR PARA MUESTRAR LA FLEXION DEL CUARTO Y QUINTO DEDO

* Opciones para inspeccionar la flexión
* DESPLAZAR PARA MUESTRAR LA FLEXION
* DESPLAZAR PARA MUESTRAR LA FLEXION DEL CUARTO Y QUINTO DEDO
  
```

Terminada la inspección, el usuario puede oprimir la tecla < F10 > para retornar al sistema de menús. Navegando en este último, se llega al primer menú de opciones para mostrar otro aspecto de SADIN referente al manejo que hace de elementos en su Base de Conocimiento.

Elegiendo esta vez la tercera opción, SADIN muestra el siguiente menú contenido todas las acciones posibles que, desde el programa de mantenimiento, se pueden efectuar sobre la Base de Conocimiento:

```

      N A V I G A R I O
-----
      Modificar Base de Conocimiento

      1. Ingresar una opción:
      2. Borrar opción
      3. Borrar grupo
      4. Borrar sistema
      5. Agregar opción
      6. Agregar sistema

      * * * * *
  
```

< F1 > Cancelar < RETURN > Egreso * * * * * Para mover el cursor

