

88362

IDRC-LB

88362

Grupo de Análisis de Sistemas Ecológicos

(asociado a Fundación Bariloche)

Proyecto Prospectiva Tecnológica de América Latina
(Subproyecto Prospectiva Ecológica de América Latina)

D E D U C

Sistema Interactivo para el procesamiento
deductivo de conceptos verbales

Guía para el usuario

Versión 1.0 para Apple II+ con CP/M-56K



Miguel Gross
Fundación Bariloche
Abril 1984

ARCHIV
681.3.02
DEDUC
G7

INDICE

A. INTRODUCCION

B. BIBLIOGRAFIA

C. EL SISTEMA DEDUC

1. Qué es DEDUC
2. La computadora, el sistema operativo v los diskettes
3. Como crear diskettes en blanco
4. Como generar réplicas (back-up) de modelos
5. Las unidades funcionales v sus ambientes
6. La operación de los ambientes

D. EL MODELO DEDUC

1. Los elementos del modelo
2. Como armar un modelo. Un ejemplo

E. LA UNIDAD DEDUC: DEFINICION DEL MODELO

1. Inicialización
2. Objetos
3. Implicaciones
4. Premisas

F. LA UNIDAD DEDUCA: PROCESAMIENTO Y ANALISIS DE CONCLUSIONES

1. Proceso deductivo
2. Conclusiones

G. LA UNIDAD DEDUCB: TRANSFERENCIAS ENTRE MODELOS DE OBJETOS E IMPLICACIONES

1. Modelo origen a archivo auxiliar
2. Archivo auxiliar a modelo destino

ANEXO. Cuadros de consulta rápida



A. INTRODUCCION

Cuando se intenta efectuar un análisis prospectivo de un sistema social, resulta extremadamente difícil aplicar los mismos métodos científicos utilizados en las ciencias naturales. Esos métodos sólo pueden describir muchos de los procesos que involucran al hombre de manera imprecisa, incompleta y distorsionada o ni siquiera pueden hacerlo.

Los sistemas sociales están determinados por acciones humanas que pueden alterar en cualquier instante su pauta de comportamiento. Esta se basa en el procesamiento de información cualitativa y simbólica. Para simular un sistema social deben reproducirse los procedimientos empleados por el hombre para seleccionar y procesar la información relevante a la toma de decisiones.

Personas con una similar base de conocimiento y un sistema semejante de valores toman decisiones parecidas en situaciones iguales. Por lo tanto las tendencias de comportamiento pueden estudiarse aún bajo nuevas condiciones, es decir no sólo como mera extrapolación, mediante un modelo cognitivo que representa el sistema conceptual y el mecanismo de procesamiento de los actores sociales. Estos son individuos o grupos de individuos que tienen un comportamiento homogéneo y que interactúan desarrollando objetivos comunes.

El hombre procesa conceptos con un cierto significado semántico de manera más o menos lógica. Tanto los conceptos como las cadenas lógicas de deducción pueden expresarse en lenguaje natural. Por tal motivo se prefiere un método cualitativo, no numérico, semejante al lenguaje natural, para reproducir los procesos cognitivos de deducción y valorización.

El modelo DEDUC, desarrollado por el Instituto de Análisis de Sistemas Aplicados y Pronóstico de Hannover, Alemania Federal, para simular procesos de deducción, se define mediante un lenguaje formal basado en la lógica simbólica ("predicate calculus"). Dicho instituto ha preparado también un programa interactivo de computadora para procesar modelos DEDUC.

Partiendo de la descripción del método y del programa (ver bibliografía a continuación) hemos desarrollado un sistema para la microcomputadora Apple II+ bajo el sistema operativo CP/M-56K. Este sistema cumple con casi todas las funciones del programa original. Únicamente se alteró el ambiente de trabajo para adecuarlo a la capacidad limitada de la Apple. Así el sistema debió dividirse en varios programas, distribuidos a su vez en varios diskettes según su función.

Esta guía consta de cinco partes. La primera describe el sistema, su composición y su operación. La segunda parte explica en qué consiste un modelo DEDUC. Si Ud. requiere mayor profundidad en el tema debe consultar la bibliografía, pues no es objetivo de este manual enseñar a construir un modelo de esta índole. Al final de ese capítulo podrá Ud. seguir un ejemplo completo. Es aconsejable que Ud. lea primero la descripción del sistema y de los modelos DEDUC y luego siga ese ejemplo en la computadora.

Las otras tres partes describen detalladamente los comandos que se utilizan en cada ambiente del sistema y cual es su función. Al final se resumen en varios cuadros todos los aspectos de DEDUC para la consulta rápida durante el procesamiento de modelos.

B. BIBLIOGRAFIA

- BOSEL, H. (Ed.) *Concepts and Tools of Computer-assisted Policy Analysis.* Vols. 1-3. 1977. Birkhäuser Verlag, Basel, Stuttgart.
- HORNUNG, B. R. *Qualitative Systems Analysis as a tool for Development Studies en Dependence and Inequality. A Systems Approach to the Problems of Mexico and other Developing Countries.* Geyer, R. F. & Van der Zouwen, J. (Eds.). Pergamon Press, Oxford, New York.
- BOSEL, H.; HORNUNG, B.; MÜLLER-REISSMANN, K. F. *A Basic Needs Approach to Impact Analysis and Evaluation for Long-term Policy Making. Nonnumerical Modelling with the Interactive Program DEDUC.*
- *Kognitive Systemanalyse. Konzeptverarbeitung und Verhaltenssimulation. Zusammenfassender Abschlussbericht. Teil IV.* Instituto de Análisis de Sistemas Aplicados y Pronóstico. 1982. Hannover.

C. EL SISTEMA DEDUC

1. Qué es DEDUC

DEDUC es un sistema interactivo para el procesamiento deductivo de conceptos verbales. Estos conceptos y las reglas de deducción son descritas mediante un modelo DEDUC. Este modelo está compuesto por objetos (estructurados jerárquicamente), implicaciones y premisas. Como resultado del proceso deductivo, que consiste en aplicar las implicaciones a las premisas, se obtienen las conclusiones. Estas pueden convertirse luego en premisas para nuevos procesos deductivos.

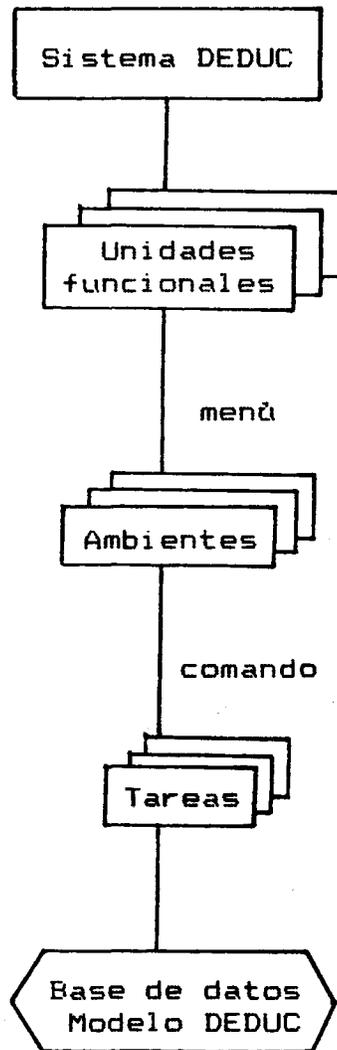
Las premisas y conclusiones son sentencias lógicas compuestas por un predicado y un conjunto de objetos. En el próximo capítulo se describirá detalladamente como se construye un modelo DEDUC.

El sistema DEDUC almacena cada modelo en una base de datos. Mediante las distintas funciones del sistema Ud. puede agregar elementos, eliminarlos, modificarlos, exhibirlos en pantalla, imprimirlos, etc. Para ello se utilizan comandos muy simples, cuyas palabras identificatorias se muestran en pantalla como ayuda-memoria.

Como los mismos comandos pueden aplicarse a distintos elementos de un modelo y su significado puede variar según el contexto en que se expresa, se ha dividido el sistema en ambientes. Así se evita la proliferación de comandos con sufijos y/o prefijos. Por ejemplo, existen ambientes para el manejo de objetos, de implicaciones, de premisas, etc.

Debido a la limitada capacidad de los diskettes los ambientes DEDUC se distribuyen en varias unidades funcionales, cada una de las cuales corresponde a un diskette. Para acceder a una unidad funcional, Ud. coloca el diskette correspondiente y tipea su nombre (que puede leerse en la etiqueta del diskette). Inmediatamente aparecerá en pantalla un menú que detalla los ambientes accesibles desde esa unidad funcional. Ud. selecciona el ambiente deseado, que es exhibido luego en pantalla. De ahí en adelante Ud. debe utilizar los comandos válidos para ese ambiente para efectuar la tarea deseada. Cuando Ud. da por terminada su tarea en ese ambiente, el sistema vuelve a exhibir el menú de la unidad funcional.

En síntesis, el sistema DEDUC está compuesto por unidades funcionales, desde las cuales Ud. selecciona un ambiente a partir de un menú, en el cual desarrolla una serie de tareas mediante comandos relativamente simples.

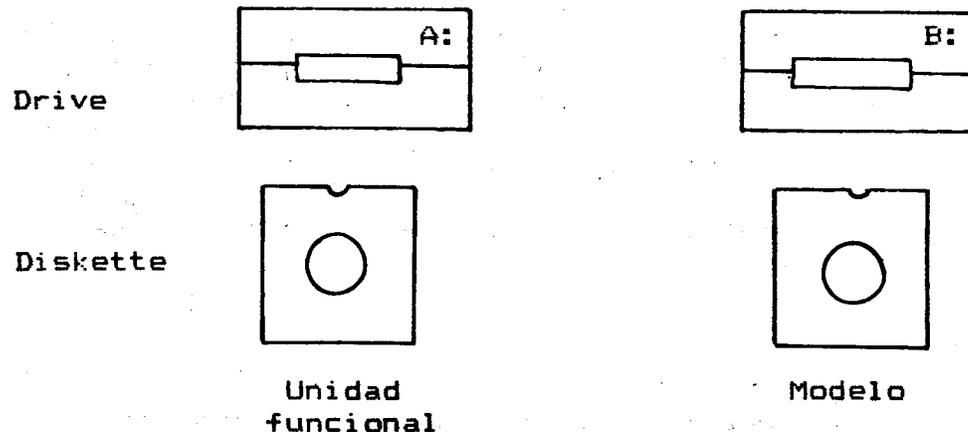


2. La computadora, el sistema operativo y los diskettes

El sistema DEDUC, cuya versión se describe aquí, funciona en el siguiente equipo:

- Computadora Apple II+ con 48K propios
- Tarjeta de expansión de memoria de 16K (RAMCARD)
- Procesador Z80A con CP/M-56K (Softcard)
- Tarjeta de video de 80 columnas (p.ej. VIDEX-Videoterm)
- Dos unidades de diskette de doble densidad
- Un monitor
- Una impresora

Cada unidad funcional se encuentra almacenada en un diskette que Ud. debe colocar siempre en el drive (unidad de diskette) A:, es decir en el que normalmente se ubica a la izquierda. De la misma manera, cada modelo DEDUC ocupa un diskette entero que debe colocarse siempre en el drive B: (a la derecha).



Inmediatamente después de reemplazar un diskette, sea para trabajar con otro modelo, sea para entrar en otra unidad funcional, Ud. debe advertir al sistema operativo de la nueva situación mediante el comando Ctrl-C. Este se obtiene manteniendo presionada la tecla CTRL y tipeando una C. Ud. advertirá que el drive A: estará activo durante unos instantes, al término de los cuales aparecerá en pantalla nuevamente la señal A> del sistema operativo.

Cuando Ud. advierta la señal A> del sistema operativo y que los drives se encuentran inactivos, significa que el sistema operativo espera que Ud. tipee el nombre de la unidad funcional instalada en el drive A:. Cuando haya finalizado de tipear el nombre, debe presionar la tecla RETURN (fin de mensaje). Inmediatamente se activará primero el drive A:, luego el drive B: y después aparecerá en pantalla el menú correspondiente.

Si en cambio Ud. no colocó el diskette apropiado, recibirá como respuesta el nombre tipeado y un signo de interrogación. Si por casualidad Ud. tipeó mal el nombre de la unidad funcional y éste coincide con alguno de los programas almacenados en el diskette, puede interrumpir el procesamiento manteniendo presionada la tecla CTRL y presionando la tecla RESET. Volverá a aparecer A>. Esto lo puede hacer en cualquier momento y en cualquier circunstancia, pero puede perder información, pues el sistema almacena en forma permanente recién al salir de un ambiente de trabajo la última información ingresada. Por lo tanto, sólo debe utilizar ese comando como último recurso.

Si en algún momento durante una sesión de trabajo aparece un mensaje como BDOS ERR significa que hubo un error de acceso a disco, ya sea por haberse dañado el diskette o por alguna interferencia en el sistema. Lo más probable, sin embargo, será que aparezca al comienzo de la sesión por no haberse colocado el diskette en el drive B:. Una vez hecho esto, se debe presionar Ctrl-C y volver a empezar. El mensaje NO FILE, en cambio, significa que el diskette en el drive B: no contiene un modelo DEDUC.

En síntesis:

- a) Verificar que siempre existan diskettes en ambos drive.
- b) Verificar que en drive A: siempre haya una unidad funcional y en drive B: un modelo o un diskette en blanco (para crear un nuevo modelo).
- c) Inmediatamente después de un reemplazo de diskettes, debe presionar Ctrl-C.
- d) Ante un mensaje de error del sistema operativo (XXXXX?, NO FILE ó BDOS ERR...) verificar si los diskettes son los correctos o si el comando tenía sentido (p.ej. no se puede trabajar con un modelo sin haberlo inicializado).
- e) No salir nunca de un ambiente, si no es por la vía del comando apropiado (Terminar).

3. Como crear diskettes en blanco

Para que un diskette pueda ser utilizado, debe Ud. someterlo al proceso de formateado. Para ello coloque el diskette maestro de CP/M (Softcard) en el drive A: y el diskette a formatear en el drive B:. Este diskette puede ser virgen o un diskette cuyo contenido va no le interesa conservar. A continuación ejecute la siguiente secuencia de comandos (ante cada exhibición de la señal A>):

```
Ctrl-C
```

```
FORMAT B:<RETURN>
```

Se le pedirá la inserción del diskette: conteste presionando <RETURN>. También recibirá una advertencia si el diskette no es virgen: conteste Y. Luego espere la señal del sistema operativo A> y tipee

```
COPY B:=A:/S<RETURN>
```

Conteste todas las preguntas con <RETURN>.

Cuando vuelva a aparecer A> habrá creado un diskette en blanco listo para recibir un modelo DEDUC. Coloque una etiqueta con la fecha, la leyenda CP/M-56K y la identificación que Ud. prefiera.

4. Como generar r plicas (back-up) de modelos

Para evitar los problemas suscitados por interrupciones no normales de una sesi n de trabajo o por la fragilidad de los diskettes, es conveniente poseer r plicas de los diskettes que contienen los modelos. Tambi n se pueden generar r plicas para efectuar experimentos con los modelos, manteniendo la versi n original de los mismos.

Para generar una r plica debe colocarse en el drive A: el diskette maestro de CP/M y tipear luego de cada A>

Ctrl-C

COPY B:=A:<RETURN>

El computador pedir  que inserte el diskette a copiarse en el drive A: y el diskette que deber  contener la r plica en el drive B:. Este debe ser un diskette formateado. Cualquier informaci n que haya contenido antes, ser  borrada. Luego presione <RETURN> y espere un nuevo mensaje del computador, que preguntar  si desea hacer m s copias del mismo diskette. Proceda seg n corresponda. Recuerde etiquetar los nuevos diskettes.

5. Las unidades funcionales y sus ambientes

El sistema DEDUC está compuesto por tres unidades funcionales. Cada uno de ellos se invoca colocando el correspondiente diskette en el drive A:, presionando Ctrl-C y tipeando su nombre seguido por <RETURN>.

Una vez aparecido el menú de ambientes, se selecciona el deseado presionando la tecla con el número correspondiente. Un cero provoca la inmediata aparición de la señal A> (y un mensaje que recuerda presionar Ctrl-C luego de un reemplazo de diskettes).

El siguiente cuadro muestra las unidades funcionales con sus ambientes:

DEDUC = Almacenamiento y recuperación de un modelo

- 1 Inicialización de un nuevo modelo
- 2 Objetos
- 3 Implicaciones
- 4 Premisas

DEDUCA = Procesamiento de un nuevo modelo

- 1 Proceso deductivo
- 2 Conclusiones

DEDUCB = Transferencias entre modelos de objetos e implicaciones

- 1 Modelo origen a archivo auxiliar
- 2 Archivo auxiliar a modelo destino

La operación de los ambientes

Salvo la inicialización, que sólo consulta el nombre del modelo, el proceso deductivo y la transferencia de elementos del archivo auxiliar a un modelo, que son totalmente automáticos, los restantes ambientes son interactivos. Dentro de cada uno de ellos Ud. se maneja mediante comandos DEDUC.

Así como el sistema operativo utiliza A> como señal de espera de comandos CP/M, DEDUC tiene también un símbolo de espera:

(numeral)

Esta señal es colocada en la pantalla cuando DEDUC finaliza una tarea y espera que Ud. le indique una nueva.

La estructura de un comando DEDUC es muy simple:

Nombre <comando>

Cada comando DEDUC empieza con un nombre que lo identifica. El cuerpo del comando depende del ambiente en que se lo utiliza y será explicado en los capítulos correspondientes. Por ejemplo,

IMPRIMIR 5

en el ambiente de las Premisas significa listar en la impresora la premisa identificada por ese número.

Notará Ud. que, cuando DEDUC emite su símbolo de espera, #, simultáneamente exhibe en la parte superior derecha de la pantalla los nombres de todos los comandos disponibles para ese ambiente como ayuda-memoria.

DEDUC no procesará un comando mientras Ud. no presione la tecla <RETURN> señalando el fin de mensaje. Ud. tendrá así oportunidad de revisarlo y corregir algún error que advierta. La única forma de corregir un error en un comando DEDUC es mediante la tecla de retroceso, <- , cuya acción consiste en retroceder el cursor hasta la posición deseada, borrando los caracteres que encuentra en su desplazamiento.

El comando DEDUC acepta hasta 500 teclas presionadas incluyendo retrocesos y el fin de mensaje. Esto significa que el comando puede ocupar más de una línea.

Una vez que Ud. haya presionado la tecla <RETURN>, DEDUC analizará su comando. Cualquier error o inconsistencia que encuentre, le será advertida inmediatamente mediante un mensaje. En algunas ocasiones le será indicado el lugar preciso en el comando, donde fue descubierto el problema. A continuación volverá a aparecer el símbolo #.

Ud. no podrá corregir el comando, ya que DEDUC no posee un editor de comandos. Deberá tipearlo nuevamente, previo análisis de la causa del error.

En cambio, si DEDUC acepta el comando, lo ejecutará inmediatamente. Esto puede demorar mayor o menor cantidad de tiempo según el comando, la estructura del modelo en ese momento y la situación en que se emite (pues DEDUC utiliza parte de la memoria central como almacenamiento rápido provisorio que al comienzo de la operación de un ambiente está vacío).

DEDUC utiliza la pantalla agregando líneas en forma vertical. Una vez llenas las 24 líneas, sólo utilizará la inferior para la exhibición de nuevos mensajes, avanzando hacia arriba el resto de la pantalla. Así las líneas superiores desaparecen. Las únicas excepciones son los menús de las unidades funcionales, el inicio de un ambiente y el ayuda-memoria de comandos. Existen casos en que la cantidad de información que se ha requerido excede el espacio de pantalla. Por tal motivo puede ocurrir que Ud. no haya tenido tiempo de examinar las primeras líneas antes de su desaparición por la parte superior. CP/M provee para ello un comando de interrupción. Presionando Ctrl-S puede Ud. congelar la salida. Presionando luego cualquier tecla el listado continúa.

La impresora debe ser prendida y puesta en línea antes de emitir el primer comando de impresión, pues en caso contrario perderá toda la información nueva que Ud. haya ingresado durante la operación del ambiente en que se produjo el incidente. El control del sistema se recupera con un Ctrl-RESET, pero fuera de la unidad funcional que debe ser invocada nuevamente.

Ud. apreciará también que la impresora demora la emisión de la última línea. Con cada comando de impresión se listará la línea pendiente y las del nuevo comando. Cuando Ud. sale del ambiente, se imprimirá la última línea del último comando de impresión.

Con el primer comando de impresión de un ambiente se imprime un encabezamiento con el nombre del modelo.

D. EL MODELO DEDUC

1. Los elementos del modelo

El sistema DEDUC está basado en el procesamiento de conceptos verbales. Esos conceptos se integran en un modelo cognitivo que puede utilizarse para simulaciones cualitativas. Los elementos de ese tipo de modelos son los siguientes:

- Objetos
- Estructuras de objetos
- Predicados
- Premisas
- Implicaciones

A continuación los describiremos sintéticamente. Para un estudio más profundo de los modelos DEDUC debe Ud. consultar la bibliografía.

Objetos:

Son entidades definidas en forma cualitativa, p.ej. "Ambiente", "Materia prima", "Economía", "Tiempo", "País", "Región", "Presente", "Futuro", etc.

Estos objetos corresponden a las variables de estado de los modelos de simulación numéricos. A diferencia de éstos no se les asigna un valor. En cambio forman parte de sentencias que declaran un estado cualitativo de los mismos.

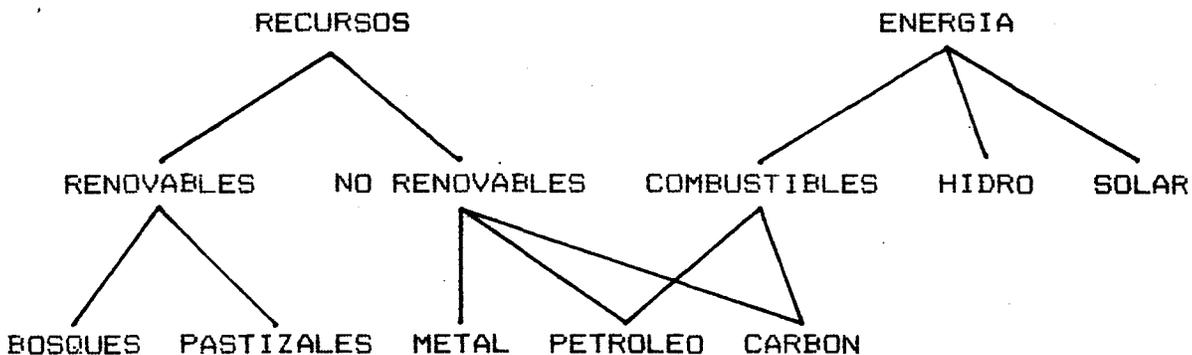
Estructuras de objetos:

Un objeto puede representar a un grupo de objetos. En el ejemplo de arriba "Presente" y "Futuro" forman parte del conjunto "Tiempo". Así, cualquier referencia a "Tiempo" se hace extensiva a cada uno de sus integrantes a menos que se indique expresamente que la referencia es para el conjunto en forma global.

Como se verá más adelante, las implicaciones interpretan los objetos como grupos. El proceso deductivo trabaja con los individuos del grupo o con todo el grupo según las circunstancias. De esta manera se efectúa no sólo una deducción horizontal, que consiste en evaluar el lado izquierdo de la implicación y obtener como resultado el lado derecho, sino también una deducción vertical a través de las estructuras de objetos.

Las estructuras pueden consistir de varios niveles. Un objeto, que forma parte de un grupo, puede representar a su vez un conjunto de objetos. También se permite que un objeto forme parte de más de un grupo. Debe prestarse atención, sin embargo, que no se produzca una incongruencia, es decir, que a través de las relaciones entre objetos uno pertenezca a sí mismo.

Un ejemplo de estructura es el siguiente:



Un concepto importante es la intersección de objetos, operación fundamental del proceso deductivo. La intersección de dos objetos son los objetos contenidos en ambos al máximo nivel posible. Con la estructura de arriba tenemos los siguientes ejemplos:

$\text{INT}(\text{RECURSOS}, \text{RENOVABLES}) = \text{RENOVABLES}$

$\text{INT}(\text{BOSQUES}, \text{METAL}) = \text{vacía}$

$\text{INT}(\text{ENERGIA}, \text{ENERGIA}) = \text{ENERGIA}$

$\text{INT}(\text{ENERGIA}, \text{RECURSOS}) = \text{PETROLEO}, \text{CARBON}$

DEDUC aplica continuamente esta operación para ir de lo general, definido en las implicaciones, a lo especial contenido en las premisas. También lo utiliza para aplicar las conclusiones a los objetos que realmente corresponden.

Predicados:

Describen el estado cualitativo de objetos, p.ej. Creciendo, Contaminado, Exporta, Escaso, Caro, Desarrollo, etc. Algunos de los predicados pueden utilizarse también como objetos. Lo importante es que Ud. defina claramente qué significa cada cosa.

El predicado puede aplicarse a uno o más objetos. Un predicado con un conjunto ordenado de objetos forma una sentencia. DEDUC maneja las sentencias mediante implicaciones y premisas. La sentencia está formada por un predicado y una n-upla de objetos.

predicado (objeto 1, objeto 2, . . . , objeto n)

Las distintas sentencias que se forman con un mismo predicado deben tener siempre la misma cantidad de objetos y, los objetos ubicados en la misma posición deben pertenecer a un mismo conjunto de objetos aunque no necesariamente del mismo nivel.

Por ejemplo es correcto decir:

ESCASO (METAL)

ESCASO (PETROLEO)

ESCASO (RECURSOS)

Estas son sentencias compatibles pues los tres objetos tienen un significado afín.

No tiene sentido en cambio

ESCASO (TIEMPO)

pues se refiere a otro tipo de objetos. En este caso debe Ud. definir otro predicado.

La compatibilidad a que hacemos referencia aquí puede no ser tal en el momento de evaluar una implicación. Por ejemplo, ESCASO(METAL) v ESCASO(PETROLEO) son dos expresiones que no tienen punto de contacto, pero a priori son legítimas pues se refieren a la falta de RECURSOS. Lo importante es la correcta definición del predicado.

Un ejemplo con más de un objeto es:

FALTA(METAL, TIEMPO)

FALTA(PETROLEO, FUTURO)

En cambio, no es válido: FALTA(HOY, RECURSOS) pues se invirtió el orden de los objetos. Tanto METALES - PETROLEO como TIEMPO - FUTURO están relacionados, pero en la tercer sentencia no existe relación HOY - METALES ó RECURSOS - TIEMPO.

DEDUC exige la consistencia de las sentencias pues no analiza su contenido. Si existe inversión de orden en alguna de ellas, DEDUC puede rechazar incorrectamente la evaluación de una implicación por no encontrar una sentencia apropiada.

La importancia del orden se puede apreciar mejor en la siguiente sentencia:

EXPORTA(PAIS-A, PAIS-B, PRODUCTO, TIEMPO)

que significa "el país A exporta al país B un PRODUCTO en determinado periodo de TIEMPO". En esta situación se podrían producir confusiones entre el país comprador y el país vendedor mediante sentencias incorrectas pero estructuralmente válidas. Por ejemplo

EXPORTA(Argentina, URSS, Granos, hoy)

y EXPORTA(URSS, Argentina, Granos, hoy)

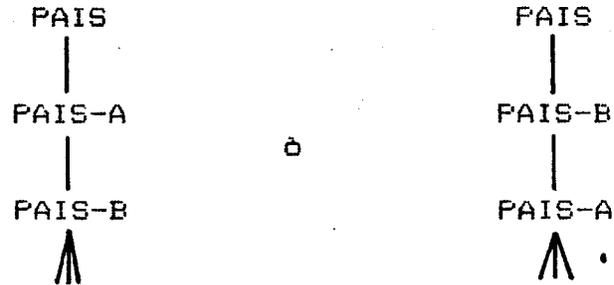
son aceptables para DEDUC y probablemente obtenga conclusiones extrañas, pero sólo la primera es verdadera.

Este ejemplo es útil además para observar otro aspecto a tener en cuenta. No deben repetirse objetos en una sentencia.

Si se definia

EXPORTA(PAIS.PAIS.PRODUCTO.TIEMPO)

no se lograba el significado esperado. Esta expresión implica que cada vez que DEDUC genera una sentencia con el predicado EXPORTA repetirá el mismo objeto en los primeros dos lugares. Debe enfatizarse que se trata de dos países distintos, A y B, y luego colocarlos en la estructura como



PAIS-A y PAIS-B son objetos auxiliares.

Premisas:

Una sentencia se refiere a una situación que puede ser verdadera, falsa o indeterminada. Son utilizadas en las implicaciones para definir los mecanismos de deducción, es decir, la estructura del modelo (juntamente con la estructuración de los objetos). Mediante las premisas Ud. activa algunas sentencias asignándoles un valor, verdadero o falso. La premisa describe el estado de objetos determinados. Define una sentencia y le asigna un valor.

Por ejemplo:

EXPORTA(Argentina,URSS,Granos,Tiempo)

NO ESCASO(Granos,Argentina,hoy)

Las premisas corresponden a los valores iniciales, umbrales o parámetros de los modelos de simulación numérica. Constituyen el punto de partida para el proceso deductivo. A ellas se les aplican las implicaciones para obtener las conclusiones que también son sentencias con un valor determinado. Las conclusiones que va generando el proceso deductivo intervienen en los sucesivos pasos del mismo, como si fueran premisas, pero no participan en futuros procesos deductivos a menos que sean transformadas explícitamente en premisas.

Como Ud. puede no estar muy seguro del valor de la declaración que Ud. expresa mediante una premisa. DEDUC le permite asignarle un factor de certeza (FC). Este factor, cuyo rango es 0 : 100, no debe ser confundido con el concepto de probabilidad. No hace ninguna afirmación acerca de la probabilidad de ocurrencia del suceso expresado en la premisa. Sólo se refiere al valor subjetivo de la expresión, a la certeza que Ud. tiene de lo que está expresando con la premisa.

El factor de certeza no influye en el proceso deductivo, sólo caracteriza a las conclusiones, en donde el valor no será nunca mayor que el de las sentencias que las originaron.

Implicaciones:

Describen las relaciones entre sentencias mediante la cláusula

SI <expresión> ENTONCES <conclusiones>

La expresión es una expresión lógica que tiene sentencias como operandos y utiliza los operadores lógicos NO, Y y O. Los operadores son evaluados en ese mismo orden de precedencia y de izquierda a derecha para operadores iguales. Ud. puede alterar el orden mediante el uso de paréntesis como en una expresión algebraica.

Por ejemplo:

"Si el país A exporta al país B un producto en cualquier instante, entonces el país A mejora sus ingresos en ese instante."

SI EXPORTA(PAIS-A,PAIS-B,PRODUCTO,TIEMPO) ENTONCES
MEJORA(INGRESO,PAIS-A,TIEMPO);

"Si un país mejora sus ingresos en cualquier instante y no aumenta su importación, entonces desarrolla su economía."

SI MEJORA(INGRESO,PAIS,TIEMPO) Y NO
AUMENTA(IMPORTACION,PAIS,TIEMPO) ENTONCES
DESARROLLA(ECONOMIA,PAIS,TIEMPO);

Es importante entender como funciona el proceso deductivo vertical. Las implicaciones hacen referencia a grupos de objetos. Cada uno de esos grupos será representado por un único objeto (que puede ser el mismo) durante la evaluación de la implicación. Este objeto debe ser el mismo para todas las veces en que aparece mencionado el grupo en las diversas sentencias y se proyecta a las conclusiones.

Si, por ejemplo, tenemos como premisas

MEJORA(INGRESO,PERU,HOY)

NO AUMENTA(IMPORTACION,PERU,FUTURO)

resulta que al objeto TIEMPO le corresponde en la primer sentencia el objeto HOY y en la segunda FUTURO. La intersección de ambos objetos es vacía. Por tal motivo el valor de la expresión es indeterminado y no se logra una conclusión.

Si las premisas hubiesen sido

MEJORA(INGRESO,PERU,TIEMPO)

NO AUMENTA(IMPORTACION,PERU,FUTURO)

podríamos asignar al grupo TIEMPO la intersección de los objetos TIEMPO y FUTURO, que resulta ser el mismo FUTURO. La conclusión sería

DESARROLLO(ECONOMIA,PERU,FUTURO)

Si hubiésemos tenido cuatro premisas, dos para HOY y dos para FUTURO, la implicación se hubiese evaluado dos veces, ya que ambos son válidos, y se hubiesen obtenido dos conclusiones.

Las implicaciones son válidas para cada uno de los objetos abarcados por los conjuntos de objetos mencionados en ellas.

Si llamamos X_i a todos los objetos (grupos) que aparecen en una implicación $P(X_1, X_2, \dots)$, el significado concreto de ésta es el siguiente:

$P(X_1, X_2, \dots)$ es válida para todo conjunto de objetos x_i , tal que cada uno pertenece al X_i correspondiente.

Si volvemos a tomar nuestro segundo ejemplo, la implicación es válida para cualquier conjunto de 4 objetos, uno perteneciente al grupo INGRESO, uno a IMPORTACION, uno a PAIS y uno a TIEMPO.

Si Ud. desea que una sentencia sólo tenga validez para el grupo completo que integra un objeto, es decir que sólo sea cierta si es cierta para todos los objetos del grupo, DEDUC le permite señalar tal situación con un ":" delante del nombre del objeto. Si hubiese colocado esa señal delante del objeto INGRESO, no hubiera habido ningún efecto, pero delante de PAIS hubiese impedido la deducción pues PERU no contiene ni es igual a PAIS. Los dos puntos tienen el efecto de cortar momentáneamente la relación del objeto con sus descendientes en la estructura.

Las conclusiones pueden recibir el valor FALSO, por ejemplo

SI AUMENTA(PRECIO) ENTONCES NO INCREMENTA(INGRESO)

En la presente versión DEDUC no aplica las implicaciones por la negativa. Si la forma normal de una implicación es

SI A ENTONCES B

y existe la premisa

NO B .

debería concluirse

NO A

DEDUC no lo provee, por lo que Ud. debería definir la inversa de las implicaciones como nuevas implicaciones si así lo desea.

También se puede asignar un factor de certeza a las implicaciones. Tiene el mismo significado que en las premisas. El mecanismo de evaluación del factor de certeza sigue las siguientes reglas:

- a) $FC(P \vee Q) \leq \min(FC(P), FC(Q))$
 b) $FC(P \wedge Q) \leq \max(FC(P), FC(Q))$
 c) $FC(R) \leq FC(L) \times FC(\text{SI } L \text{ ENTONCES } R) / 100$

donde P, Q y L son sentencias (reales o intermedias del proceso de evaluación de la expresión lógica); R es una conclusión.

La negación no altera el factor de certeza.

También se puede definir un coeficiente de impacto para cada implicación, cuyo rango es -100 : 100. Sólo tiene sentido en implicaciones del tipo

SI sentencia ENTONCES sentencia

que relacionan conclusiones de la parte cognitiva del modelo con predicados correspondientes a la parte valorativa. Por eso son implicaciones más simples. Los predicados de esta sección del modelo son de tipo abstracto y cumplen el papel de "orientadores" (ver bibliografía).

Por ejemplo

SI INCR-DESOCUPACION(PAIS, TIEMPO) ENTONCES
 AFECTA(ESTABILIDAD); CI -80

"Un incremento de la desocupación afecta la estabilidad en el empleo en forma negativa en un 80%".

El impacto sobre una conclusión se calcula como

$$CI(R) \leq CI(L) \times CI(\text{SI } L \text{ ENTONCES } R) / 100$$

donde R y L son sentencias.

El factor tiempo:

El tiempo es un factor importante en los modelos de simulación. Para DEDUC es un objeto más, lo que implica que sólo puede manejarse en forma discretizada. El tiempo puede manejarse como un conjunto de objetos, cada uno de los cuales representa un periodo. Pero el sistema DEDUC en su presente versión no maneja sucesiones. Un conjunto de objetos es interpretado como tal en el estricto sentido matemático del término conjunto, es decir no hay orden en sus elementos. Si Ud. desea que las implicaciones tengan validez para pares consecutivos de periodos temporales debe replicar esas implicaciones para cada par.

2. Como armar un modelo. Un ejemplo.

El propósito de esta guía no es enseñar el método DEDUC, que, por otra parte, es aún experimental. El objetivo es que Ud. conozca a fondo como funciona el sistema y aprenda a manejarlo. Antes de entrar en los detalles para cumplir ese fin, se desarrolla a continuación un ejemplo sencillo que ilustra la construcción de un modelo y el uso del sistema. Adjuntamos a la descripción el listado del contenido final del modelo, obtenido mediante los comandos IMPRIMIR TODO de cada ambiente.

El ejemplo que desarrollamos está basado en el artículo de G.C. Gallopín *Development and Environment. An Illustrative Model* (Journal of Policy Modeling 2(2), 239-254, 1980). En ese artículo se presenta un sencillo modelo matemático del ambiente que ilustra la relación entre el desarrollo económico y el ambiente. Hemos extraído algunos conceptos con el único fin de mostrar el funcionamiento de DEDUC. Son, por supuesto, muy escasos como para representar una situación real, pero así puede Ud. controlar el proceso deductivo del ejemplo.

Es aconsejable que Ud. siga en la computadora paso a paso el proceso que le describimos a continuación:

- a) Encienda la computadora.
- b) Tome el diskette del sistema CP/M (Master diskette Softcard) y un diskette virgen; siga el procedimiento para preparar un diskette en blanco descrito en C.3. Deje el diskette en blanco en el drive B:.
- c) Tenga a mano los diskettes DEDUC y DEDUCA para usar sucesivamente en el drive A:.
- d) Coloque en el drive A: el diskette DEDUC y presione CTRL-C. Luego tipee DEDUC<RETURN>. Ud. verá ahora el menú de la unidad de definición del modelo.
- e) Como debe crear un nuevo modelo, tipee 1.
- f) El sistema le preguntará a continuación por el nombre del modelo. Tipee, por ejemplo, GUIA<RETURN>.

La inicialización demora algo más de 4 minutos. después de los cuales volverá a aparecer el menú de la unidad funcional. Espere que el sistema le solicite un nuevo ítem a seleccionar.

- g) Cuando vuelva a titilar el cursor después de la palabra CUAL?, tipee 3. Con ello ingresará en el ambiente de implicaciones. El sistema limpiará la pantalla, colocará un título y el símbolo #. A la derecha Ud. verá la lista de comandos DEDUC disponibles.

Ahora codificaremos los conceptos verbales que hemos preparado en base a la lectura del texto mencionado.

"Si la calidad ambiental es mayor que la capacidad de carga de un sistema ecológico, esa calidad disminuirá pero no será inferior a la capacidad (es decir, tiende a ella)".

Típee (no se preocupe por el fin de línea):

```
SI MAYOR(AMBIENTE,CARGA,SISTEMA):DISMINUYE(AMBIENTE,SISTEMA),
NO MENOR(AMBIENTE,CARGA,SISTEMA)<RETURN>
```

A continuación aparecerá el mensaje IMPLICACION 1 y el símbolo #. Esto significa que esa implicación se identifica por el número 1.

Si Ud. notó un error de tipeo antes de presionar <RETURN> retroceda con la tecla <- . Si lo notó después, tipee ELIMINE 1<RETURN> y vuelva a tipear la línea completa.

En esta implicación tenemos tres objetos con el siguiente significado:

```
AMBIENTE = calidad ambiental
CARGA    = capacidad de carga
SISTEMA  = sistema ecológico
```

El significado de los tres predicados surge de la expresión verbal.

La palabra ENTONCES se reemplaza por un ":" que es más breve.

Los objetos y predicados se representan mediante palabras de hasta 8 caracteres. Si exceden esa cantidad, las letras sobrantes son ignoradas como Ud. notará tipeando MOSTRAR 1<RETURN> (DISMINUY). Los caracteres pueden ser letras mayúsculas o minúsculas (usar Ctrl-A para pasar de una a otra), dígitos o símbolos (excepto paréntesis, coma, punto y coma, dos puntos). El primer carácter debe ser letra.

Pasemos ahora rápidamente a las otras implicaciones.

"Si la calidad ambiental de un sistema ecológico es menor que su umbral de extinción, habrá un colapso de aquella".

Típee:

```
SI MENOR(AMBIENTE,EXTIN,SISTEMA):COLAPSO(AMBIENTE,SISTEMA)
<RETURN>
```

"Si la calidad ambiental de un sistema es menor que su capacidad de carga y mayor que su umbral de extinción, la calidad crecerá pero no será mayor que la capacidad de carga".

Como podrá ver hemos obviado las relaciones temporales. Tipee ahora:

```
SI MENOR(AMBIENTE,CARGA,SISTEMA) Y MAYOR(AMBIENTE,EXTIN,
SISTEMA):AUMENTA(AMBIENTE,SISTEMA),NO MAYOR(AMBIENTE,CARG
A,SISTEMA)<RETURN>
```

Los espacios en blanco sólo tienen importancia como separadores. Deben colocarse siempre que pueda existir confusión, por ejemplo entre SI y MENOR, después del Y y del NO, etc.

Trate de provocar errores para ver la reacción del sistema, p.ej. olvidando la palabra SI, olvidando paréntesis, el ":", etc.

"Si no aumenta el desarrollo en el sistema ecológico, no crecerá la calidad de vida en el sistema".

```
SI NO AUMENTA(DESARROLLO,SISTEMA):NO CRECE(VIDA,SISTEMA)
<RETURN>
```

"Si el desarrollo aumenta y no hay colapso ambiental, crecerà la calidad de vida".

SI AUMENTA (DESARROLLO, SISTEMA) Y NO COLAPSO (AMBIENTE, SISTEMA): CRECE (VIDA, SISTEMA) <RETURN>

"Si el desarrollo crece y es negativo para el sistema ecológico, disminuye la calidad ambiental".

SI NEGATIVO (DESARROLLO, SISTEMA) Y AUMENTA (DESARROLLO, SISTEMA): DISMINUYE (AMBIENTE, SISTEMA) <RETURN>

"Si el desarrollo crece y es positivo para el sistema ecológico, aumenta la calidad ambiental".

SI POSITIVO (DESARROLLO, SISTEMA) Y AUMENTA (DESARROLLO, SISTEMA): CRECE (AMBIENTE, SISTEMA) <RETURN>

"Si el umbral de extinción del sistema ecológico es alto y la calidad ambiental es menor que la capacidad de carga del sistema, entonces la calidad está cerca de la extinción".

SI ALTO (EXTIN, SISTEMA) Y MENOR (AMBIENTE, CARGA, SISTEMA): CERCA (AMBIENTE, EXTIN, SISTEMA) <RETURN>

"Si la calidad ambiental está cerca de la extinción y disminuye, entonces cae por debajo del umbral".

SI CERCA (AMBIENTE, EXTIN, SISTEMA) Y DISMINUYE (AMBIENTE, SISTEMA): MENOR (AMBIENTE, EXTIN, SISTEMA); <RETURN>

"Si el umbral de extinción del sistema ecológico es bajo, la calidad ambiental no se acerca a ese umbral".

SI BAJO (EXTIN, SISTEMA): NO CERCA (AMBIENTE, EXTIN, SISTEMA) <RETURN>

Para ver lo que ha escrito debe utilizar el comando MOSTRAR. Como Ud. puede apreciar en la parte superior derecha de la pantalla, sólo la M de Mostrar está en mayúsculas. Por lo tanto cualquier palabra que empieza con esa letra será interpretada como el comando MOSTRAR.

Típee:

MUESTRE TODO

Recuerde que después de cada comando debe presionar la tecla <RETURN>. No volveremos a incluirla en el texto. Las únicas excepciones son el Ctrl-C, que es de acción instantánea, y la selección de ambientes, donde el sistema sólo espera un dígito.

Apreciará Ud. que las líneas sucesivas se moverán hacia arriba. Cuando exhiba la implicación 7, desaparecerá la primera. Ud. puede detener el movimiento de la pantalla con un Ctrl-S. Eso congelará todo el sistema. Si Ud. desea seguir viendo, presione cualquier tecla.

Para salir del ambiente de implicaciones, típee

TERMINAR

Nuevamente aparecerá el menú de la unidad DEDUC.

- h) El texto que estamos interpretando para este ejemplo caracteriza dos tipos de sistemas ecológicos en función de la relación entre sus parámetros fundamentales, capacidad de carga y umbral de extinción. Definiremos entonces una estructura para el objeto SISTEMAS.

Típee 2

Ahora se encuentra Ud. en el ambiente de objetos.

Típee:

DEFINA SISTEMAS:SIMPLES,COMPLEJOS

Los sistemas simples pueden ser, por ejemplo, pestes, especies invasoras, sucesiones, etc.

Típee:

DEF SIMPLES:PESTES,INVASORAS,SUCESION

Los sistemas complejos pueden ser naturales (p.ej. bosques tropicales, ecosistemas abisales) o artificiales (p.ej. cultivos, pastizales).

Tipee:

```
DEFINIR COMPLEJOS:NATURALES(BOSQUES,ABISAL),ARTIFICIALES(CULTIVOS,PASTIZALES)
```

Para ver la estructura, tipee

```
MOSTRAR SISTEMAS
```

Ahora tipee TERMINAR, para salir del ambiente y volver al menú principal.

i) Definiremos ahora algunas premisas. Tipee 4.

En primer lugar hay dos premisas básicas que caracterizan los dos tipos de sistemas. Los complejos tienen un umbral de extinción alto y los simples uno bajo.

Tipee:

```
DEF ALTO(EXTIN,COMPLEJO)
```

```
D BAJO(EXTIN.SIMPLES)
```

j) Agregaremos ahora dos premisas más específicas para ensayar una corrida del modelo.

Tipee:

```
DEF AUMENTA(DESARROLLO,PASTIZAL)
```

```
DEF NEGATIVO(DESARROLLO,PASTIZAL)
```

Es decir, introducimos un desarrollo negativo en el pastizal.

Tipee ahora MOSTRAR TODAS y salga del ambiente de premisas con TER.

- k) Ahora deseamos ejecutar el proceso deductivo. Para ello hay que pasar a otra unidad.

Típee 0 (cero)

- l) Extraiga el diskette DEDUC del drive A: y coloque en su reemplazo DEDUCA. Presione Ctrl-C.

Ahora típee DEDUCA

Verá el menú de la unidad de procesamiento.

Típee 1

- m) Mientras se está procesando el modelo, Ud. verá aparecer los números de las implicaciones que están siendo procesadas. Sólo se verán las que tienen alguna posibilidad de generar conclusiones.

Ud. habrá visto que el sistema terminó rápidamente sin otro mensaje, lo que significa que no hubo conclusiones. El sistema no volvió al menú de la unidad, sino directamente al ambiente de conclusiones, aunque en este caso inútilmente.

Igual típee MOSTRAR TODO para confirmar la inexistencia de conclusiones. A continuación típee TERMINAR para volver al menú de la unidad.

- n) Qué pasó? Por qué no hay conclusiones cuando debería haberlas? El problema fue que definimos la estructura SISTEMAS pero en las implicaciones aparece el objeto SISTEMA. Para DEDUC son dos objetos distintos. No pudo relacionar entonces este objeto con los componentes del conjunto SISTEMAS.

Para arreglar la situación lo más rápidamente posible, relacionamos SISTEMA con SISTEMAS.

Típee 0 para salir de la unidad.

- o) Coloque nuevamente el diskette DEDUC. Presione Ctrl-C y típee DEDUC. Entre al ambiente de objetos y típee

DEF SISTEMA:SISTEMAS

Luego tipee TERMINAR y en el menù principal 0 para cambiar de unidad.

- p) Coloque diskette DEDUCA, presione Ctrl-C y tipee 1.

Verà Ud. que el sistema recorrerà varias veces el conjunto de implicaciones generando conclusiones, en este caso 2. Cuando vuelva al ambiente de conclusiones, tipee MOSTRAR TODO.

- q) Tipee ahora

ORIGEN 5

Ud. verà en pantalla las implicaciones y premisas (o conclusiones) que puedan haber dado origen a la conclusión 5.

- r) Efectùe ahora algunas correcciones el modelo. Para ello elimine primero las conclusiones para no interferir el nuevo procesamiento.

Tipee ELIMINAR TODO

Luego TERMINAR

Tipee 0 para cambiar unidades.

- s) Coloque DEDUC y presione Ctrl-C. Tipee DEDUC.

- t) Agregaremos factores de certeza a dos implicaciones. Entre a ese ambiente primero.

Daremos un FC de 90 a la implicación 2 y de 70 a la 9.

Tipee CAMBIAR 2 FC 90

Ahora pida ver esa implicación

MOSTRAR 2

Típee CAMBIAR 9 FC 70

MOSTRAR 9

Ahora TERMINAR

Entre ahora al ambiente de premisas. Agregue un factor de certeza de 80 a la premisa 1.

CAMBIAR 1 FC 80

MOSTRAR 1

Ahora agregaremos una premisa. Típee

DEF CERCA(AMBIENTE,EXTIN,PASTIZAL):FC 95

Luego TERMINAR

- u) Antes de abandonar la unidad imprimiremos el estado actual del modelo. Encienda la impresora. Entre al ambiente de objetos y tipee

IMPRIMIR SISTEMA

Cuando haya terminado, salga con TERMINAR.

Entre al ambiente de implicaciones y tipee

IMPRIMIR TODO

v luego TERMINAR

Repita lo mismo con el ambiente de premisas. Ahora si abandone la unidad.

- v) Coloque el diskette DEDUCA. presione Ctrl-C y tipee DEDUCA. Seleccione el procesamiento (1) y observe la secuencia. Demorará varios minutos. Deben salir en total 4 conclusiones.

- w) Cuando haya entrado en el ambiente de conclusiones, pida la impresión de los resultados.

IMPRIMIR TODO

Si desea, puede analizar el origen de las conclusiones. Luego tipee TERMINAR y salga de la unidad. Retire los diskettes y apague el equipo. Terminò el ejemplo. El listado que Ud. obtuvo debe ser igual al que adjuntamos a continuación.

MODELO DEDUC : GUIA

SISTEMA
 SISTEMAS
 SIMPLES
 PESTES
 INVASORA
 SUCESSION
 COMPLEJO
 NATURALE
 BOSQUES
 ABISAL
 ARTIFICI
 CULTIVOS
 PASTIZAL

MODELO DEDUC : GUIA

IMP 1
SI MAYOR(AMBIENTE.CARGA.SISTEMA) ENTONCES DISMINUY(AMBIENTE.SISTEMA),
NO MENOR(AMBIENTE.CARGA.SISTEMA)

IMP 2
SI MENOR(AMBIENTE.EXTIN,SISTEMA) ENTONCES COLAPSO(AMBIENTE,SISTEMA); FC 90

IMP 3
SI MENOR(AMBIENTE.CARGA.SISTEMA) Y MAYOR(AMBIENTE,EXTIN,SISTEMA) ENTONCES
AUMENTA(AMBIENTE,SISTEMA),NO MAYOR(AMBIENTE,CARGA,SISTEMA)

IMP 4
SI NO AUMENTA(DESARROL,SISTEMA) ENTONCES NO CRECE(VIDA,SISTEMA)

IMP 5
SI AUMENTA(DESARROL,SISTEMA) Y NO COLAPSO(AMBIENTE,SISTEMA) ENTONCES
CRECE(VIDA,SISTEMA)

IMP 6
SI NEGATIVO(DESARROL,SISTEMA) Y AUMENTA(DESARROL,SISTEMA) ENTONCES
DISMINUY(AMBIENTE,SISTEMA)

IMP 7
SI POSITIVO(DESARROL,SISTEMA) Y AUMENTA(DESARROL,SISTEMA) ENTONCES
CRECE(AMBIENTE,SISTEMA)

IMP 8
SI ALTO(EXTIN,SISTEMA) Y MENOR(AMBIENTE.CARGA,SISTEMA) ENTONCES
CERCA(AMBIENTE,EXTIN,SISTEMA)

IMP 9
SI CERCA(AMBIENTE,EXTIN,SISTEMA) Y DISMINUY(AMBIENTE,SISTEMA) ENTONCES
MENOR(AMBIENTE.EXTIN,SISTEMA); FC 70

IMP 10
SI BAJO(EXTIN,SISTEMA) ENTONCES NO CERCA(AMBIENTE,EXTIN,SISTEMA)

MODELO DEDUC : GUIA

PREM 1
ALTO (EXTIN.COMPLEJO) ; FC 80

PREM 2
BAJO (EXTIN.SIMPLES)

PREM 3
AUMENTA (DESARROL.PASTIZAL)

PREM 4
NEGATIVO (DESARROL.PASTIZAL)

PREM 5
CERCA (AMBIENTE,EXTIN,PASTIZAL) ; FC 95

MODELO DEDUC : GUIA

CONCLUSION 6
DISMINUY
AMBIENTE
PASTIZAL

CONCLUSION 7
NO CERCA
AMBIENTE
EXTIN
SIMPLES

CONCLUSION 8
MENOR
AMBIENTE
EXTIN
PASTIZAL

FC = 67

CONCLUSION 9
COLAPSO
AMBIENTE
PASTIZAL

FC = 60

E. LA UNIDAD DEDUC. DEFINICION DEL MODELO (ALMACENAMIENTO Y RECUPERACION)

La unidad DEDUC está compuesta por cuatro ambientes

- 1 Inicialización
- 2 Objetos
- 3 Implicaciones
- 4 Premisas

A continuación se describirán en detalle tanto la sintaxis como la semántica de cada uno de los comandos permitidos en cada ambiente.

1. Inicialización

En el ambiente de inicialización no existen comandos del usuario. La única intervención de éste es para tipear el nombre del modelo que se quiere inicializar. Para ello el sistema escribe en la pantalla a continuación del menú de la unidad el siguiente texto:

```
DEDUC: NUEVO MODELO
```

```
NOMBRE? ■
```

Ud. contesta tipeando el nombre, que se verá en pantalla a continuación del signo de interrogación. Luego presione <RETURN>.

La función de este ambiente es preparar en el diskette colocado en el drive B: las estructuras de un nuevo modelo DEDUC. Estas estructuras consisten de dos pequeños archivos de parámetros, DEDUC.TAB y DEDUC.NOM, y el gran archivo DEDUC.MOD que contiene la base de datos del modelo.

Esos tres archivos ocupan aproximadamente el 80% de la capacidad de un diskette Apple. Por eso debe usarse un diskette en blanco. Si el diskette ya fue usado como modelo o para otra finalidad (pero en CP/M) puede borrarse mediante el comando CP/M

ERA B:*.*

Este comando debe emitirse sólo en el ambiente del sistema operativo, es decir cuando se ve la señal A> seguida por el cursor, antes de tipearse DEDUC.

Si el diskette para el modelo es virgen o fue utilizado con otro sistema operativo, por ejemplo DOS 3.3 de Apple, debe aplicarse el procedimiento explicado en el capítulo C.3.

El archivo DEDUC.TAB es fijo, igual para todos los modelos, y contiene los parámetros que permiten al sistema el acceso a la base de datos. El archivo DEDUC.NOM es muy importante pues contiene el estado de la base de datos. Ese archivo se regraba cada vez que se abandona un ambiente. Por eso es importante salir normalmente del mismo.

Finalizado el proceso de Inicialización el sistema vuelve al menú de la unidad funcional.

2. Objetos

El siguiente cuadro resume el conjunto de comandos mediante los cuales Ud. utiliza las funciones de este ambiente:

a) Comandos

Definir <objeto> : <lista de grupos>

Mostrar	}	TODO/TODA/TODOS/TODAS
Imprimir		<lista de objetos>

ELiminar <objeto>

TERminar

b) Elementos de los comandos

<lista de objetos> => <objeto>
 <objeto> , <lista de objetos>

<lista de grupos> => <grupo>
 <grupo> , <lista de grupos>

<grupo> => <objeto>
 <objeto> (<lista de grupos>)

<objeto> => nombre de un objeto

En primer lugar debemos explicar como se interpreta este cuadro que describe el lenguaje de comandos del ambiente de objetos.

Cada comando empieza con una palabra clave. Cuando se tipea un comando, el sistema toma el primer elemento del mismo y lo compara con las cinco palabras permitidas. En realidad sólo compara las primeras letras, que están en mayúscula en el cuadro y que son obligatorias. El resto de la palabra es ignorado. Por ejemplo, el primer comando podría escribirse como

D , DEF , DEFINA , DEFINIME , DEFINICION , DUDA , etc.

Después de la identificación del comando se tipea el cuerpo del mismo que puede ser muy simple o complejo. El símbolo <> encierra un elemento del comando que se describe en la segunda sección del cuadro. Cuando una palabra o símbolo no está encerrada entre <>, debe tipearse tal cual. Cuando es un texto, describe lo que debe Ud. tipear.

Los elementos del comando se explican en la parte b) del cuadro. Esos elementos pueden tener una o más formas optativas. Algunas de esas formas son simples, otras son complejas. El símbolo => significa "reemplazar por". En algunos casos se describe el elemento con otro u otros elementos entre <> para mayor claridad conceptual. También hay casos donde el elemento a reemplazar figura en la expresión que lo reemplaza, generalmente para señalar listas de elementos separados por coma. De todos modos describiremos a continuación cada comando con mayor detalle aclarando cada elemento del cuadro.

a) Definir

Un objeto se identifica mediante una palabra cuya longitud no debe exceder los 8 caracteres, que pueden ser letras, mayúsculas o minúsculas, dígitos o símbolos (excepto paréntesis, dos puntos, coma y punto y coma). El primer carácter debe ser una letra.

Cada vez que se menciona un objeto, ya sea mediante este comando o durante la definición de implicaciones y premisas, éste es ingresado a la base de datos del modelo. Sucesivas menciones consisten en ubicarlo dentro de la misma.

El ambiente de objetos se encarga más específicamente del manejo de las relaciones entre objetos. El comando Definir permite justamente relacionar dos objetos, en su forma más simple, un objeto con el conjunto de sus inmediatos descendientes en la estructura o varios niveles en una sola expresión.

Una estructura compleja puede expresarse con un sólo comando o con varios, según Ud. prefiera. En el ejemplo desarrollado en el capítulo anterior se utilizaron tres comandos para expresar la estructura SISTEMAS. Se podría haber hecho con uno solo:

```
DEF SISTEMAS:SIMPLES(PESTES,INVASORAS,SUCESION),COMPLEJOS(
NATURALES(BOSQUES,ABISAL),ARTIFICIALES(CULTIVOS,PASTIZALES))
```

También se podría haber subdividido aún más, por ejemplo

```
DEF SISTEMAS:SIMPLES
```

```
DEF SISTEMAS:COMPLEJOS
```

La función principal del comando es relacionar un objeto con uno o más de sus descendientes. Se pueden agregar descendientes con nuevas definiciones. Si Ud. observa el cuadro verá que la definición de la estructura está compuesta por el nombre de un objeto, seguido por dos puntos y una lista de sus descendientes separados por coma.

En el cuadro dice "lista de grupos" pues se puede aprovechar la definición para crear relaciones con los descendientes del objeto principal. Cuando se utiliza esa oportunidad, el conjunto de descendientes del objeto, a su vez descendiente, se expresa entre paréntesis. Ud. puede ver que en la definición de "grupo" se coloca un objeto sólo o un objeto seguido por una nueva "lista de grupos" entre paréntesis. Esto significa que los objetos colocados entre paréntesis pueden estar también acompañados por sus descendientes. Así se puede crear una estructura de varios niveles en un sólo comando. Lo que Ud. debe controlar es que la cantidad de paréntesis derechos iguale a los izquierdos y que estén ubicados correctamente. El primer caso no es crítico pues Ud. recibirá un mensaje de error. En el segundo caso habrá codificado una estructura incorrecta.

Existen algunas restricciones con respecto a la complejidad del comando.

- .El comando no debe contener más de 50 objetos en total.
- .La estructura definida por el comando no debe tener más de 10 niveles.

En ambos casos Ud. recibirá un mensaje de error con el consiguiente rechazo del comando, que deberá ser dividido en comandos más simples.

La cantidad total máxima de objetos en un modelo es de 500 objetos.

Recuerde que si el nombre del objeto tiene más de 8 letras, las que exceden esa cantidad son ignoradas.

b) Mostrar

La función de este comando es exhibir en pantalla estructuras. Ud. puede pedir toda la estructura dependiente de un objeto, en cuyo caso Ud. tipea

Mostrar nombre de objeto

Ese objeto puede ser descendiente de otros que no serán mostrados.

En la pantalla cada objeto ocupará un renglón. El objeto nombrado empezará en la primera columna. Sus descendientes inmediatos en la columna 4, los de éstos en la 7 y así sucesivamente; es decir, en forma indentada.

Por ejemplo, SISTEMAS se verá así:

```
SISTEMAS
  SIMPLES
    PESTES
    INVASORA
    SUCESION
  COMPLEJO
    NATURALE
      BOSQUES
      ABISAL
    ARTIFICI
      CULTIVOS
      PASTIZAL
```

En vez de un sólo objeto Ud. puede dar una lista de objetos, separados por coma. Las estructuras de esos objetos serán exhibidas una después de otra.

Tambièn puede pedir la exhibición de todas las estructuras tipeando alguna de las cuatro formas:

MOSTRAR TODO
MOSTRAR TODA
MOSTRAR TODOS
MOSTRAR TODAS

El sistema buscarà todos los objetos que no sean descendientes de ningùn otro y exhibirà su estructura en la forma descripta arriba. Esa búsqueda es muy lenta en la presente versión de DEDUC, por lo que aconsejamos no usar este comando para exhibiciones en pantalla.

c) Imprimir

Este comando es idèntico al anterior. En vez de ser exhibidas en pantalla, las estructuras se imprimen sobre papel. No olvide prender la impresora previamente, pues en caso contrario se suspenderà la ejecución del sistema y volverà el control del equipo al sistema operativo (luego de un Ctrl-RESET).

Aquí si tiene sentido pedir todas las estructuras, pues aunque demore mucho, Ud. no necesita quedarse frente a la computadora.

d) Eliminar

La eliminación de un objeto no significa que su nombre sea borrado de la base de datos del modelo. Es importante tenerlo en cuenta por la restricción de 500 objetos como máximo absoluto.

El objetivo del comando es separar al objeto de sus descendientes y antecesores. Lo mismo se hace con todos sus descendientes a menos que al mismo tiempo pertenezcan a otra estructura, lo que es perfectamente factible.

e) Terminar

Con este comando Ud. sale del ambiente. Si utilizò el comando IMPRIMIR en algùn momento, se imprimirà aquí el último renglòn pendiente.

3. Implicaciones

El cuadro de comandos es el siguiente:

a) Comandos

SI <expresión> : <lado derecho> <opcionales>

Mostrar { TODO/TODA/TODOS/TODAS

Imprimir { <implicación>

ELiminar <implicación>

TERminar

CAmbiar <implicación> <parámetro> número

b) Elementos de los comandos

<expresión> => <término> O <expresión>
<término>

<término> => <factor> Y <término>
<término>

<factor> => (<expresión>)
<sentencia>
NO <factor>

<lado derecho> => <exp.sentencia>
<exp.sentencia> , <lado derecho>

<exp.sentencia> => NO <sentencia>
<sentencia>

<opcionales> => nada
; <parámetro> número <opcionales>

<sentencia>	=>	<predicado> (<lista de clases>)
<oredicado>	=>	nombre de un oredicado
<lista de clases>	=>	<clase> <clase> , <lista de clases>
<clase>	=>	<objeto> : <objeto>
<objeto>	=>	nombre de un objeto
<implicaciòn>	=>	número de implicaciòn
<parámetro>	=>	FC CI

Seis comandos manejan la definición y recuperación de implicaciones. La base de datos de un modelo tiene espacio para 600 implicaciones. Existe también una restricción sobre la cantidad de sentencias que puede reducir bastante el límite de implicaciones. Estas tienen como mínimo dos sentencias, pero también las premisas y conclusiones están formadas por una sentencia cada una. El sistema DEDUC en su presente versión no admite más de 980 sentencias. También hay limitación para los predicados, que no pueden ser más de 319. Varias sentencias pueden tener el mismo predicado.

Cada implicación ocupa una posición de las 600 disponibles en la base de datos del modelo. El número de posición le permite identificar unívocamente cada implicación. Cada vez que Ud. define una nueva el sistema la ubicará en el primer lugar disponible.

a) Definición: SI

Es el comando más complejo del sistema DEDUC. La cláusula SI - ENTONCES es reemplazada por SI - : . Se colocaron los dos puntos en vez de la palabra ENTONCES por ser más breve. Cuando se exhiba una implicación volverá a aparecer la palabra completa.

La implicación consta de tres partes: una expresión lógica, el lado derecho con las sentencias-conclusión y los parámetros opcionales.

La expresión consta de sentencias como operandos y los operadores lógicos Y, O y NO. Además se pueden utilizar paréntesis para alterar el orden de precedencia normal. Este está reflejado en la forma en que se definió la expresión (vea el cuadro). El orden de precedencia consiste en evaluar primero de izquierda a derecha todas las sentencias que se encuentran en la expresión. Luego se da prioridad a lo que se encuentra entre paréntesis, del nivel interno al externo. Dentro de los paréntesis se sigue el orden de precedencia de los operadores. Primero se evalúan los NO invirtiendo los valores lógicos de las sentencias o expresiones parciales entre paréntesis. De esta manera se habrán evaluado los "factores". Luego se evalúan los Y, siempre de izquierda a derecha, es decir los "términos" y por último los "O" lógicos.

Los siguientes son ejemplos de expresiones:

<sentencia 1> O <sentencia 2> Y <sentencia 3>

(<sentencia 1> O <sentencia 2>) Y <sentencia 3>

En el primer caso se evalúa primero <sentencia 2> Y <sentencia 3>, mientras que en el segundo se evalúa primero <sentencia 1> O <sentencia 2>.

<sentencia>

NO <sentencia>

<sentencia 1> O NO <sentencia 2>

El lado derecho de la implicación consta de un conjunto de sentencias que pueden ser precedidas por el operador NO para señalar que la conclusión es negativa. Las sentencias se separan por comas.

Un ejemplo de implicación completa es

SI MENOR(AMBIENTE.CARGA.SISTEMA) Y MAYOR(AMBIENTE.EXTIN.
SISTEMA) : AUMENTA(AMBIENTE.SISTEMA). NO MAYOR(AMBIENTE.
CARGA.SISTEMA)

Es decir.

SI <sentencia 1> Y <sentencia 2> : <sentencia 3>. NO <sentencia 4>

Las sentencias constan de un predicado y una lista de objetos entre paréntesis. No puede haber más de 8 objetos en una sentencia.

Cuando se mencionan objetos en una implicación, el significado es que la expresión es válida para cada uno de los objetos contenidos en aquel. Si Ud. quiere hacer referencia al conjunto en forma global, es decir si desea evitar deducciones verticales, debe anteponer al nombre del objeto el símbolo : (dos puntos). Esto le dirá al proceso deductivo que ese objeto en esa situación no tiene descendientes.

Los opcionales se refieren al factor de certeza (FC) y al coeficiente de impacto (CI). Si no se colocan en una implicación. DEDUC les asigna automáticamente el valor 100. El orden en que se escriben no importa ya que deben ir acompañados por sus siglas representativas y separadas del cuerpo principal de la implicación y entre ellos por un punto y coma.

Por ejemplo,

SI <sentencia 1> : <sentencia 2> ; FC 90 ; CI 50

SI <sentencia 1> : <sentencia 2> ; CI 70

SI <sentencia 1> : <sentencia 2> ; FC 95

El comando CAMBIAR permite modificar esos valores a posteriori sin reescribir toda la implicación.

Existen restricciones para la definición de una implicación:

- . no debe haber más de 50 referencias a objetos (sean o no repetidos)

- . no debe haber más de 20 referencias a predicados
- . no debe haber más de 20 sentencias en total
- . la expresión no debe contener más de 19 elementos entre operadores y operandos (los paréntesis no cuentan)
- . no debe haber más de 5 sentencias en el lado derecho
- . existe límite a la complejidad de la expresión que puede estimarse en aproximadamente cuatro niveles de paréntesis.

b) Mostrar

Este comando permite exhibir en pantalla todas las implicaciones en forma sucesiva o una sola dando el número que la identifica.

Cuando se usa este comando, los dos puntos son reemplazados por la palabra ENTONCES. Como las implicaciones exceden a veces la longitud de un renglón, el sistema las muestra en renglones sucesivos, pero sin separar sentencias. Estas se muestran siempre en forma completa en un renglón. Lo mismo pasa con las palabras clave (ENTONCES, NO, FC, CI) y los números.

Como la capacidad de la pantalla puede ser fácilmente excedida si se piden ver todas las implicaciones, recuerde que puede utilizar Ctrl-S para congelar la salida.

c) Imprimir

Funciona igual que el comando MOSTRAR pero la salida será por la impresora, que no debe olvidar de encender antes de emitir el comando.

d) Eliminar

Mediante este comando Ud. anula una implicación determinada cuya ubicación en la base de datos del modelo queda inmediatamente disponible para cuando Ud. defina una nueva.

e) TErminar

Para salir del ambiente de implicaciones debe usar este comando. Si utilizó el comando IMPRIMIR en algún momento, se imprimirá el último renglón pendiente.

f) CAmbiar

Este comando le será muy útil para alterar los parámetros de las implicaciones (FC v CI) sin reescribirlas. La estructura es muy sencilla. Ud. debe dar el número de implicación, el nombre del parámetro a modificar y el nuevo valor de ese parámetro.

Por ejemplo,

CAMBIAR 10 FC 90

CAMBIAR 10 CI -75

donde 10 es el número de implicación.

4. Premisas

Las premisas son sentencias a las que se asigna un valor: verdadero o falso, y un factor de certeza. Su almacenamiento en la base de datos del modelo funciona igual que con las implicaciones, es decir con posiciones fijas.

Hay espacio para 319 premisas y/o conclusiones. Ambas comparten las mismas posiciones pues las últimas pueden convertirse en premisas, aunque no en este ambiente. La estructura de premisas y conclusiones es idéntica. La diferencia radica en su utilización en el proceso deductivo.

El cuadro de comandos es el siguiente:

a) Comandos

Definir <Exp.premisa> <opción>

Mostrar	}	TODO/TODA/TODOS/TODAS
		<predicado>
Imprimir		<premisa>

ELiminar	}	<predicado>
		<premisa>

TErminar

CAmbiar <premisa> FC número

b) Elementos de los comandos

<Exp.premisa>	=>	.sentencia>
		NO <sentencia>

<opción>	=>	nada
		! FC número

<predicado> => nombre de un predicado

<premisa> => número de premisa

a) Definir

La definición de una premisa es muy sencilla y no requiere mayores explicaciones. Por ejemplo,

```
DEF ALTO(EXTIN.COMPLEJO)
```

```
DEF NO ALTO(EXTIN.SIMPLE): FC 95
```

b) Mostrar

A diferencia de los ambientes anteriores, este comando tiene tres formas. La primera sirve para exhibir todas las premisas, la segunda muestra todas las premisas que tengan como predicado al mencionado, por ejemplo MOSTRAR ALTO, cualquiera sea el valor de la premisa, y la última que muestra una premisa determinada. Si se hace referencia a un número que corresponde a una conclusión, el comando será rechazado.

c) Imprimir

Funciona igual que el comando MOSTRAR pero la salida será impresa. No debe olvidar encender la impresora antes de usar este comando.

d) Terminar

Este comando se usa para salir del ambiente y volver al menú DEDUC. Si utilizó el comando IMPRIMIR, se imprimirá el último renglón pendiente.

f) Cambiar

Para modificar el factor de certeza de una premisa, puede utilizar este comando, por ejemplo, CAMBIAR 5 FC 85, con lo cual la quinta premisa tendrá un factor del 85%.

F. LA UNIDAD DEDUCA: PROCESAMIENTO Y ANALISIS DE CONCLUSIONES

Esta unidad consta de sólo dos ambientes

1 Proceso deductivo

2 Conclusiones

A continuación se describirá a grandes rasgos el proceso deductivo, que no requiere intervención del operador, y el manejo de las conclusiones, que es muy semejante a los ambientes de la unidad DEDUC.

1. Proceso deductivo

El proceso deductivo se explica en detalle en la bibliografía. Daremos aquí únicamente una idea general que puede ser de utilidad para Ud.

En primer lugar debe recordar el significado de una implicación. Esta representa una relación entre objetos (o clases de objetos), algunos de los cuales son mencionados en varias sentencias. Esa relación es válida para cada uno de los objetos contenidos en las clases. Es fundamental recordar este sentido de la implicación.

Cuando se obtienen conclusiones a partir de las premisas, las implicaciones se aplican sólo a aquellos objetos, para los cuales las premisas tienen un valor asignado. A cada clase de objeto de la implicación le corresponderá uno o más objetos de esa clase, para los cuales se cumple la expresión lógica.

El proceso deductivo consta de varias etapas. En cada una de ellas se emplea un conjunto de sentencias activas (premisas o conclusiones) para la evaluación de las implicaciones. En el primer paso ese conjunto consiste únicamente de premisas (si existen conclusiones de otro procesamiento anterior, éstas no intervendrán). Si se generan conclusiones, se efectúa un nuevo paso de deducción pero utilizando sólo esas conclusiones nuevas. Esto se repite hasta que los sucesivos nuevos conjuntos de conclusiones no generen otras por sí solos. Cuando se llega a esa situación se agregan todas las conclusiones a las premisas y se repite el proceso inicial.

Si el proceso con el conjunto completo no genera más conclusiones se detiene el proceso.

El sistema no vuelve al menú DEDUCA al finalizar, sino entra directamente al ambiente de conclusiones.

En cada paso de deducción el sistema recorre todas las implicaciones y, al final, emite un mensaje acerca de la cantidad de conclusiones obtenidas. También informa acerca de la implicación que está procesando.

El paso de deducción consta de tres etapas:

a) Determinación de las implicaciones relevantes

Se prepara una lista de predicados contenidos en las sentencias que intervendrán en el paso. Se comparan los predicados contenidos en la expresión lógica (lado izquierdo) de la implicación. Si ninguno de éstos se encuentra en la lista, la implicación es descartada. Sólo se procesarán las implicaciones aceptadas para este paso.

b) Evaluación de la expresión lógica de la implicación en base al conjunto de sentencias activas para el paso.

El mecanismo de evaluación determina si y para qué objeto o clases de objetos la expresión es verdadera. El resultado de la evaluación puede ser:

verdadero para los objetos O_1, O_2, \dots, O_n
falso para los objetos O_1, O_2, \dots, O_n
indeterminado (con respecto a sus objetos)

c) Generación y emisión de la conclusión

Sólo si la expresión es verdadera para los objetos O_1, O_2, \dots, O_n , las sentencias del lado derecho se convierten en conclusiones, reducidos sus objetos según O_1, O_2, \dots, O_n .

La conclusión es almacenada en la base de datos del modelo sólo si no existe anteriormente otra de igual valor y cuyos objetos contienen o son iguales a los de la nueva (todos los objetos). Si los objetos de la nueva conclusión contienen o son iguales a la anterior, ésta es eliminada antes de almacenar la nueva. Si hay contradicción entre los valores de la nueva y de la conclusión vieja, se almacenan ambas y se emite un mensaje.

Si existen conclusiones previo a todo el proceso deductivo, éstas pueden impedir la nueva generación de ellas, por lo que no intervendrán en los pasos sucesivos de deducción, no generando nuevas conclusiones. Por lo tanto Ud. debe tener cuidado con la presencia de conclusiones en la base de datos del modelo. En el ambiente de conclusiones Ud. tendrá posibilidad de transformarlas en premisas o eliminarlas luego de su análisis.

Cuando DEDUC procesa una implicación, agota la capacidad evaluativa total del conjunto de sentencias activas, antes de pasar a otra implicación. Es decir, se utilizan todas las combinaciones posibles de esas sentencias en la evaluación de la expresión.

Con respecto al cálculo del factor de certeza, las reglas son las siguientes:

$$FC (P \ Y \ Q) \leq \min (FC(P) , FC(Q))$$

$$FC (P \ O \ Q) \leq \max (FC(P) , FC(Q))$$

$$FC (R) \leq FC(L) \times FC (SI \ L \ ENTONCES \ R) / 100$$

donde P, Q y L son sentencias (reales o intermedias del proceso de evaluación de la expresión lógica) y R es una conclusión.

La negación no altera el coeficiente de certeza.

El impacto sobre una conclusión se calcula como

$$CI (R) \leq CI(L) \times CI (SI \ L \ ENTONCES \ R) / 100$$

donde R y L son sentencias.

2. Conclusiones

Las conclusiones tienen la misma estructura que las premisas. También ocupan las mismas posiciones en la base de datos del modelo. La diferencia radica en que las conclusiones preexistentes no intervienen en el proceso deductivo. Es más, las inhibe para continuar en ese proceso, aún si son obtenidas nuevamente durante el mismo.

En este ambiente Ud. puede convertir las conclusiones en premisas para permitir su participación plena en un proceso deductivo posterior, las puede eliminar, puede averiguar qué implicación y sentencia activa puede haberle dado origen, puede imprimirlas o simplemente exhibirlas en pantalla.

El cuadro de comandos es el siguiente:

a) Comandos

Mostrar	}	TODO/TODA/TODOS/TODAS
		<objeto>
	}	<conclusión>
Imprimir		nada
ELiminar	}	TODO/TODA/TODOS/TODA
		<objeto>
COnvertir	}	<conclusión>
TErminar		
Origen		<conclusión>

b) Elementos de los comandos

<objeto> => nombre de un objeto
 <conclusión> => número de conclusión

a) Mostrar

Este comando opera en tres formas básicas en este ambiente: exhibir todas las conclusiones presentes, exhibir todas aquellas que contengan un determinado objeto y mostrar una conclusión determinada.

La forma en que se exhiben las conclusiones es distinta a la de las premisas. Aquí se muestra el predicado y debajo los objetos. El factor de certeza y los coeficientes de impacto, positivo y negativo, sólo se muestran si son distintos de 100 y de cero respectivamente.

También se puede dar el comando sólo, sin parámetros. Esto significa que los próximos comandos ORIGEN serán exhibidos en pantalla. Si no hubo previamente un comando de impresión, esto no es necesario.

b) Imprimir

Funciona igual que MOSTRAR pero la salida es dirigida a la impresora que debe haber sido encendida y estar ON LINE previamente.

Si se da el comando sin parámetros los próximos pedidos de ORIGEN serán dirigidos a la impresora. El comando con parámetros tiene el mismo efecto, pues activa una llave interna.

c) Eliminar

En las mismas tres formas básicas de MOSTRAR anula las conclusiones dejando libre su espacio para nuevas conclusiones o premisas.

d) Convertir

También con las tres formas básicas transforma las conclusiones en premisas.

e) Terminar

Cuando se da este comando, el sistema vuelve al menú DEDUCA, imprimiendo los renglones pendientes.

f) Origen

Para encontrar el origen inmediato de una conclusión DEDUC recorre las implicaciones comparando las sentencias del lado derecho con la conclusión. Para que la implicación sea aceptada como candidata, y exhibida, la sentencia debe tener el mismo valor de la conclusión y todos sus objetos deben contener o ser iguales a los de ésta.

Luego se analizan las sentencias en el lado izquierdo. Para cada una de ellas, reducidos sus objetos a los de la conclusión, se buscan premisas o conclusiones de igual predicado cuyos objetos sean compatibles (intersección no nula) y se exhiben también.

De esta manera se puede obtener el origen inmediato de una conclusión. Luego se puede seguir retrocediendo hasta que el origen sólo contenga premisas. Es factible que DEDUC proponga varios orígenes de una conclusión. A pesar de que esto es perfectamente posible, no puede descartarse que algunas de esas posibilidades no sea realmente el causante de la conclusión.

La exhibición de esta función en forma impresa o en pantalla puede manejarse por medio de los comandos MOSTRAR o IMPRIMIR. El ambiente posee una llave de dos posiciones, una para imprimir y otra para exhibir en pantalla. Cuando se ingresa al ambiente de conclusiones se encuentra en posición de mostrar. Cada comando de MOSTRAR o IMPRIMIR altera el estado de esa llave interna. Esto sólo tiene algún efecto para el comando ORIGEN, pues es el único que hace uso de esa llave para decidir por donde emitir la salida (MOSTRAR e IMPRIMIR alteran el estado de esa llave antes de ser ejecutados).

Como es muy probable que Ud. desee ver un origen antes de imprimirlo, será muy natural que Ud. no requiera un MOSTRAR o IMPRIMIR específico antes de hacer lo mismo con ORIGEN. Por lo tanto DEDUC le da la posibilidad de alterar el estado de la llave con comandos MOSTRAR e IMPRIMIR en blanco, es decir sin parámetros.

G. LA UNIDAD DEDUCB. TRANSFERENCIAS ENTRE MODELOS DE OBJETOS E IMPLICACIONES

Como muchos modelos pueden tener la misma base cognitiva, se ha previsto un sistema mediante el cual se pueden transferir implicaciones y objetos (estructuras). Lamentablemente esa transferencia requiere una excesiva manipulación de diskettes debido a la reducida capacidad de la Apple y sus unidades de diskette. Además CP/M no permite el cambio de diskette en medio del proceso, por lo que no se puede utilizar el drive B: pues allí se colocarán sucesivamente los modelos origen y destino de las transferencias.

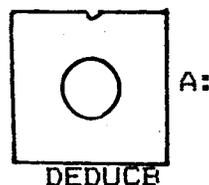
Se creó entonces la unidad funcional DEDUCB que contiene dos ambientes de transferencia y un archivo auxiliar, DEDUC.AUX.

Los dos ambientes sirven para transferir hacia y desde el archivo auxiliar desde y hacia los modelos origen y destino respectivamente.

Pero este archivo auxiliar, que reside en el mismo diskette que los programas, tiene una capacidad limitada. Por tal motivo, es posible que Ud. deba efectuar la transferencia en varias etapas.

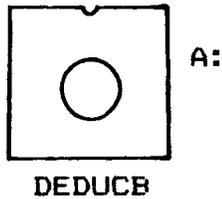
El procedimiento a seguir es el siguiente:

- a) Coloque en el drive A: DEDUCB y en el drive B: el modelo origen. Presione Ctrl-C.



- b) Tipee ERA DEDUC.AUX para borrar la versión anterior del archivo auxiliar.
- c) Tipee DEDUCB y seleccione 1.
- d) Transfiera objetos e implicaciones hasta que reciba el aviso de "capacidad colmada" o haya completado la tarea. Salga del ambiente.

- e) Coloque en el drive B: el modelo destino. Presione Ctrl-C.



- f) Tipee DEDUCB y seleccione 2.
- g) Si aún quedan elementos del modelo a transferir, coloque en drive B: el modelo origen, presione CTRL-C y vuelva a b). En caso contrario proceda según Ud. lo requiera.

Si Ud. desea transferir esos elementos a varios modelos, puede repetir en cada etapa los pasos e) y f) para cada uno de ellos. También podría almacenar un determinado archivo auxiliar en otro diskette. Para ello lo más práctico es hacer una copia total del diskette con el procedimiento del capítulo C.4, es decir igual que con los modelos.

A continuación describiremos los dos ambientes de DEDUCB.

1. Modelo origen a archivo auxiliar

Este ambiente permite transferir estructuras de objetos e implicaciones al archivo auxiliar. También puede Ud. ver esos elementos previamente para confirmar si ha elegido correctamente lo que pensaba transferir.

Cuando el archivo auxiliar se llena, el último comando no es ejecutado y se mostrará un mensaje al respecto.

El cuadro de comandos es el siguiente:

Mostrar	}	nombre de un objeto
		número de una implicación
EXtraer	}	

TERminar

a) Mostrar

Este comando muestra implicaciones y estructuras de objetos de la misma manera que en sus respectivos ambientes.

b) EXtraer

Dando el nombre del objeto, cuya estructura se desea transferir, o el número de implicación, recupera el elemento designado y lo almacena en el archivo auxiliar si tiene espacio. En caso contrario no ejecuta el comando emitiendo un mensaje aclaratorio.

c) TERminar

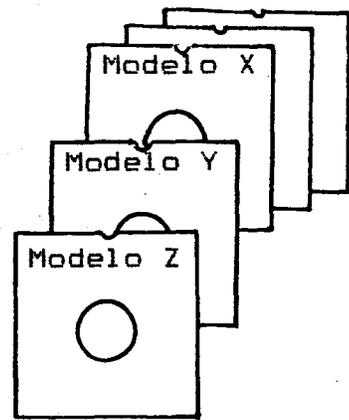
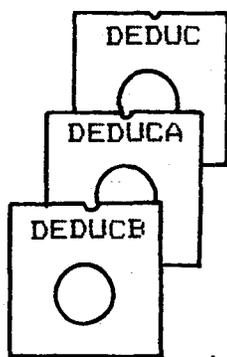
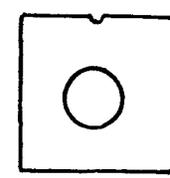
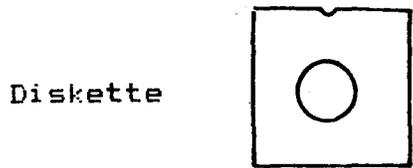
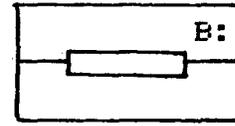
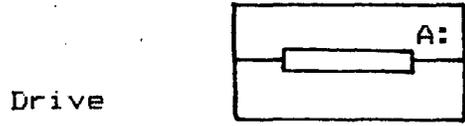
Mediante este comando Ud. abandona el ambiente. El control vuelve al CP/M (A>) para que Ud. pueda reemplazar el diskette modelo en B:.

2. Archivo auxiliar a modelo destino

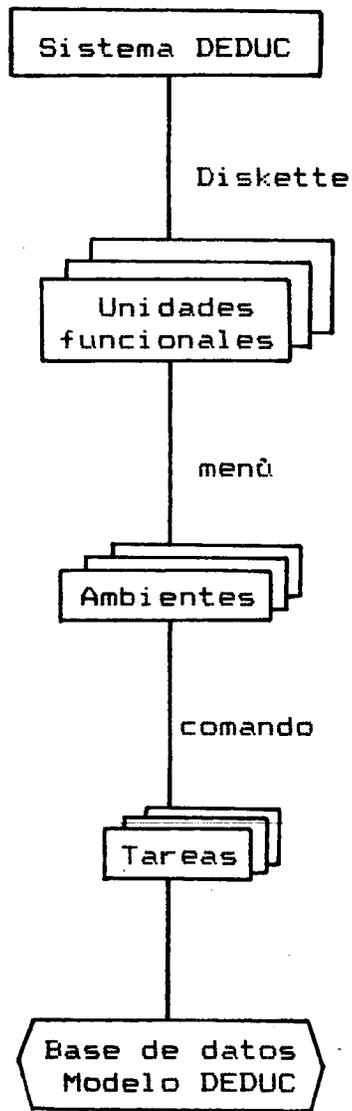
Este ambiente es totalmente automático; no requiere su intervención. Vuelca todos los objetos e implicaciones del archivo auxiliar al modelo destino. Cuando termina devuelve el control al CP/M (A>).

ANEXO. CUADROS DE CONSULTA RAPIDA

a) Diskettes



b) Acceso



c) Unidades funcionales

DEDUC = Almacenamiento v recuperación de un modelo

- 1 Inicialización de un nuevo modelo
- 2 Objetos
- 3 Implicaciones
- 4 Premisas

DEDUCA = Procesamiento de un nuevo modelo

- 1 Proceso deductivo
- 2 Conclusiones

DEDUCB = Transferencias entre modelos de objetos e implicaciones

- 1 Modelo origen a archivo auxiliar
- 2 Archivo auxiliar a modelo destino

d) Comandos

Ambiente DEDUC - Objetos

Definir <objeto> : <lista de grupos>

Mostrar { TODO/TODA/TODOS/TODAS

Imprimir { <lista de objetos>

ELiminar <objeto>

TErminar

<lista de objetos> => <objeto>
 <objeto> , <lista de objetos>

<lista de grupos> => <grupo>
 <grupo> , <lista de grupos>

<grupo> => <objeto>
 <objeto> (<lista de grupos>)

<objeto> => nombre de un objeto

Ambiente DEDUC - Implicaciones

SI <expresión> : <lado derecho> <opcionales>

Mostrar	}	TODO/TODA/TODOS/TODAS
Imprimir		<implicación>

ELiminar <implicación>

TERminar

CAmbiar <implicación> <parámetro> número

<expresión> => <término> 0 <expresión>
<término>

<término> => <factor> Y <término>
<término>

<factor> => (<expresión>)
<sentencia>
NO <factor>

<lado derecho> => <exp.sentencia>
<exp.sentencia> , <lado derecho>

<exp.sentencia> => NO <sentencia>
<sentencia>

<opcionales> => nada
; <parámetro> número <opcionales>

<sentencia> => <predicado> (<lista de clases>)

<predicado> => nombre de un predicado

<lista de clases> => <clase>
<clase> , <lista de clases>

<clase>	=>	<objeto> : <objeto>
<objeto>	=>	nombre de un objeto
<implicación>	=>	número de implicación
<parámetro>	=>	FC CI

Ambiente DEDUC - Premisas

Definir <Exp.premisa> <opción>

Mostrar {
 TODO/TODA/TODOS/TODAS

<predicado>

Imprimir {
 <premisa>

ELiminar {
 <predicado>

<premisa>

TErminar

CAmbiar <premisa> FC número

<Exp.premisa> => <sentencia>
 NO <sentencia>

<opción> => nada
 ; FC número

<predicado> => nombre de un predicado

<premisa> => número de premisa

Ambiente DEDUCA - Conclusiones

Mostrar	}	TODO/TODA/TODOS/TODAS
		<objeto>
		<conclusi3n>
Imprimir	}	nada

ELiminar	}	TODO/TODA/TODOS/TODA
		<objeto>
COvertir	}	<conclusi3n>

TERminar

Origen <conclusi3n>

<objeto> => nombre de un objeto

<conclusi3n> => n3mero de conclusi3n

Ambiente DEDUCB - Transferencia modelo origen a archivo auxiliar

Mostrar

nombre de un objeto

número de una implicación

EXtraer

TErminar

