

CURSO DE ENTRENAMIENTO PARA BIBLIOTECARIAS

26 al 30 Octubre de 1981

INTRODUCCION A LA CATALOGACION AUTOMATIZADA

P R O G R A M A

- 1<sup>er</sup> día : ¿Qué son los computadores?  
¿Qué son los micros?  
(Hugo Cisternas y Rodrigo Cifuentes)
- 2<sup>a</sup> día : ¿Qué es un programa?  
D.B.M.S.  
(Hugo Cisternas y Rodrigo Cifuentes)
- 3<sup>er</sup> día : ¿Qué es un formato?  
Formato MARC  
(María Luisa Arenas)
- 4<sup>a</sup> día : Formato MARC UC  
(María Luisa Arenas)
- 5<sup>a</sup> día : Programas disponibles  
(mañana)  
Proceso de catalogación automatizada y  
Funciones de cada programa  
(Ana María Torres, Alejandro Burgos, Rodrigo  
Cifuentes)
- (tarde) : Mesa redonda  
Demostración

## QUE ES UN COMPUTADOR

=====

Antes de definir lo que es un computador, es necesario notar que hay diferentes tipos de computadores. Entre ellos podemos distinguir dos grandes grupos:

- Computadores compuestos por dispositivos fisicos.
- Computadores abstractos.

En el primer grupo estan:

- Computadores digitales
- Computadores analogicos
- Computadores mecanicos
- Computadores opticos, etc.

En el segundo grupo estan:

- Computadores teoricos
- Computadores virtuales y simulados.

Sin embargo, el gran porcentaje de los computadores lo componen los computadores **digitales**, y es a ellos a los que nos referiremos a continuacion.

Se ha definido un computador digital de muchas formas, por ejemplo:

Ej. 1.- Es un dispositivo electro-mecanico que es capaz de aceptar, almacenar, y manipular informacion, que incluye tanto a los datos como a los programas. La informacion es manejada en la forma de digitos binarios ( 0 y 1 ) que es representada por niveles de voltaje duales.

Ej. 2.- Es un dispositivo capaz de aceptar informacion, comparar, sumar, restar, multiplicar, dividir e integrar esta informacion, que esta en la forma de digitos binarios ( 0 y 1 ), y luego entregar los resultados de este proceso en alguna forma.

Es evidente que todas las definiciones son incompletas, sin embargo, lo importante es enfatizar los aspectos relevantes de un computador digital. Por esta razón, en lugar de entregar una definición definitiva, estudiaremos un computador digital desde tres puntos de vista, respondiendo a tres preguntas fundamentales:

- Como es un computador digital?

- Para que sirve?

- Como funciona?

## COMO ES UN COMPUTADOR DIGITAL ?

=====

Lo primero que es necesario establecer, es que un computador digital es una **maquina** o dispositivo electro-mecanico. Es decir, esta compuesto por partes electronicas y partes mecanicas.

A modo de ejemplo, podemos decir que una gran cantidad de artefactos que usamos a diario son dispositivos electro-mecanicos: Las maquinas de escribir electricas, las lavadoras de ropa, etc. Por otro lado, una maquina de escribir manual es un tipico dispositivo mecanico.

En general, un computador digital se puede dividir esquematicamente en tres grupos de partes:

- Una unidad central
- Un conjunto de perifericos
- Y un conjunto de instrucciones y reglas que controlan su funcionamiento, llamado **SOFTWARE**

### LA UNIDAD CENTRAL

La unidad central de un computador es aquel conjunto de partes que controlan el funcionamiento de la maquina, y realizan los procesos de calculo y manipulacion de la informacion.

La unidad central se compone, en general, de:

- Unidad central de proceso ( C.P.U. )
- Memoria interna
- Control de perifericos

## LOS PERIFERICOS

Los perifericos son el conjunto de dispositivos que permiten la entrada, almacenamiento y salida de informacion. Son aquellos aparatos que nos permiten comunicarnos con el computador y almacenar grandes cantidades de informacion.

Entre los diferentes perifericos tenemos:

- Terminales ( pantallas de video, teletipos, etc. )
- Impresoras
- Lectoras y dispositivos de captura de datos
- Dispositivos de almacenamiento de informacion:

- unidades de diskette
- unidades de disco
- unidades de cinta

## SOFTWARE

El software lo constituye el conjunto de instrucciones y reglas que especifican las operaciones que debe realizar.

Una parte importante del software es el llamado **SISTEMA OPERATIVO** que controla el funcionamiento del computador, y los **PROGRAMAS** que permiten realizar operaciones especificas. Esta parte se veran mayor detalle mas adelante.

## CARACTERISTICAS DEL LOS COMPUTADORES DIGITALES

Entre las características mas relevantes de un computador digital, podemos mencionar:

- **Rapidez.** Un computador puede realizar mas de un millon de operaciones por segundo ( ej. sumas )
  
- **Precision.** Un computador puede realizar calculos con gran precision.
  
- **Capacidad de almacenamiento.** Un computador, a travez de los dispositivos de almacenamiento de informacion, por ej. en un disco, puede contener entre cien y mil millones de caracteres, es decir, 50.000 a 500.000 hojas escritas en tamaño carta.
  
- **Capacidad de calculo.**

## PARA QUE SIRVE UN COMPUTADOR DIGITAL ?

=====

En los ultimos años, los computadores se han usado en las mas variadas aplicaciones, sin embargo, en general podemos clasificar sus usos en dos grandes rubros:

- Procesar informacion
  
- Realizar calculos

Para procesar informacion, el computador nos permite el almacenamiento, recuperacion y modificacion de la informacion, y ademas sintetizar la informacion para obtener resúmenes.

Para realizar calculos, el computador nos permite utilizar operaciones aritmeticas ( sumas, multiplicaciones, etc. ) simples y complejas, y operaciones logicas ( comparaciones, conjunciones, disyunciones, etc. ).

## COMO FUNCIONA ?

=====

Basicamente, el computador ejecuta una secuencia precisa de instrucciones llamados PROGRAMA, que indican sin ambigüedades lo que debe hacer.

Por otro lado, el funcionamiento del computador puede visualizarse como un esquema de administración centralizada, donde la unidad central controla el funcionamiento de los periféricos.

### UNIDAD CENTRAL

La unidad central esta compuesta por:

- La unidad central de proceso ( CPU ), que cumple o ejecuta las instrucciones del programa y del sistema operativo.
- La memoria interna, que es donde residen el sistema operativo y los programas.
- La unidad de control de periféricos, que permite a la unidad central de proceso comunicarse con los distintos periféricos.

## PERIFERICOS

Los distintos tipos de perifericos tienen funciones características:

- Terminales

Permiten entregar las instrucciones al computador.  
Sirven para la entrada de datos.

- Dispositivos de almacenamiento

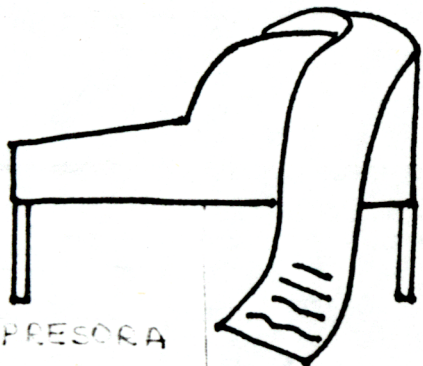
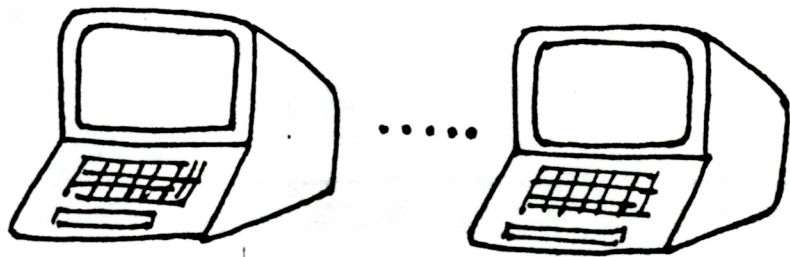
Permiten guardar los datos.

- Impresoras

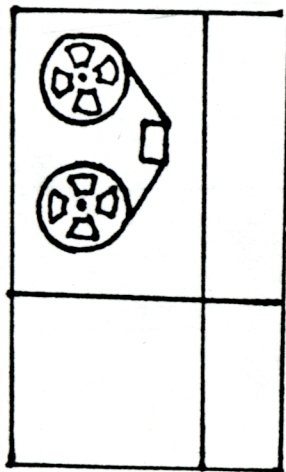
Permiten generar listados.

- Lectoras o dispositivos de captura de datos

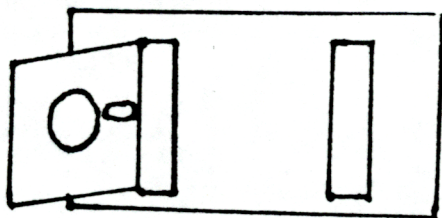
Permiten la entrada de datos codificados en formas específicas: Tarjetas perforadas, códigos de barras, etc.



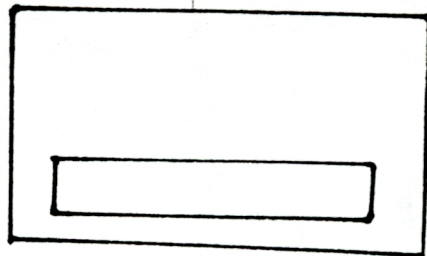
IMPRESORA



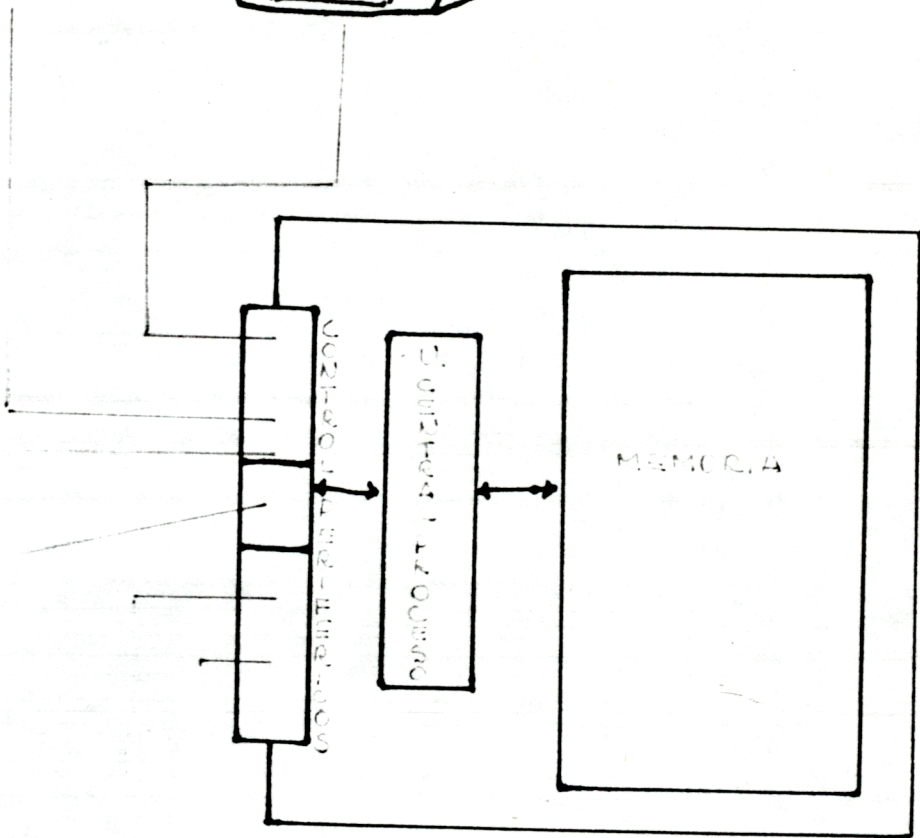
TAPE DRIVE



SCANNER



PRINTER



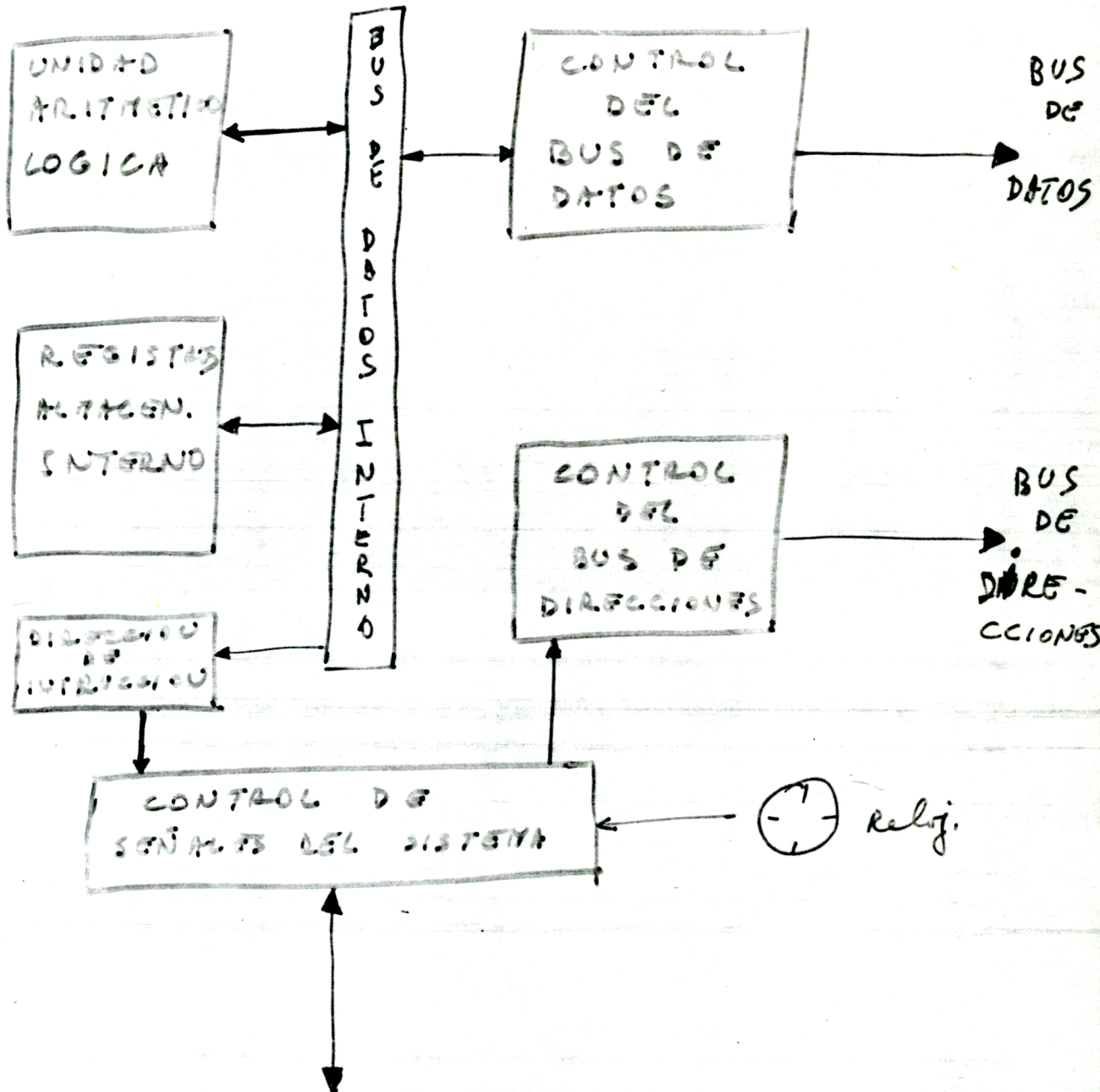
## UNIDAD CENTRAL DE PROCESO ( C.P.U. )

La unidad central de proceso, llamada C.P.U. ( Central Processing Unit ), es la parte medular de un computador, puesto que controla el funcionamiento de este.

La C.P.U. se compone de las siguientes partes fundamentales:

- Unidad Aritmetico Logica, que se encarga de realizar las operaciones aritmeticas , funciones logicas, etc.
- Registros, que almacenan transitoriamente unidades o elementos de informacion para ser procesados por la C.P.U.
- Control de señales del sistema, que envia y recibe señales del resto del sistema para cotrolar su funcionamiento ( Leer, Escribir, Esperar, Sincronizar, etc. )
- Bus de datos, que es el camino que recorren los datos entre la memoria o perifericos y la unidad central de proceso ( C.P.U. )
- Bus de direcciones, que es el medio que utiliza la C.P.U. para indicar a otras partes del sisistema el lugar donde tienen que escribirse o del que debe obtenerse un dato.
- Reloj, que es un generador de pulsos discretos en el tiempo y es el factor de sicronizacion de todo el sistema.

# UNIDAD CENTRAL DE PROCESOS (CPU)



- SEÑALES:
- LEER
  - ESCRIBIR
  - ESPERA
  - SINCRONIZACION

## PERIFERICOS

La función de los periféricos en un computador digital es la de servir a la entrada, salida y almacenamiento de información. Para ello, los distintos tipos de periféricos permiten funciones específicas:

- Terminales

Permiten la comunicación del ser humano con el computador, ya sea para especificar las instrucciones o programas que debe ejecutar, o para el ingreso de datos.

- Unidades de almacenamiento

Permiten "almacenar" gran cantidad de información en forma recuperable por el computador.

- Impresoras

Estos dispositivos permiten generar listados.

- Lectoras o dispositivos de captura de información

Permiten ingresar datos al computador desde elementos externos. La información contenida en estos elementos (tarjetas, papel, etc.) está en formatos que, si bien no son directamente recuperables como es el caso de las unidades de almacenamiento de información, se pueden "leer" utilizando estos dispositivos: Lectoras de tarjetas perforadas, lectoras de barras, etc.

## UNIDADES DE DISCO

Una de las partes importantes de un computador son las unidades de disco, ya que en ellas se puede almacenar una gran cantidad de informacion, y la recuperacion es de alta velocidad.

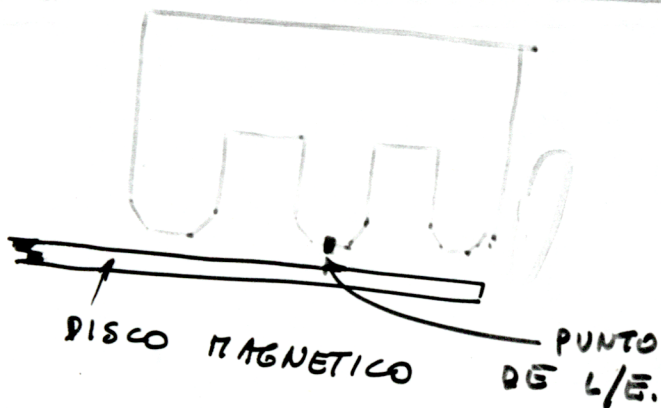
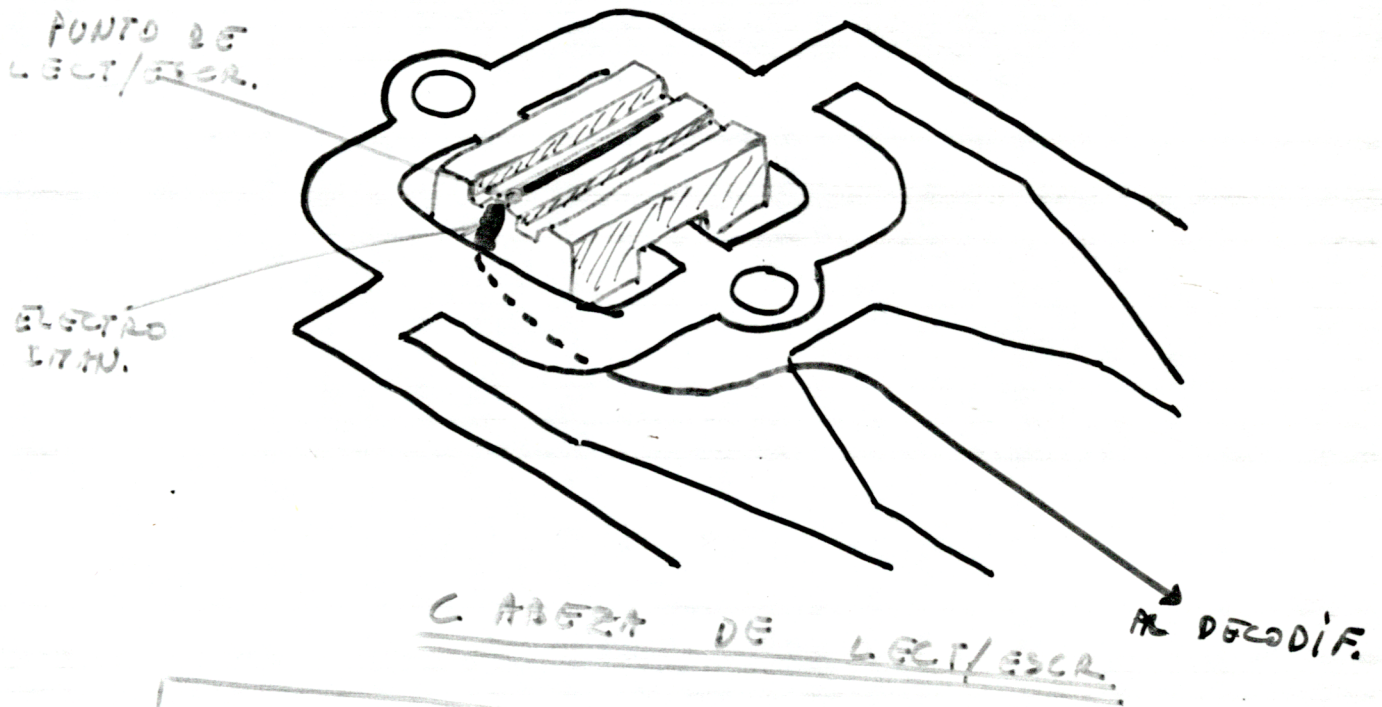
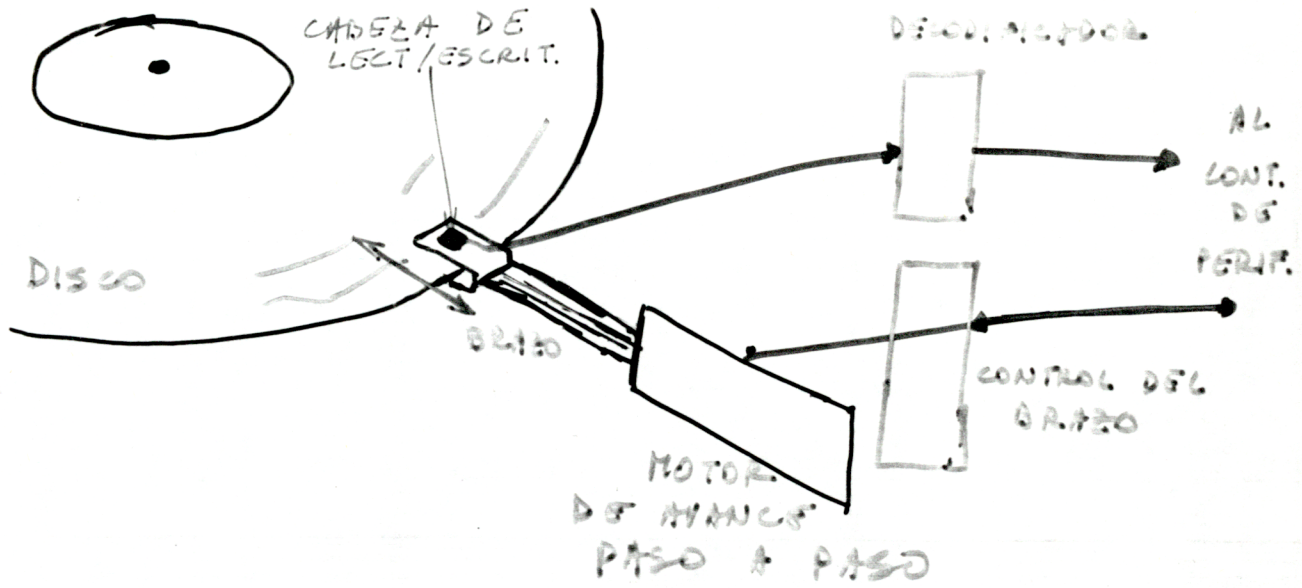
Las unidades de disco magnetico constituyen la llamada memoria secundaria del computador y consiste de uno o mas discos metalicos ( platos ) montados en un eje comun y que se mantienen girando a gran velocidad ( Ej.: 3.600 revoluciones por minuto ).

Estos discos metalicos estan recubiertos por una delgada pelicula de material magnetico.

Una cabeza lecto-grabadora se situa en el extremo de un brazo que la une con un motor de "paso a paso", que le permite moverse en sentido radial con respecto al disco.

La cabeza de lectura/escritura graba la informacion en pistas circulares y concentricas en el plato. El motor de paso a paso le permite acceder a cualquiera de estas pistas.

# DISCOS MAGNETICOS



## QUE ES UN MICROCOMPUTADOR?

Esta pregunta se puede contestar desde dos puntos de vista:

-Para una persona que quiere adquirir un computador, el micro es una maquina para uso personal, que maneja pocos recursos y tiene un precio bajo.

-Para un tecnico, es un computador construido con una tecnologia especial, que permite tener una Unidad de Proceso completa en un solo CHIP (MICROPROCESADOR), que por el momento permite hacer maquinas sencillas, que manejan pocos recursos, y a un costo muy por debajo del de las otras tecnologias usadas hasta hoy.

Los componentes de un Micro normalmente son los perifericos (una pantalla, un par de unidades de Diskette y una impresora) y un grupo de tarjetas en las cuales residen las funciones basicas del computador (Ver figura 1):

-Una tarjeta de Memoria, contiene hasta 64Kb.

-Una tarjeta para control de la pantalla y la impresora.

-Una tarjeta para control de las unidades de Diskette.

-Una tarjeta que contiene el MICROPROCESADOR y otros circuitos que facilitan su comunicacion con el resto del sistema.

Decir que un micro es SENCILLO y que maneja POCOS recursos es una inexactitud. Cualquier micro de hoy es mucho mas poderoso que cualquier Gigante de los años 45 al 60, o mas.

Ademas, como se vio antes, el crecimiento de la capacidad de los computadores es permanente, y, aunque la configuracion basica de un micro sigue siendo la misma que vimos recien, la tendencia es a ocupar estas maquinas, cada vez mas, en aplicaciones comerciales pequenas, industriales y cientificas, que salen del ambito del uso personal, lo cual, junto a los avances de la tecnologia, ha impulsado a los fabricantes a construir maquinas con mas posibilidades y mas flexibles.

Ejemplo de esto es que ya se puede conectar a los micros algunos perifericos que antes se veian solo en los Grandes Computadores, como Unidades de Cinta y Discos de alta capacidad (100 veces la de un Diskette), Impresoras de alta velocidad (1200 lineas por minuto), etc. (Ver

# COMPONENTES DE UN MICROCOMPUTADOR

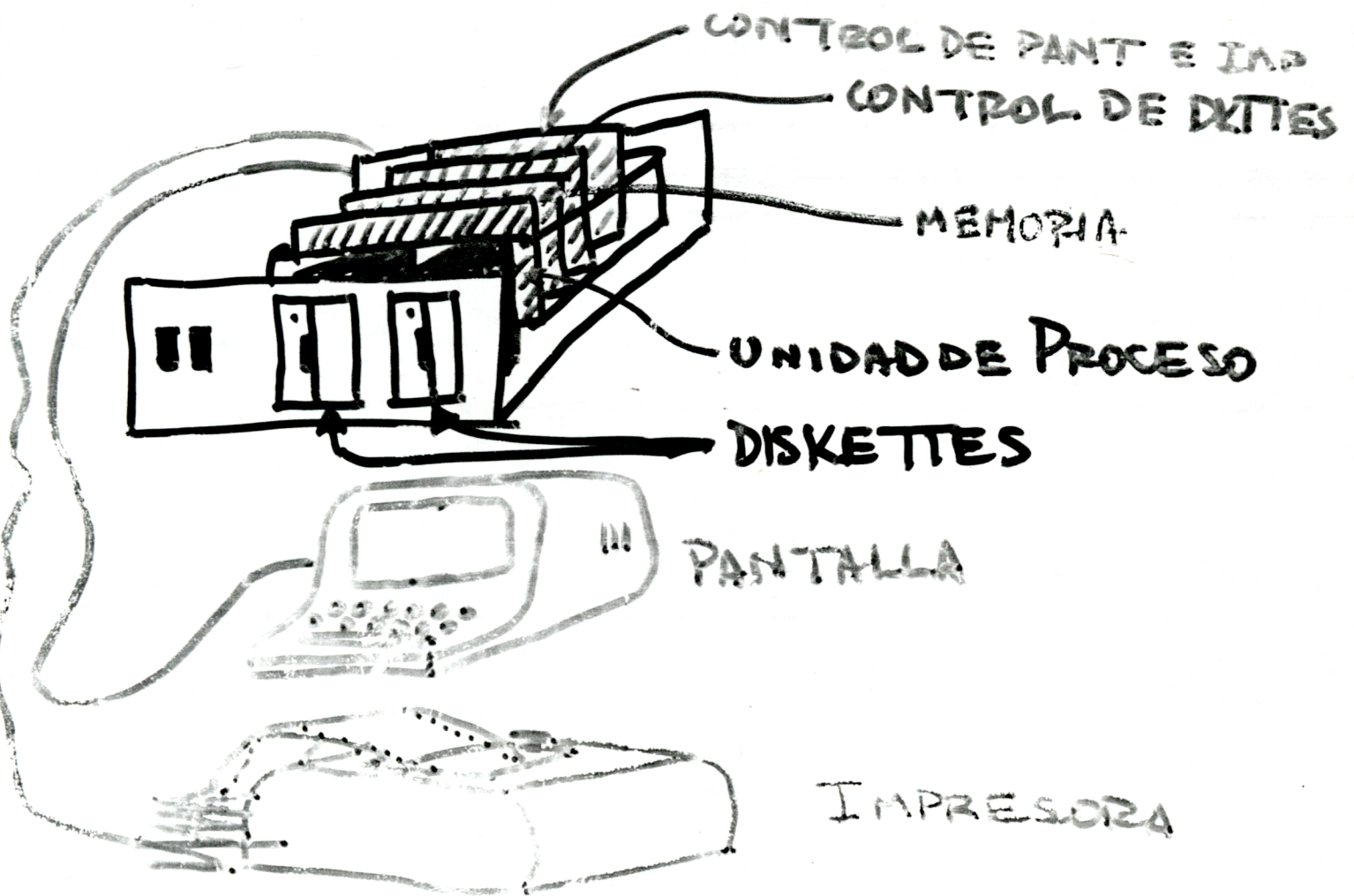


FIG 1

figura 2).

Ademas , hoy se puede comprar por US\$1.000 una tarjeta que contiene un Microprocesador, 64kb de memoria, control de I/O para una pantalla y una impresora, y capacidad de interconectarse a otras tarjetas similares, con lo cual se puede formar una RED que actua bajo el control de una de ellas, que hace las veces de Directora.

Cada una de estas tarjetas es en si un micro independiente, al cual solo le falta acceso a disco u otro medio de almacenamiento (Ver figura 3).

La Tarjeta Directora, llamada MASTER maneja un set completo de tarjetas de control de I/O, con lo cual es capaz de acceder discos, cintas, otras impresoras, etc. El Master es capaz de 'prestar' sus perifericos a los demas micros de la red (SLAVES), en la medida que ellos lo necesitan y si dos o mas slaves requieren un mismo recurso simultaneamente, el se encarga de resolver el problema poniendo las solicitudes en cola , con prioridades, etc. (Ver figura 4).

Todo esto lleva a preguntarse si los micros no llegaran a reemplazar a los grandes computadores. La verdad es que nadie en este momento lo sabe con certeza.

Los Grandes manejan una cantidad enorme de recursos (Ver figura 5) para lo cual necesitan gran cantidad de circuitos especializados y muy complejos, que con la tecnologia de hoy no pueden ser reducidos a un solo Chip. En particular , la Unidad de Proceso de un computador Grande ocupa tipicamente entre 4 y 5 tarjetas de circuitos, cada uno de ellos tan complejo como el Microprocesador mas poderoso que existe hoy (ver figura 6).

Sin embargo , los fabricantes de microprocesadores hacen Chips cada vez mas completos , lo que hace pensar que en un plazo no muy grande todo lo que ACTUALMENTE hace un computador grande, va a caber en un solo MICROPROCESADOR.

El problema es que nadie sabe de que van a ser capaces los Grandes en ESE momento.

3

- UN GRAN AVANCE PARA LOS  
MICROS, ES QUE HOY PUEDEN  
TRABAJAR CON PERIFERICOS  
DE GRAN CAPACIDAD:

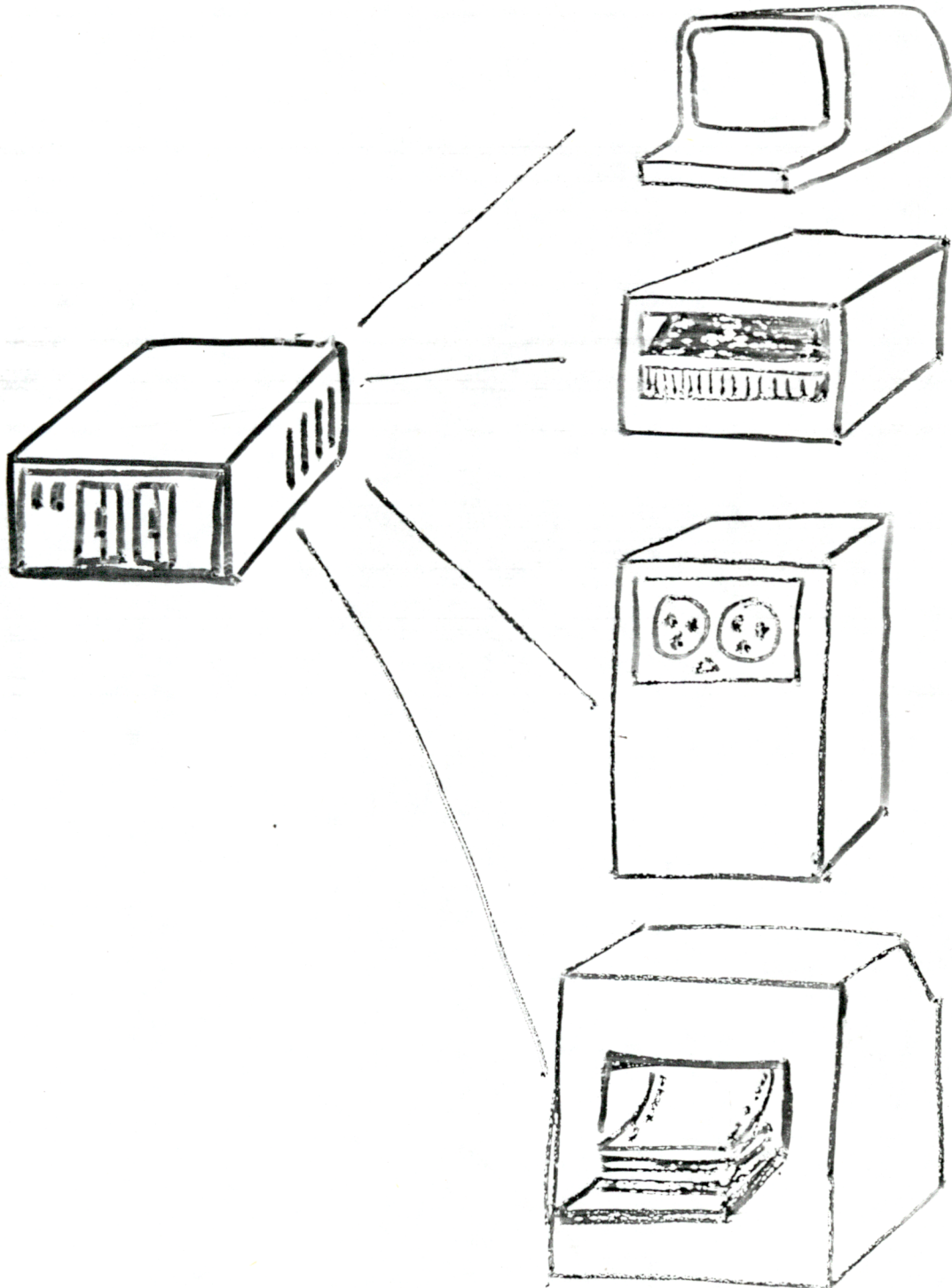


FIG. 2

9

- OTRO AVANCE ES PONER TODOS  
LOS COMPONENTES BASICOS  
EN UNA TARJETA

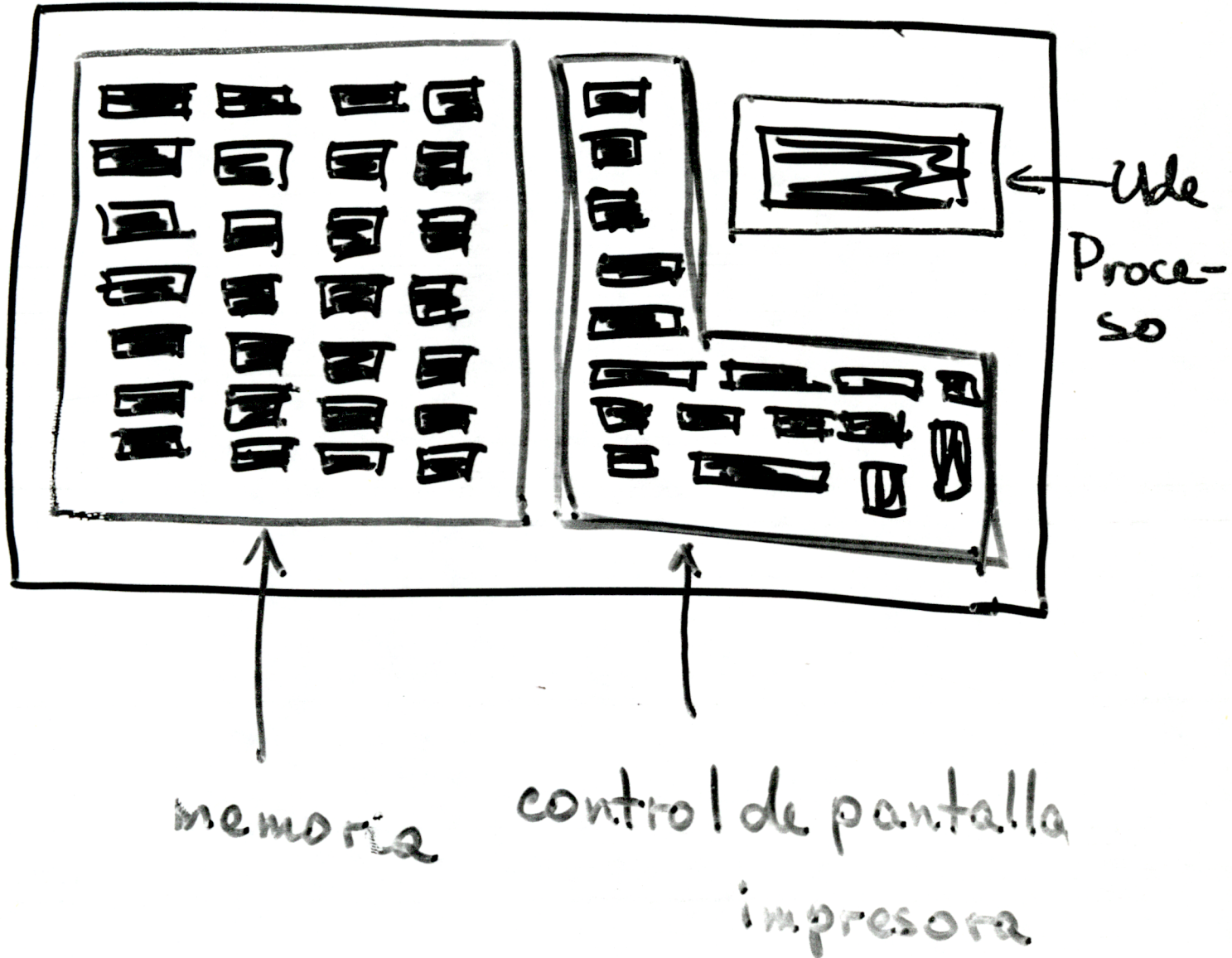


FIG 3

# "MICRO" PARA VARIOS USUARIOS

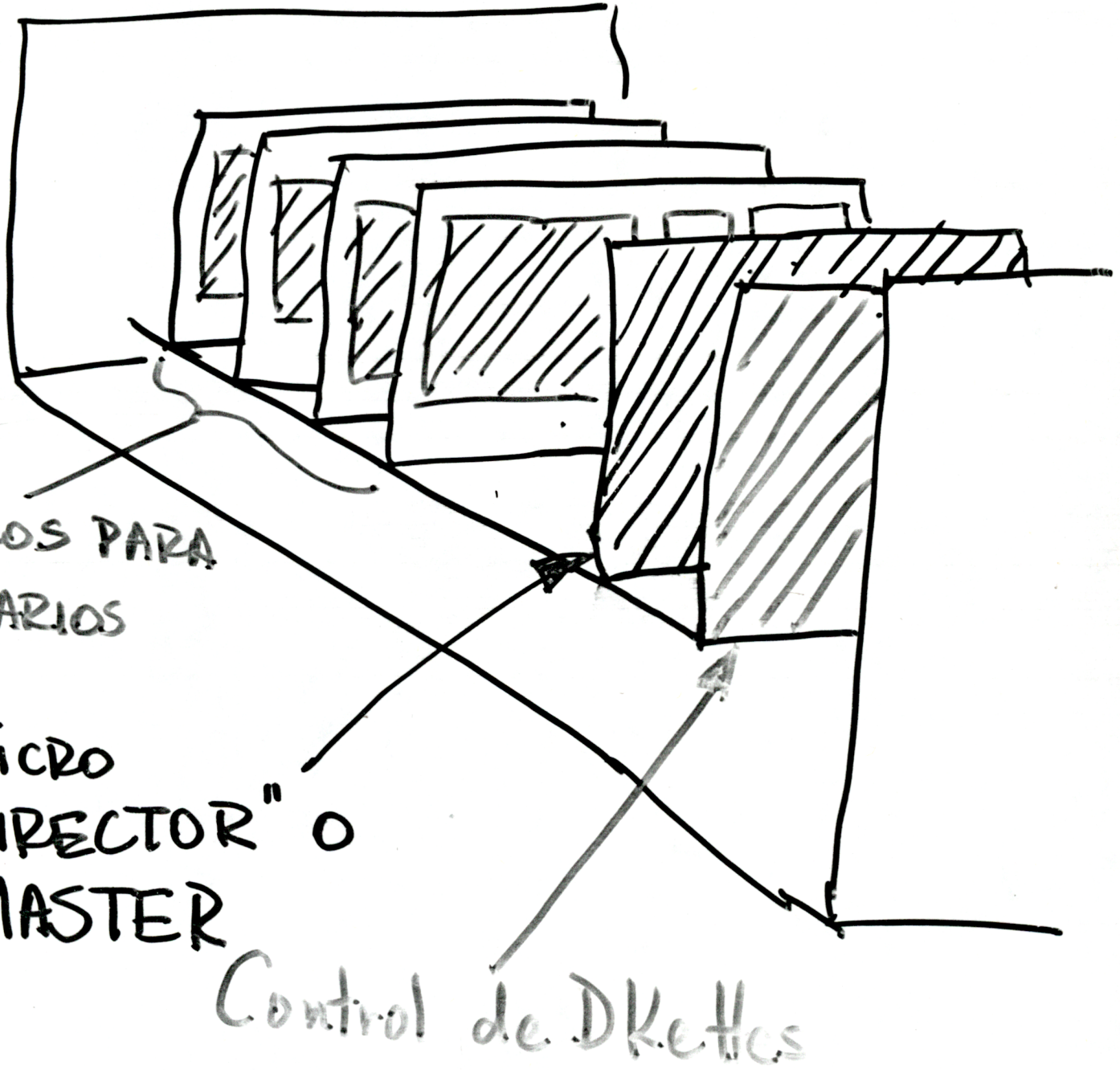
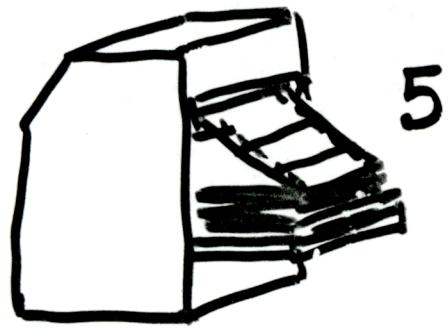
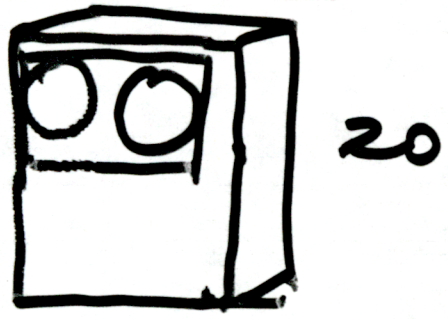
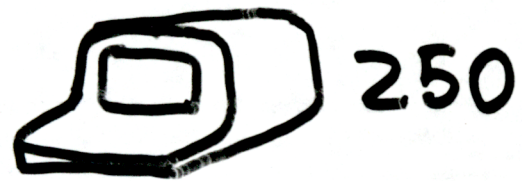


FIG 4

- GRANDES



MEMORIA

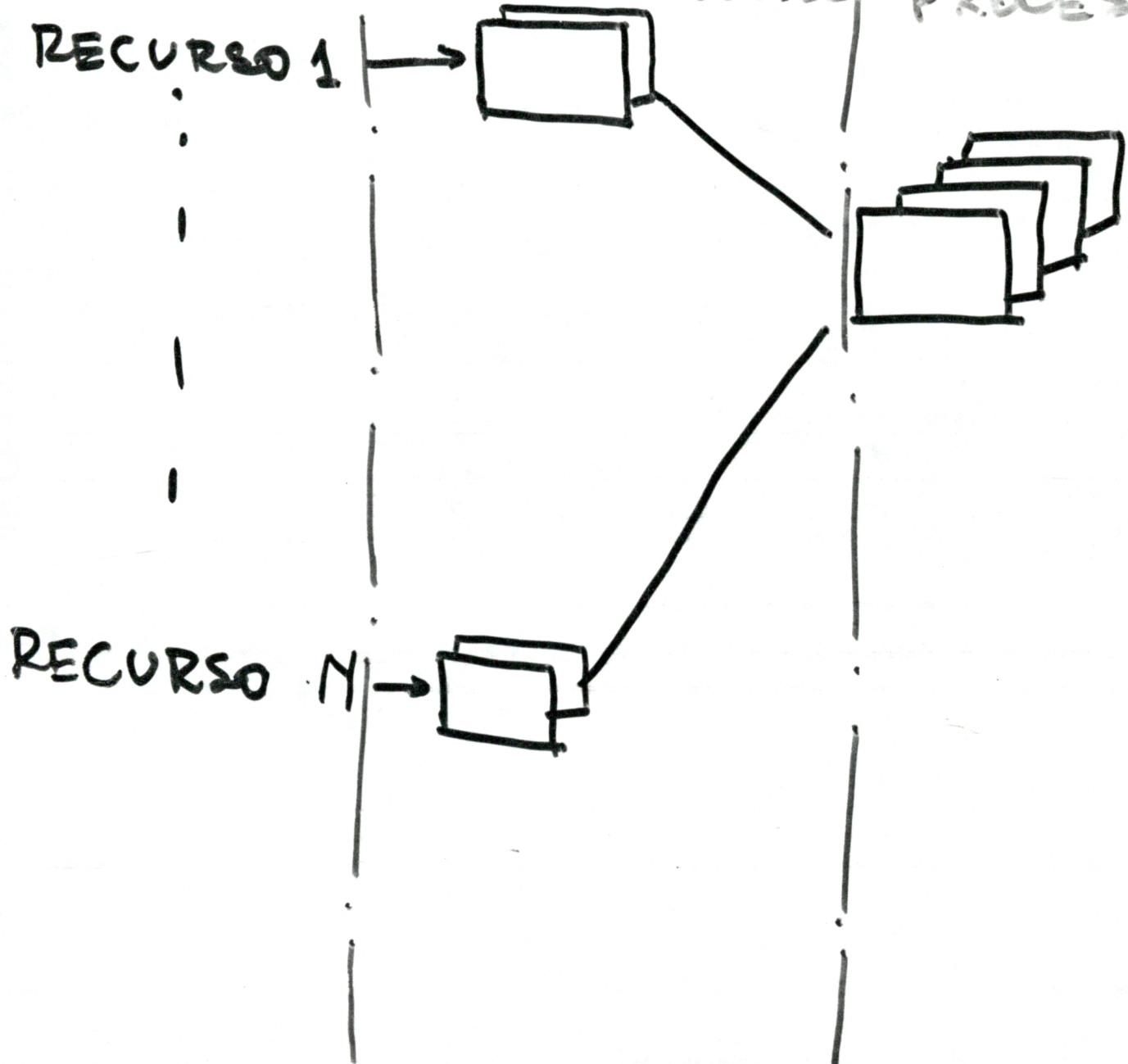
4 M by



FIG 5

- CIRCUITOS MUY ESPECIALIZADOS

RECURSOS!	TARJETAS DE MICROCIRCUITOS	UNIDAD DE PROCESO
-----------	----------------------------	-------------------



- U de PROCESO MUY COMPLEJA, REQUIERE DE MUCHOS MICROCI- CUITOS

FIG 6

## QUE ES UN PROGRAMA

En el lenguaje cotidiano es muy comun escuchar la palabra **programa**, y su uso pocas veces esta relacionado con actividades computacionales. Normalmente el significado en cada ocasion esta determinado por el contexto en que la palabra se inserta y no por una definicion precisa y unica. Por ejemplo:

- Programa del V Congreso de Cardiologia de Chile
- Programa del Seminario Sobre Uso de los Medios de Comunicacion.
- Programa de Actividades del Centro de Estudios Internacionales para el año 1981
- etc.

Puede decirse, en todo caso, que los distintos usos de la palabra hacen referencia a una **Especificacion de Acciones** de caracter mas o menos obligatorio. En el caso del Congreso de Cardiologia, por ejemplo, el **programa** debe ser cumplido en forma mas o menos estricta por quienes estan a cargo de la organizacion del evento.

Los **programas** vistos tienen, ademas, una característica comun: todos ellos se "ejecutan" una sola vez. Por ejemplo, si despues del V Congreso de Cardiologia se hace otro, ya no sera el V sino el VI, y tendra otras actividades distintas de las del anterior.

Hay **Especificaciones de Acciones** que todos nosotros usamos diariamente, a las cuales no llamamos **programas** pero que estan mucho mas cerca de lo que son los programas computacionales que los ejemplos anteriores:

- Sumar dos numeros enteros de varias cifras
- Catalogar un libro
- Leer un libro en español

Los **programas** de estos ejemplos tienen características que no se encontraban en los ejemplos anteriores y que los convierten en casos mas particulares de lo que hemos llamado **Especificacion de Acciones**:

- Son mas complejos, ya que no comprenden solo una secuencia de acciones o actividades, sino que incluyen tomas de decisiones en casos en que se presenta mas de un posible camino a seguir.

-Son mucho mas precisos, ya que no dejan esas decisiones al libre arbitrio de quien ejecuta el programa, sino que se especifica en forma mas o menos clara cuales son las condiciones bajo las cuales se debe elegir uno u otro camino. P ej.: Si no se encuentra el libro en catalogo por autor, buscar por titulo; o: si existe la Autoridad de Autor, Confirmar si esta de acuerdo a las reglas II; Si no lo esta, hacer la investigacion correspondiente.

-Tienen caracter repetitivo, es decir, son acciones que se ejecutan muchas veces: cada vez que es necesario hacer una suma, "ejecutamos" el programa que suma dos numeros enteros (poner los numero uno sobre el otro, alinearlos a la derecha; sumar los digitos de mas a la derecha ; si el resultado es un numero de mas de una cifra, anotar el digito de mas a la izquierda del resultado (reserva) sobre los digitos siguientes de los sumandos ...etc).

### **El concepto de ALGORITMO.**

Todo programa hace uso de ciertos conocimientos previos de quien lo va a ejecutar. Para hacer la suma de dos numeros enteros de varios digitos es necesario saber previamente sumar dos numeros enteros de un digito, tener el concepto de 'encima', 'alinear', 'derecha', 'mas a la derecha', 'izquierda', etc.

El Ser Humano tiene un conjunto de conocimientos adquiridos que le permite enfrentar ciertos problemas haciendo deducciones ( a veces subconcientes ) que le permiten tomar las decisiones apropiadas y ejecutar las acciones necesarias para solucionarlos.

Por ejemplo, si a una persona se le asigna como tarea mantener llenas las copas de los invitados durante una fiesta, se le esta entregando un programa que recurre a una gran cantidad de conocimientos previos para solucionarlo, como por ejemplo : donde estan las bebidas, cuales hay que mantener heladas, donde esta el refrigerador de la casa, como se preparan las combinaciones de bebidas mas tradicionales, como mantener un stock de copas limpias, etc, etc, etc.

Un programa de este tipo puede llevar a distintos resultados dependiendo de quien lo ejecute: Una persona sin iniciativa y/o los conocimientos minimos necesarios para cumplir la tarea puede hacer fracasar la fiesta. Otra que se ubique perfectamente en la situacion puede hacer que la fiesta sea un exito.

Un programa que al ser ejecutado por distintas personas que poseen el mismo set de conocimientos previos lleve siempre al mismo resultado se llama **ALGORITMO**.

El programa que suma dos numeros enteros de varias cifras es un algoritmo. En general los procedimientos para resolver problemas aritmeticos son algoritmos. Tambien lo son las especificaciones de flujo de trabajo, aunque en ellas se recurre a un conocimiento mas profundo que hace que el conjunto de personas que lo puedan llevar a cabo sea mas reducido, y el resultado pueda variar en algunos aspectos: Un libro puede ser catalogado solo por Bibliotecarios, mas aun, por bibliotecarios con experiencia en catalogacion; por otra parte un mismo libro puede recibir catalogaciones levemente diferentes si lo catalogan personas distintas.

Como se ve, un **Algoritmo** es un tipo de **Especificacion de Acciones** mas precisa aun que las anteriores, y su precision se basa en gran medida en

-Definir exactamente los elementos minimos que debe conocer previamente quien ejecutara el algoritmo.

-Definir exactamente la secuencia de acciones y las decisiones que debera tomar.

Sabiendo que es un algoritmo, podemos contestar mas facilmente, entonces, la pregunta que nos interesa.

### **QUE ES UNPROGRAMA COMPUTACIONAL**

El computador tiene un Conocimiento Previo muy pequeño y tiene el problema adicional de que no es capaz de incrementarlo como los seres humanos.

La consecuencia mas importante de esto es que esta maquina es incapaz de hacer ninguna deducción por si sola. Esto obliga a que los programas escritos para ser ejecutados por ella deban ser muy precisos.

La precision reside en que:

-Solo debe incluirse instrucciones que ella conozca.

-Las instrucciones deben estar especificadas exactamente en el orden en que deben ser ejecutadas.

-Si hay que tomar una decision, se debe indicar los casos que se pueden dar y para cada uno de ellos la accion o acciones que se deben ejecutar.

Es necesario saber entonces cuales son las Acciones o Instrucciones que un computador conoce.

### Instrucciones que conoce un computador

Aunque los fabricantes de computadores estan constantemente cambiando las características de los equipos, y a pesar de que hay variaciones a veces significativas entre los distintos modelos y marcas, se puede decir que todos los computadores conocen un grupo mas o menos igual de instrucciones, en base a las cuales se construyen los programas.

Estas son, resumidamente, las siguientes:

- Hacer calculos sobre datos numericos (sumarlos, restarlos, multiplicarlos, elevar uno al otro, etc)

- Pedir datos al usuario (Leer)

- Entregar resultados al usuario (Escribir)

- Hacer comparaciones ( Es  $A > B$  ? )

- Evaluar una aseveracion ( Es verdad que  $A = 5$  )

- De acuerdo al resultado de una Comparacion o Evaluacion tomar uno entre dos caminos alternativos (Si  $A > B$  escriba A, sino escriba B)

- Mover datos dentro de la memoria (move A to B, o como se usa en el Lenguaje PASCAL, que usamos en nuestra unidad,  $B := A;$ )

- Ir a un paso dado dentro de la secuencia de instrucciones.

Con estos elementos basicos se construyen todos los programas computacionales a traves de combinaciones mas o menos complejas de ellos.

En la figura 1 se muestra el programa **M** (por **Mini**), que lee un dato de la pantalla, le suma 1, si el resultado es  $> 5$  lo escribe y si es  $< 0 = 5$  lo multiplica por 5 y lo escribe.

En la parte b) de la figura se muestra a que tipo de instruccion corresponde cada uno de los pasos del programa **M**.

M

- 1. - LEER UN N° DESDE PANTALLA
- 2. - SUMARLE 1
- 3. - SI EL RESULTADO ES > QUE 5
  - (a) ESCRIBIR EN PANTALLA ELSE
  - (b) MULTIPLICARLO POR 5
  - (c) ESCRIBIRLO EN PANTALLA

a)

- 
- 1: LEER
  - 2: HACER CALCULO
  - 3: COMPARAR Y TOMAR CAMINOS DISTINTOS DE ACUERDO AL RESULTADO

- 3a ESCRIBIR
- 3b HACER CALCULO
- 3c ESCRIBIR

b)

Figura 1

En Pascal este programa tendria el siguiente aspecto:

```
Program M;           --      COMENTARIOS
  var A:integer;    -- Se le dice al comp que se
                    -- usara un dato entero llamado A.
begin              -- Comienza el programa.
  read(A);         -- Lee el dato (lo pideal usuario).
  A:=A+1;         -- Le suma 1.
  if A>5 then     -- Hace la comparacion.
    write(A)      -- >5 , escribe A.
  else            -- En caso contrario...
    begin        -- Hagatodolo que sigue hasta el
                -- primer 'end' que encuentre.
      A:=A*5;    -- Multiplica A por 5.
      write(A);  -- Escribe el resultado.
    end         -- Encuentra el end.
end.           -- Fin del programa.
```

Como se ve a todo programa se le debe decir cuales son los datos sobre los cuales va a trabajar y luego las acciones que se debe efectuar sobre esos datos.

### **EL COMPUTADOR POR DENTRO.**

#### **Como entiende el computador estas instrucciones**

El computador no entiende directamente estas instrucciones. Cada computador tiene un lenguaje propio y solo lo que se le diga en ese lenguaje es comprensible para el.

A tal lenguaje se le llama Lenguaje de Maquina. Todo lenguaje que no sea el de maquina debe ser "traducido" para ser comprensible al computador.

#### **Elementos del lenguaje de maquina**

Partamos por lo mas basico.

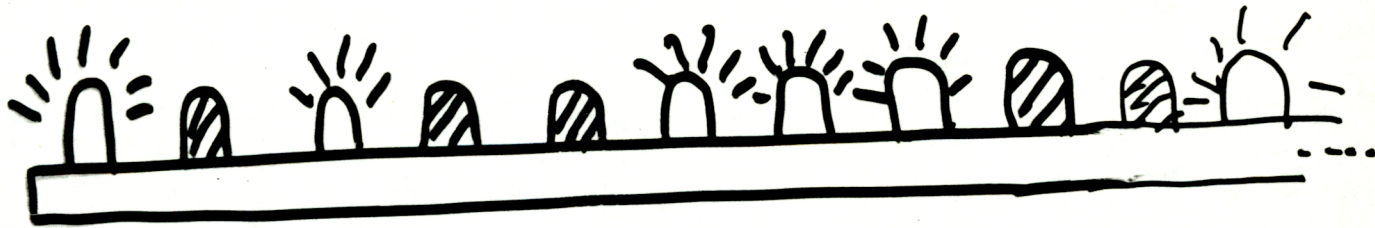
El computador almacena en su memoria los datos e instrucciones con los cuales operara.

La memoria esta compuesta de una gran cantidad de elementos que pueden tomar solo dos estados. Cada uno de ellos se llama BIT.

(Ver figura 2)

(Aunque no es esta exactamente la forma de los elementos basicos de la memoria, sirve para el ejemplo).

Si queremos representar numeros, solo podemos representar el cero (0) y el uno (1).



Representación Simplificada  
de la Memoria.

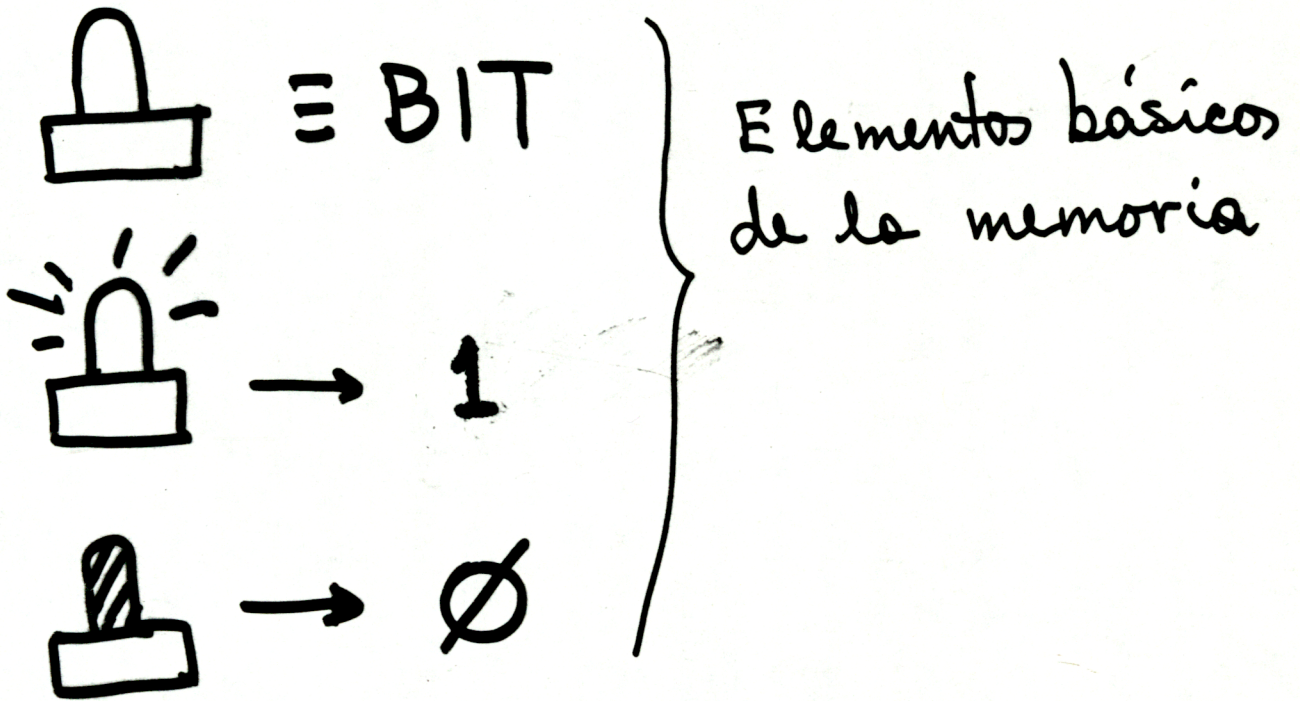


Figura 2.-

Como podemos representar el resto de los numeros con BITS?. La solucion es agrupar de a varios BITS. Por ejemplo con 2 BITS podemos representar los numeros 0, 1, 2 y 3. (Ver Figura 3).

Con 4 podemos representar los numeros del 1 al 16 lo cual nos da un mayor margen aunque todavia es insuficiente. De hecho con cuatro bits podemos representar todos los digitos del 0 al 9, por lo cual a un grupo de 4 bits podriamos llamarlo Dígito. Como se puede representar 16 cosas con ellos, se les llama tambien Dígito Hexadecimal, y a las 16 combinaciones mostradas en la siguiente figura se les llama 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F.(Ver figura 4).

Con dos digitos hexadecimales podemos representar 256 simbolos distintos (numeros y letras del alfabeto, o instrucciones, etc.).(ver figura 5).

¿COMO REPRESENTO EL 2? ⑧

$$\begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \equiv \emptyset \emptyset = \emptyset$$

$$\begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \equiv \emptyset 1 = 1$$

$$\begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \equiv 1 \emptyset = 1$$

$$\begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \begin{array}{|c|} \hline \text{[ ]} \\ \hline \end{array} \equiv 1 1 = 2$$

COMO REPRESENTO

0, 1, ..., 9, A, ..., Z, ., /, etc

Figura 3.-

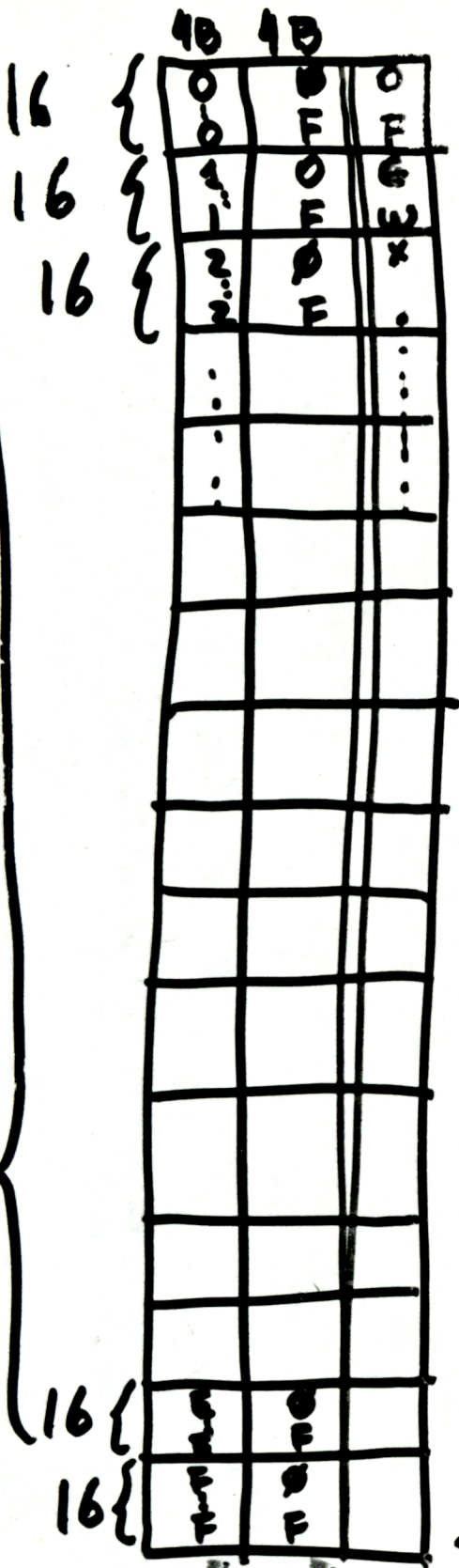
4 BITS

9

0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	A
1	0	1	1	B
1	1	0	0	C
1	1	0	1	D
1	1	1	0	E
1	1	1	1	F

16 símbolos

Figura 4.-



256 símbolos  
 (INSTRS  
 DATOS  
 DIRECCIONES)

Byte

DATOS

DIGITOS HEXADECIMALES

Figura 5.-

En un lenguaje de maquina necesitamos representar 3 tipos de cosas:

- a) Instrucciones
- b) Nombres de Datos
- c) Contenido de los Datos

### **Las Instrucciones**

Como vimos recién con dos digitos hexadecimales podemos representar hasta 256 cosas distintas. Supongamos que tenemos dos maquinas, una con 200 instrucciones y otra con 250 (ver figura 6).

De esta manera cada vez que queramos mover informacion desde el registro interno de la CPU a la memoria, usaremos la instruccion FA en la maquina B.

### **Los Nombres de Datos**

Las instrucciones actuan sobre datos. Cada dato tiene un nombre y un valor (por ejemplo, una variable puede llamarse A y contener un 10). En lenguaje de maquina no se trabaja con nombres de variables sino con DIRECCIONES de las variables. Asi, la variable que nosotros llamamos A es llamada internamente por su direccion, p. ej. 7D.

Las direcciones tambien se representan en hexadecimal. En la figura 7 se representa una maquina con 16 celdillas. La celdilla 1 se llama 0, la 2 se llama 1, la 3 02, etc, hasta la 16, que se llama F. En esta maquina se necesita un solo digito hexa para dar nombre a todas las celdillas.

En otro caso, si tenemos 256 celdillas para los datos, se necesita dos digitos hexa para representar todas las direcciones de la memoria. Para mayor numero de celdillas se necesitan mas digitos hexa. Nuestros equipos tienen 65536 celdillas. Para representar las direcciones de todas ellas, se necesita 4 digitos hexa, con los que se representan los numeros 0 al 65535.

### **Contenido de los Datos**

El contenido de una celdilla de datos puede ser una letra o un numero o un signo especial. Con dos digitos hexa podemos representar 256 cosas. Supongamos que hacemos una tabla de equivalencia, como la de la figura 9. Con ella "traducimos" nuestros datos a la forma en que la maquina los entiende.

# LAS INSTRUCCIONES

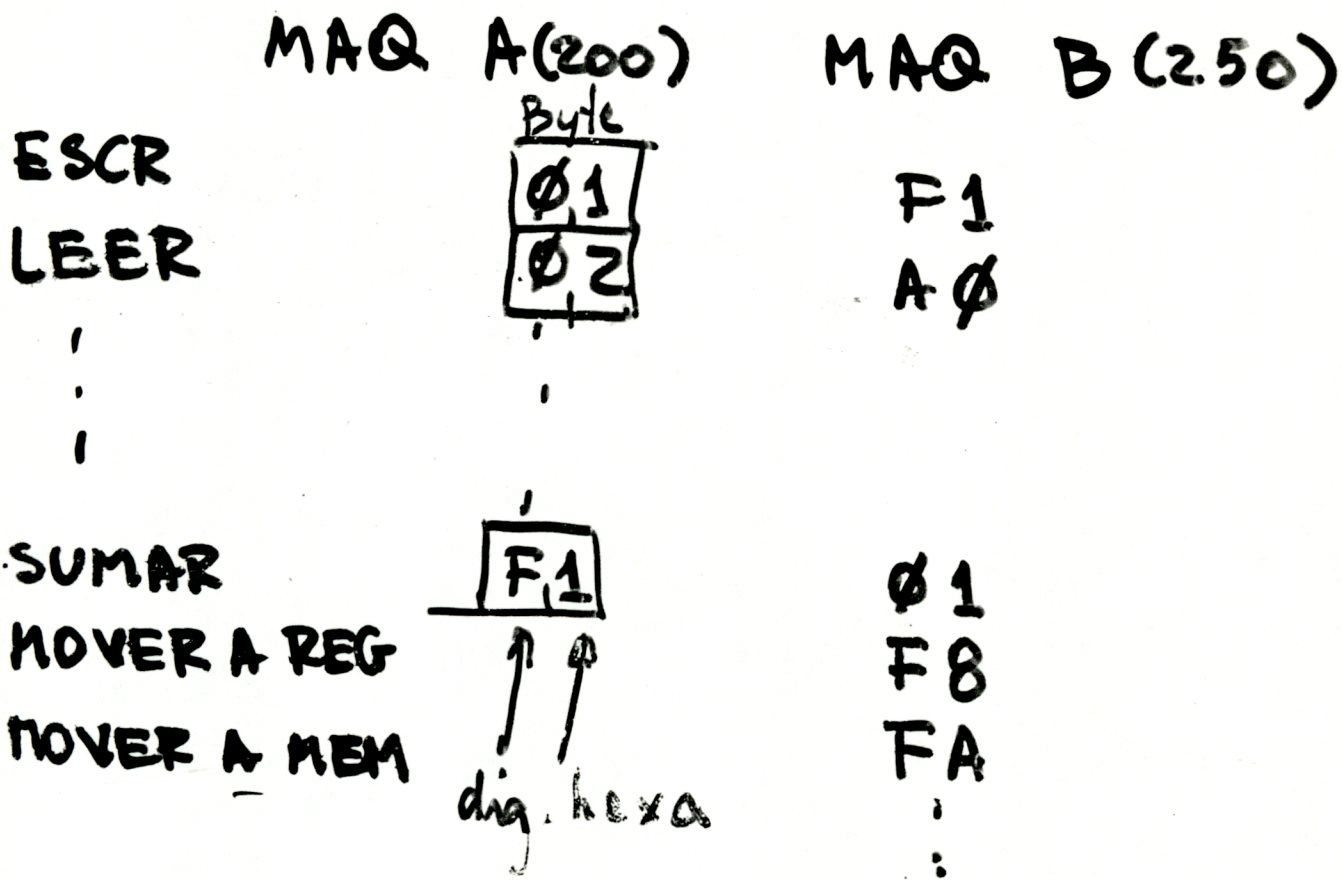


Figura 6.-

# LAS DIRECCIONES

SE DIRECCIONA UN BYTE

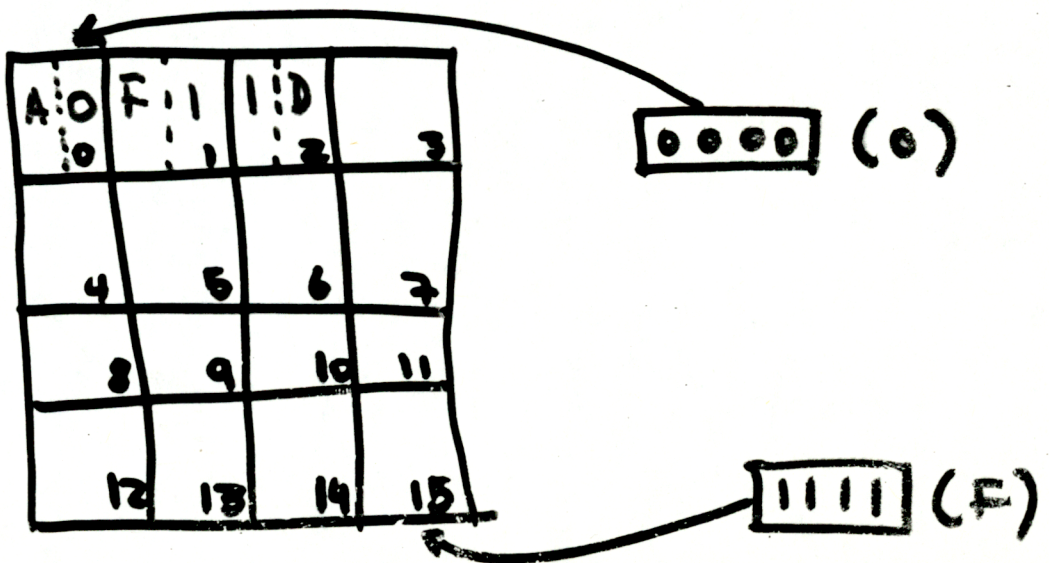


Figura 7.-

Supongamos que queremos representar un numero 856 , su representacion interna sera

080506

El dato ABC sera representado por

101112

etc.

### El Programa

Un programa en lenguaje de maquina es un grupo de PARES Instruccion-Direccion (que se debe hacer y sobre que dato), seguido de los datos que seran afectados.

En la figura 8 hay una maquina de 16 celdillas, que contiene un programa sencillo.

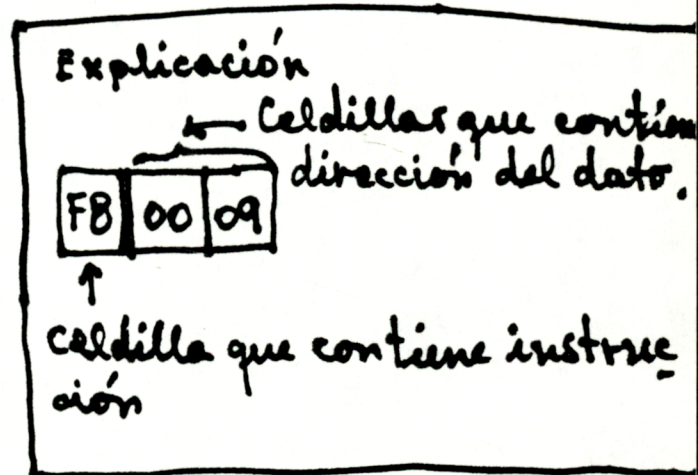
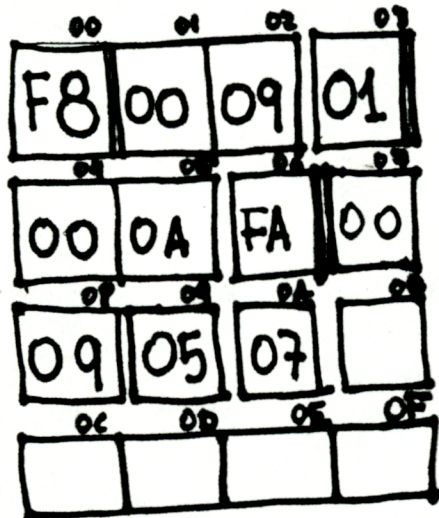


Figura 8.

## **Compiladores**

Actualmente es muy poca la gente que programa en lenguaje de maquina. Se ha inventado un metodo que permite programar en lenguajes mas humanos, que hacen mucho mas rapida y agradable la tarea. El metodo consiste fundamentalmente en dos puntos:

- Inventar un lenguaje muy preciso, que permita programar el computador, que ademas sea sencillo de aprender y tecnicamente completo.
- Escribir un programa (en lenguaje de maquina) que sea capaz de leer un programa escrito en el lenguaje inventado y lo traduzca a lenguaje de maquina.

Una vez que se tiene estas dos cosas, cualquiera que aprenda el lenguaje inventado puede escribir programas en el y luego traducirlos a lenguaje de maquina con el traductor.

A los lenguajes inventados de esta forma se les llama De Alto Nivel y a los programas traductores se les llama Compiladores.

El lenguaje PASCAL, usado por nuestra unidad de sistemas es De Alto Nivel.

Actualmente todos los programadores (o casi) escriben en lenguajes De Alto Nivel (como el PASCAL) y luego los Compilan, dejandolos listos para ser ejecutados cada vez que sea necesario.

## **Sistemas Operativos**

Los computadores tienen una serie de recursos: Discos, Diskettes, Pantallas, Impresoras, etc.

El manejo de estos recursos es muy complejo y se necesita saber bastante de la estructura fisica de la maquina para hacerlo. Los fabricantes de computadores venden junto con la maquina un programa mas o menos complejo que es capaz de manejar todos esos recursos y ayudar al usuario con algunas labores basicas. Este programa se llama Sistema Operativo. Al encenderse el computador, este programa se ejecuta automaticamente, y queda en ejecucion hasta que se apaga. Cuando se quiere ejecutar un programa, se le pide al S.O. que lo haga mediante una instruccion que permite copiar el programa desde el disco a la memoria y luego lo ejecuta. Cuando termina la ejecucion del programa, este entrega el control nuevamente al S.O. Todas las operaciones de manejo de recursos que se hacen en un programa, se hacen mediante llamados al S.O. el que ejecuta al funcion deseada y entrega lo necesario al programa.

El S.O., como dijimos antes, además contiene algunas ayudas básicas para el usuario, como por ejemplo, programas pequeños que muestran el directorio de un disco, o copian archivos de un disco a otro, o que borran archivos del disco, etc.

### **Notas Finales**

Hemos intentado aclarar en unas pocas páginas un grupo de conceptos básicos y fundamentales de computación. Normalmente estos conceptos son enseñados en varios cursos separados por su complejidad. No es la intención de estos apuntes dejar todo claro, sino dejar algunas ideas sobre las cuales el (la) lector(a) podrá profundizar personalmente y/o con la ayuda del personal de nuestra Unidad.

ARCHIVOS Y BASES DE DATOS  
=====

Hugo Cisternas.

	Pagina
I.- INFORMACION.	2
II.- ALMACENAMIENTO MASIVO.	7
REGISTROS.	9
ARCHIVOS.	13
III.- ACTUALIZACION	18
IV.- LIMITACIONES DE LOS SISTEMAS TRADICIONALES	21
V.- BASES DE DATOS	22
VI.- SISTEMAS DE INFORMACION	27

## I.- INFORMACION.

=====

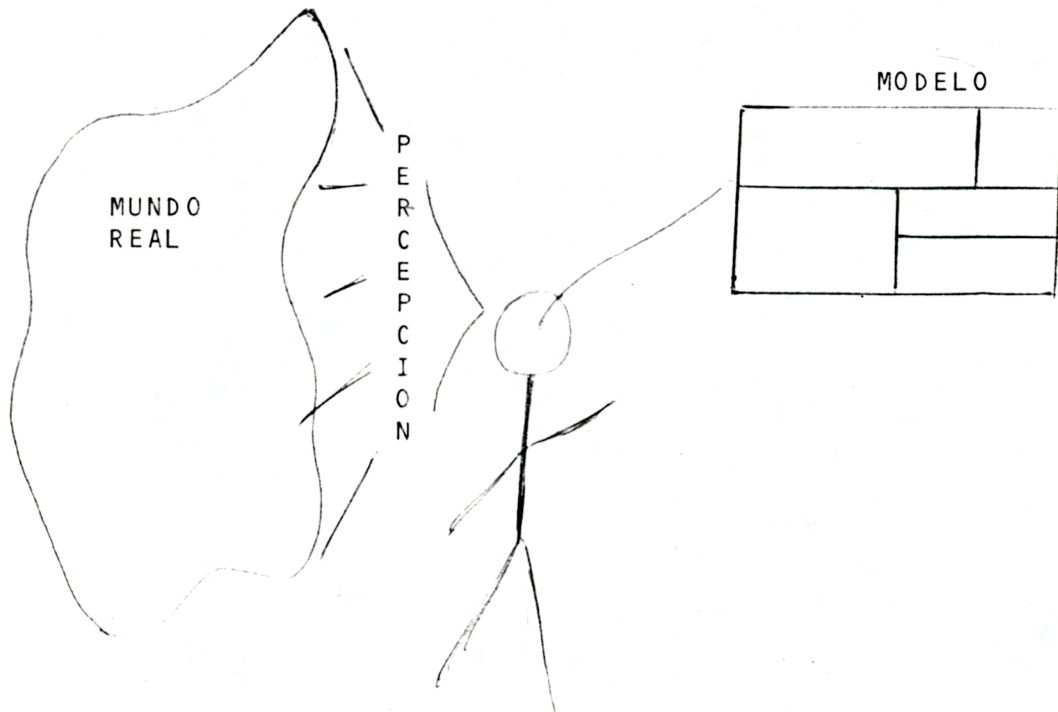
La informacion se podria definir, en una primera aproximacion, como el "conocimiento derivado de la observacion".

Esta definicion se refiere, implicitamente, a un **MUNDO REAL**, al cual observamos ( o mas precisamente, **percibimos** ), y a un proceso de **abstraccion**, a travez del que nos hacemos un **modelo** del mundo real.

En este mundo real existen **objetos**, que son precisamente aquellas "cosas" de las que queremos obtener informacion. Estos objetos son tratados como un todo; sin embargo, no necesitan ser efectivamente indivisibles ( atomicos ), es decir, pueden ser compuestos.

Estos objetos que percibimos tienen una serie de **propiedades o atributos**.

A travez de la observacion del mundo real ( percepcion ) nos creamos una **imagen**. Esta imagen es una abstraccion del mundo real y sera entonces, un **modelo** de este mundo real. De hecho, es imposible para nosotros lograr una imagen exacta y completa del mundo real, debido a que nuestra percepcion es muy limitada; por esta razon, en nuestra vida diaria manejamos ( inconscientemente ) modelos.



## **ABSTRACCION**

=====

A travez de el proceso de **abstraccion**, nosotros vamos a nombrar y clasificar los objetos del mundo real, construyendo, como hemos visto, un modelo. Para este proceso, usamos un elemento que llamaremos alfabeto.

## **ALFABETO**

Un alfabeto es un conjunto finito. En general este conjunto estara compuesto de "signos", por ejemplo: letras, simbolos, numeros, etc. Estos signos tienen, en general, una representacion fisica, que nos permite comunicarnos.

En base al alfabeto, construimos un **lenguaje**, que nos permitira nombrar a los objetos que percibimos.

Otro elemento abstracto que es necesario destacar son los **valores** o medidas.

## **VALOR**

Podemos decir que un valor es una cantidad, un numero. Es evidente que un valor tomado aisladamente, no tiene ningun significado. Un valor adquiere significado como expresion de una propiedad o atributo en un objeto determinado, esto es, una medida cuantitativa o cualitativa de una propiedad.

Veamos algunos ejemplos:

- Supongamos la propiedad **color**.  
Al "medir" esta propiedad en un objeto, podremos decir que el objeto es de color rojo.

Entonces, **rojo** es un **valor** y **color** es un **atributo**

- Supongamos la propiedad **altura**.  
Al medir la altura de un objeto, podriamos obtener como resultado que el objeto tiene una altura de **tres metros**.

Entonces, **tres metros** es un **valor** y **altura** es un **atributo**.

Se llama UNIVERSO de valores, al conjunto de todos los valores posibles.

Dentro de este universo de valores, es importante distinguir dos valores especiales:

$\Omega$  es un valor faltante pero irrelevante.

$\emptyset$  es un valor faltante e importante.

Ejemplos:

1.- En personas, medimos si han hecho el servicio militar. Si no se dispone de la respuesta de una persona determinada, tendremos dos posibilidades de asignar el valor:

pondremos  $\Omega$  si es mujer y  
 $\emptyset$  si es hombre.

2.- Al catalogar un libro, pondremos  $\emptyset$  si no disponemos de un campo obligatorio, pero pondremos  $\Omega$  si el campo faltante no es obligatorio.

OBSERVACION:

La eleccion entre  $\emptyset$  y  $\Omega$  es en general muy relativa, como en el ejemplo 1.

Un "elemento de informacion" estara compuesto, entonces, por una triada que lo definira totalmente:

- OBJETO
- ATRIBUTO
- VALOR

Es decir, un elemento de informacion consistira de un objeto determinado, un atributo que el objeto tiene (o no tiene), y un valor que es el resultado de medir el atributo en el objeto en cuestion.

A esto se le llama una "triada ordenada",

( objeto, atributo, valor )

La informacion la expresaremos entonces por un conjunto de triadas ordenadas.

Por lo tanto, este conjunto de triadas ordenadas constituye lo que llamaremos el **espacio tridimensional** de la informacion. Este espacio tendra pues, tres ejes: el eje de los objetos, el eje de los atributos y el eje de los valores; un elemento de informacion sera entonces un punto determinado en este espacio, definido por estas tres **coordenadas**.

Muchas veces es necesario introducir una cuarta dimencion en este espacio de la informacion: el **tiempo**.

#### **DATO.**

Un DATO puede ser definido, entonces, como un conjunto finito de simbolos fisicos de un alfabeto, que son usados para representar un elemento de informacion, con fines de almacenamiento, comunicacion o procesamiento.

Es importante notar que dato no es sinonimo de informacion, sino que es una representacion fisica de la informacion. En un sistema computacional usaremos distintos "tipos" de datos, que seran distintas formas de representar la informacion, para poder procesarla.

## RESUMEN

=====

### OBJETO.

- En el mundo real, todos los objetos son distinguibles.
- Los objetos no necesitan ser atómicos e indivisibles. Pueden ser compuestos.
- Sin embargo, cada objeto es tratado como un todo.

### ATRIBUTOS o PROPIEDADES.

- Un atributo o propiedad es "algo" que un objeto tiene ( o no tiene ).
- En nuestra conversación normal, es denotado por una frase verbal que no necesariamente requiere de un objeto gramatical.

"es feo" ==> apariencia.

"es rojo" ==> color.

### MODELO.

Un modelo es una imagen abstracta de la realidad.

### ALFABETO.

Un alfabeto es un conjunto de signos ( letras, números, etc. )

### DATO.

Los datos son representaciones físicas de la información. Para esto se utilizan símbolos físicos del alfabeto.

## II.- ALMACENAMIENTO MASIVO

=====

Para almacenar informacion necesitamos un **soporte**.  
Este soporte podria ser:

- .Papel (libros, revistas, etc)
- Cintas magneticas.
- Discos magneticos
- etc, etc, etc.

Nos preocuparemos, en este momento, de los soportes magneticos, que constituyen la base del almacenamiento masivo en computadores.

Dentro de los soportes magneticos, podemos distinguir los soportes portables, como son las cintas, diskettes y discos removibles, que permiten transportar informacion entre diferentes maquinas, y los soportes fijos, como son los discos winchester y los tambores magneticos, que permiten gran volumen de almacenamiento y velocidad de recuperacion pero no son transportables.

Estos soportes estan compuestos de una delgada capa de material magnetizable ( p ej. oxido de fierro ) sobre una base que puede ser de material plastico flexible en el caso de los diskettes y cintas magneticas, o metal en el caso de los discos "duros".

La informacion, en este tipo de soportes, es almacenada en la forma de puntos magnetizados ( algo similar a la grabacion de cassettes de musica ).

Las caracteristicas del soporte magnetico nos imponen una serie de cuidados especiales que es de importancia mencionar. Dado que las cintas y diskettes son portatiles, estan expuestos a muchos riesgos que pueden afectar la informacion almacenada en ellos. Asi por ejemplo, la precencia de campos magneticos en las proximidades de un diskette o cinta puede alterar y borrar el contenido de esta. Los imanes, los cables electrificados, transformadores, y diversos aparatos electronicos generan estos campos magneticos, y es necesario evitar su cercania. El polvo y la humedad son otro factor importante de daños.

Trataremos a continuacion dos puntos importantes para el almacenamiento masivo de informacion.

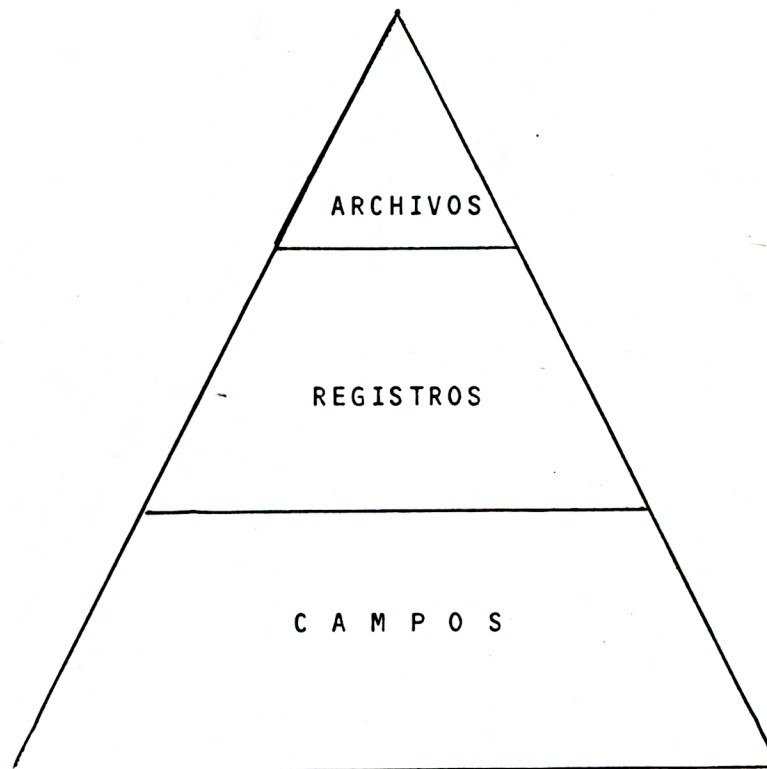
- REGISTROS

- ARCHIVOS

Al examinar los registros nos referiremos principalmente a los diferentes "formatos" de registros.

Luego, veremos diferentes tipos de archivos tradicionales y su organizacion.

Mas adelante, veremos las BASES DE DATOS.



## REGISTROS.

=====

Un registro es un conjunto de datos en un formato que permite su manipulacion por un computador.

El formato de un registro corresponde a la "forma" en que los datos estan dispuestos dentro del registro. Entonces, el formato nos dara la **receta** de como obtener un dato determinado dentro del conjunto de datos que componen el registro. El formato, entonces, nos dira donde y como esta cada dato; y a este lugar dentro del registro, donde se encuentra un dato, lo llamaremos un **CAMPO**.

Los formatos mas usados son:

- Posicional
- Relacional
- Subindicado
- Con rotulo

En general, en sistemas de informacion de mediana a gran envergadura, se utilizan combinaciones de estos formatos, tratando de aprovechar las ventajas de cada uno. Ejemplo de esto es el formato M.A.R.C., que es una combinacion de todos estos formatos.

Veamos cada uno de ellos en detalle. Para mayor claridad, veremos ejemplos de cada uno de ellos, en base la misma informacion.

Supongamos que nuestros "objetos" son libros, y de ellos medimos los atributos: autor, titulo, año de edicion y numero de clasificacion Dewey. Tendremos por ejemplo:

- Autor: Messiah, Albert
- Titulo: Mecanica cuantica
- Fecha: 1973
- Dewey: 530.12

## FORMATOS

### 1.- POSICIONAL.

En un formato posicional, los campos estan en posiciones fijas, predefinidas y conocidas, y son de largo fijo.

Ejemplo: Definamos un registro de de largo 50 caracteres,

campo	largo	posicion
autor	20	1
titulo	15	21
fecha	4	36
Dewey	11	40

Messiah, Albert	Mecanica cuanti	1973	530.12
-----------------	-----------------	------	--------

1	21	36	40

#### Ventajas:

- Es util cuando los valores de los atributos varian entre rangos conocidos y no dispersos.
- Es muy simple de programar.

#### Desventajas:

- Puede producir truncamiento en los campos (ver ejemplo ).
- Obliga a dejar muchos espacios en blanco.

### 2.- RELACIONAL.

En un registro relacional, se coloca un separador o **delimitador** entre cada campo. El delimitador no pertenece al alfabeto de los datos, es decir, es un simbolo que no se puede usar en los datos.

#### Ejemplo:

Representaremos el delimitador con el simbolo #.

Messiah, Albert#Mecanica cuantica#1973#530.12
---

Ventajas:

- Permite manejar campos de largo variable sin perder espacios en blanco ni truncar.
- Es bastante simple de programar.

Desventajas:

- Es necesario conocer y mantener el orden entre los campos.
- Es necesario colocar el delimitador aun cuando no hay dato en un campo, para no alterar el orden.
- Es necesario "revisar" el registro desde el comienzo para obtener un campo ( no sabemos a priori la localizacion del campo ).

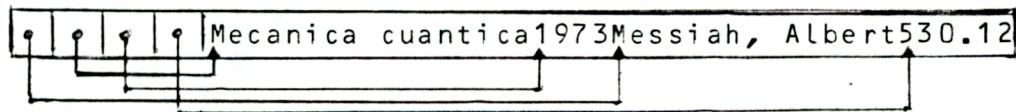
3.- SUBINDICADA.

Al comienzo del registro va una descripcion de su contenido, que puede ser de las siguientes formas:

- La posicion del comienzo de cada campo.
- La posicion del ultimo caracter de cada campo.
- El largo de cada campo.

A esta parte del registro se le llama "encabezamiento", y es de largo fijo y conocido.

Ejemplo: el indice apunta al comienzo de cada campo.



Ventajas:

- No es necesario revisar todo el registro para obtener un campo; basta con el encabezamiento.
- No se requiere un orden especial entre los campos.

Desventajas:

- El encabezamiento puede ocupar mucho espacio.
- Si un campo no tiene dato, es necesario colocar un "indice nulo" para preservar el orden.

#### 4.- CON ROTULO.

A cada campo se le antepone una identificacion del atributo al cual se le llama "rotulo".

Los rotulos pueden ser de cualquier largo y constituyen **palabras reservadas**, es decir no pueden aparecer como parte de ningun campo para evitar ambigüedades.

Ejemplo:

campo	rotulo
autor	R1
titulo	R2
fecha	R3
Dewey	R4

```
R1Messiah, AlbertR31973R4530.12R2Mecanica cuantica
```

```
==                ==      ==      ==
```

Ventajas:

- Si hay un campo sin dato, simplemente el rotulo no aparece.
- Los campos pueden ir en cualquier orden.

Desventajas:

- La eleccion de los rotulos es un serio problema para evitar toda ambigüedad.
- Para encontrar un campo se debe revisar el registro desde el comienzo. No se puede saber a priori su localizacion.

OBSERVACION:

Cuando nos referimos a un campo sin dato, significa que el valor medido es  $\Omega$ , es decir, un valor que falta y es irrelevante.

## ARCHIVOS

=====

El termino "archivo" es uno de los mas antiguos en el mundo de la computacion, y proviene de mucho antes aun que el advenimiento de los computadores. Se llamaba ( y aun hoy se usa en las oficinas ) archivo a los gabinetes donde se guardaban papeles, tarjetones contables, etc. Un archivo se referia entonces, al lugar fisico donde se guardaba informacion. Esta relacion era perfectamente valida en los tiempos de las tarjetas perforadas ( "archivo de tarjetas" ), y luego con las cintas magneticas. Con la irrupcion de los discos magneticos, en los que es posible realizar acceso directo, la definicion comienza a apartarse de sus origenes y aparecen diferentes tecnicas de "acceso de archivos".

### DEFINICION:

Un archivo es un conjunto de registros sin ninguna relacion entre ellos salvo la de estar en el mismo archivo.

El concepto de archivos puede ser confundido con el concepto de bases de datos. Sin embargo, si bien ambas son un conjunto de registros, en el caso de las bases de datos, existen variadas relaciones entre los registros que la componen.

Recordemos de la pagina 4, el espacio de la informacion. El concepto de archivo esta estrechamente ligado al espacio de la informacion. Es mas, podemos decir que un archivo representa un subconjunto de este espacio (una representacion fisica de este subconjunto); veamos en detalle:

- a) Cada registro de un archivo (o un conjunto determinado de registros) esta asociado a un objeto de nuestro espacio de informacion; asi por ejemplo, en un archivo de libros, cada registro representara un libro.
- b) Cada campo de un registro representara un atributo (en particular, del objeto asociado al registro); por ejemplo, el campo de autor esta asociado al atributo autor.
- c) Cada dato dentro de un campo, como hemos visto anteriormente, es la representacion fisica de un valor.

Entonces, podemos decir ahora, que al procesar archivos, estamos efectivamente procesando informacion. El problema es, generalmente, escoger el subconjunto adecuado del espacio de informacion sobre el cual vamos a trabajar, y estar concientes de la asociacion que estamos creando.

## TIPOS DE ARCHIVOS

Podemos clasificar los archivos segun diversos puntos de vista.

### Importancia.

Si dividimos los archivos segun su importancia relativa dentro de un sistema, tendremos:

- archivos ACTIVOS y PERMANENTES, que son aquellos que se estan utilizando permanentemente.
- Archivos temporales.
- Archivos de respaldo, que tienen muy poco movimiento ( mas vale! ) y son una copia de un archivo activo en un instante de tiempo determinado, que es usada en caso de que la version original se arruine.

### Volatilidad.

Dividimos los archivos en funcion de la frecuencia de la modificaciones sobre los registros del archivo.

### Acceso.

Segun el tipo de acceso utilizado, tendremos archivos

- Secuenciales: El archivo se recorre desde el comienzo hasta encontrar el registro que necesitamos.
- Directo: puedo obtener directamente el registro que me interesa.

Podemos clasificar los archivos de muchas otras formas aun, como por ejemplo, segun el numero de accesos por unidad de tiempo, segun el apoyo de software que se disponga para este archivo, etc.

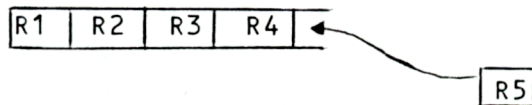
## ORGANIZACION DE ARCHIVOS

Como hemos visto, un archivo es un conjunto de registros almacenados en algun medio magnetico (en general). Necesitamos, entonces, alguna forma de recuperar estos registros una vez almacenados. Para esto, daremos a nuestros archivos una estructura interna claramente definida que nos permita diseñar una estrategia computacional de recuperacion.

A esta estructura interna la llamaremos **organizacion** del archivo. Existen una serie de organizaciones satandard, que veremos a continuacion.

### SECUENCIAL.

En una organizacion secuencial, los registros estan geograficamente proximos dentro del archivo. Cada nuevo registro se agrega "a la cola" de los anteriores.

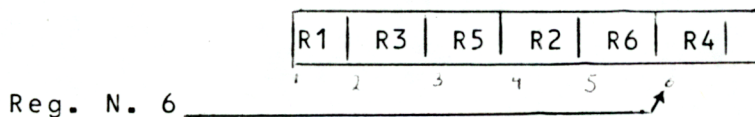


Este tipo de organizacion se usa principalmente cuando la volatilidad del archivo es alta y las actualizaciones se realizan por lotes (en batch), o para archivos de corta vida como son los archivos temporales o de respaldo. Esta es la organizacion tipica usada en las cintas magneticas.

Con la tecnologia actual de las bases de datos en disco, las redes de computadores y el trabajo en linea, se utiliza cada vez menos este tipo de organizacion debido a sus limitaciones y problemas de actualizacion y recuperacion selectiva.

### RELATIVO.

En este tipo de organizacion los registros se pueden obtener indicando su posicion relativa con respecto al primero del archivo. Se puede, entonces, obtener directamente un registro si sabemos en que posicion dentro del archivo se encuentra (ahi esta el problema!).



## INDEXADO SECUENCIAL.

En esta organizacion dividimos el archivo en tres partes (o usamos tres archivos):

- Area principal o de datos.
- Area de indices.
- Area de rebalse o de "overflow".

El area de datos esta dividida en "bloques" de registros. Dentro de estos bloques, los registros estan almacenados secuencialmente, y **ordenados** segun algun campo dentro de cada registro al que llamaremos **clave de busqueda**. Los bloques, a su vez, estan organizados en forma relativa, es decir, es posible obtener el **primer** registro de cada bloque en forma directa.

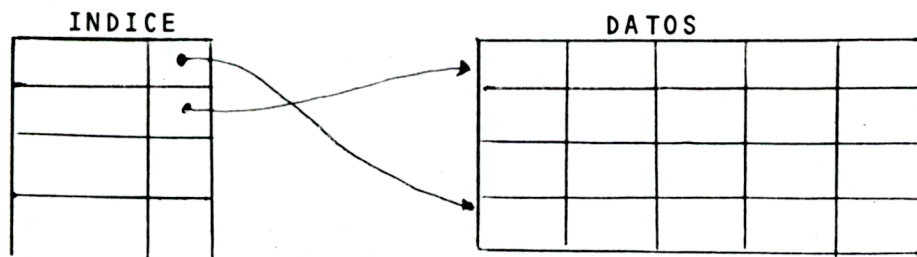
El area de indices esta formada por un tipo especial de registro llamados precisamente "indices", es decir, indican en que lugar se encuentran los registros que buscamos. En este tipo de organizacion, los indices contienen dos campos importantes: La clave de busqueda del ultimo registro de cada bloque y la posicion relativa de este mismo bloque. Estos registros estan ordenados a su vez por la clave de busqueda.

El area de rebalse, esta organizada en forma relativa, y contiene registros que se han "rebalsado" de un bloque.

La recuperacion de un registro en este tipo de organizacion se realiza de la siguiente forma:

Utilizando la clave de busqueda, consultamos el indice hasta encontrar el bloque donde deberia estar el registro. Del indice obtenemos la posicion de este bloque.

Recorremos secuencialmente este bloque hasta que suceda una de las siguientes condiciones: encontramos el registro o se nos acaba el bloque. si este bloque tiene una indicacion de que esta rebalsado seguiremos buscando en el area de rebalse, si no, significa que el registro no esta.



## PARTICIONADO.

Un archivo particionado es un archivo que reúne un conjunto de archivos secuenciales; es un "archivo de archivos".

A este tipo de organizacion se le suele llamar biblioteca de archivos, y consiste de un **directorio** de archivos y varios **miembros** que son los archivos que lo componen.

## LISTA INVERTIDA.

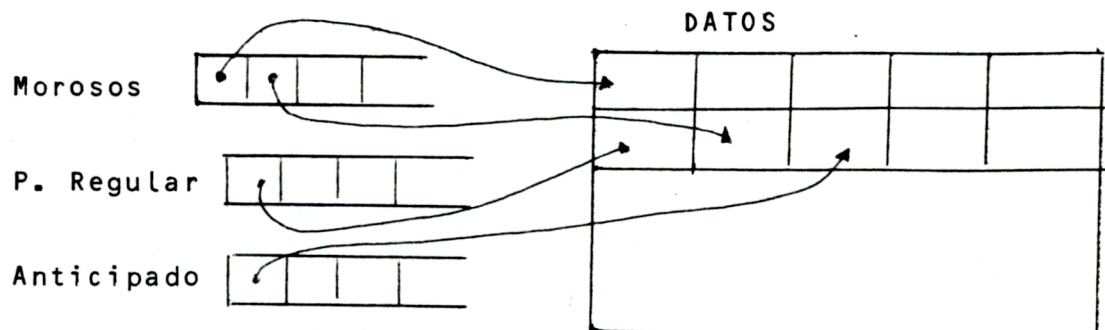
Este tipo de organizacion utiliza varios indices o directorios y un area de datos organizada en forma relativa.

Cada uno de los directorios hara referencia a un aspecto de un atributo explicito en los registros (este "aspecto" es en realidad un atributo implicito).

Por ejemplo, si tomamos un supuesto registro de usuarios de una biblioteca, y consideramos crear una lista invertida en funcion del campo **fecha de devolucion**, podremos tener tres directorios:

- Usuarios morosos (la fecha de devolucion esta excedida ).
- devoluciones en plazo regular.
- devoluciones anticipadas.

Esta organizacion nos permite obtener registros de determinadas características en forma selectiva.



### III.- ACTUALIZACION

=====

Al mencionar los archivos, hemos hecho referencia a la actualizacion. Este es un concepto de gran importancia en un sistema de informacion, y conviene presizarlo antes de referirnos a las bases de datos.

Existen diversos tipos de actualizacion o **updates**, que se derivan de conceptos anteriores a los computadores.

Veamos algunos ejemplos historicos, antes de presizar el concepto moderno de actualizacion.

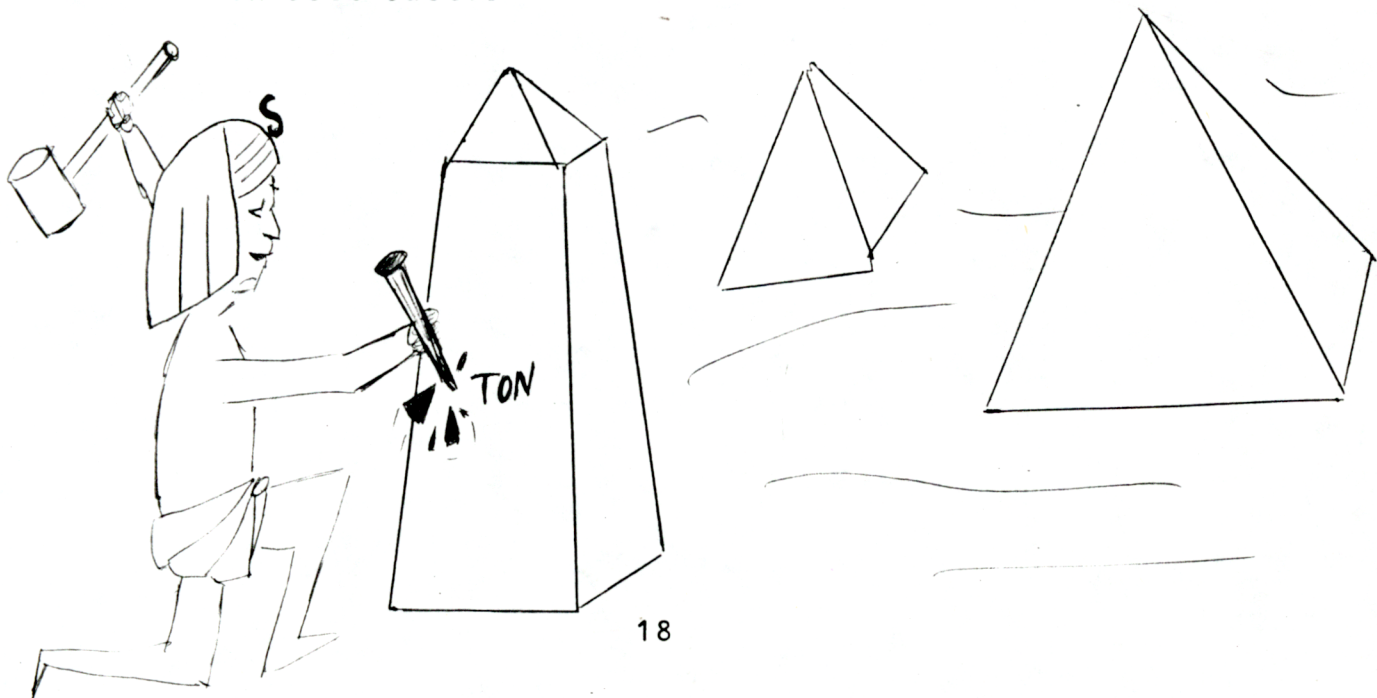
#### ATON UPDATE.

Alrededor del año 1400 A.C., el faraon que se hacia llamar AKENATON ( adorador de aton ), habia tratado de instaurar el monoteismo en egipto, presentando un dios unico: ATON.

Los sacerdotes de esta epoca tenian gran poder en egipto, y al ser desplazados sus cultos por el de aton conspiraron hasta lograr la muerte del faraon.

Logrado este objetivo, decidieron eliminar el nombre de aton de la faz de la tierra.

De esta forma, los registros de egipto sufrieron una "actualizacion" a fuerza de cincel y martillo. Esta es una actualizacion selectiva: se ubica la palabra ATON, y se procede a eliminarla (dejando un hoyo en la piedra en este caso).



#### ALEJANDRIA UPDATE.

En el año 47 A.C., la biblioteca de alejandria, que contenia la mayor cantidad de papiros de la antigüedad, se incendio quemandose todos los millones de papiros en su interior.

Decimos entonces que egipto sufrio una actualizacion no selectiva de sus registros, o en otras palabras, una actualizacion por **catastrofe**.

#### PALIMPSESTOS.

Durante la edad media, mucho antes de la invencion del papel, los monjes (que eran de los pocos que sabian escribir) utilizaban papiros.

El problema era que no siempre era facil de conseguir, de manera que para poder seguir escribiendo, utilizaban un tipo de papiro que se podia lavar, al cual se le llama **palimpsesto**.

De este modo, cuando se les acababan los papiros en blanco, revizaban los papiros mas antiguos y procedian a lavar los que contñian escritos que no sirvieran y continuaban escribiendo sobre ellos, es decir, los actualizaban.

Podemos decir entonces, que actualizar es "alterar" el contenido de un archivo, ya sea agregando, modificando o eliminando partes de un archivo (registros o campos de un registro) o el archivo en su totalidad. Se habla entonces de "poner al día", "limpiar" o "eliminar" un archivo, según el tipo de actualización que se realice.

El tipo de actualización y la forma en que esta se realice dependerá tanto de la organización del archivo como del soporte (y también de que el programa lo haga correctamente, cosa más difícil de verificar).

En los primeros tiempos de la computación, cuando se usaban los archivos en tarjetas perforadas, las actualizaciones eran simples: se reemplazaban las tarjetas correspondientes. El problema era que las actualizaciones por catastrofe ocurrían con excesiva frecuencia; bastaba que a alguien se le cayera un archivo al suelo, se le mojaran o arrugaran algunas tarjetas, se "trancara" la lectora, etc. Sin embargo, las actualizaciones del tipo ATON UPDATE no se podían realizar debido a las dificultades para "tapar" las perforaciones.

Con la invención de la cinta magnética (sucesora del palimpsesto) se creyó que se tenían solucionados los problemas, las cintas se podían reutilizar (como el antiguo palimpsesto), y no había forma de desordenar o "arrugar" los registros contenidos en ellas.

Sin embargo, contra lo que se podría pensar, las actualizaciones por catastrofe ocurren mucho más a menudo de lo que muchos quisieran admitir. Por otro lado, las actualizaciones selectivas tipo ATON UPDATE tampoco se pueden realizar, salvo quizás cortando la cinta en pedazos y volviéndola a pegar (el problema es saber donde cortar).

Las unidades de disco magnético de gran capacidad, el boom de nuestros días en materia de almacenamiento de información, nos permiten realizar actualizaciones selectivas sin ningún problema y la tecnología las ha hecho muy seguras (aunque no totalmente) contra actualizaciones por catástrofes físicas como la de Alejandría.

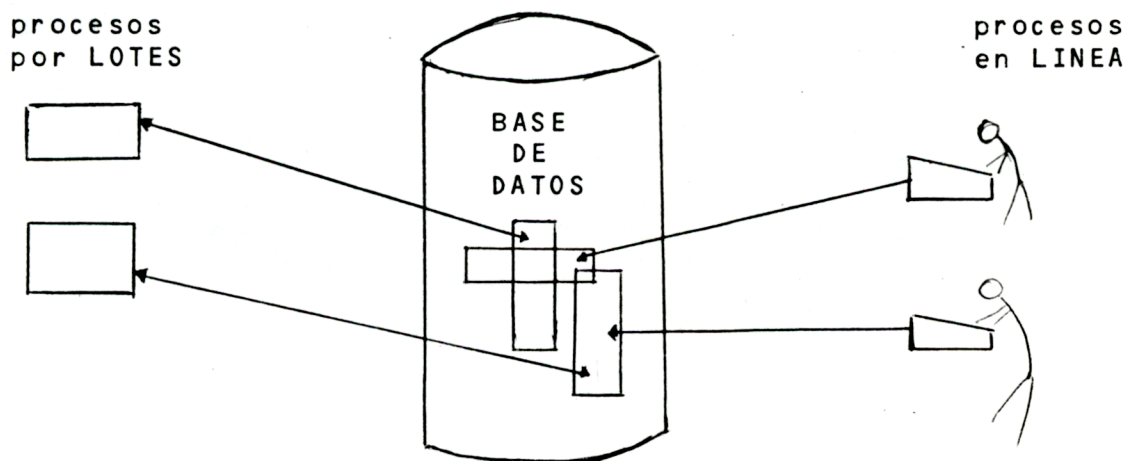
Sin embargo, muchos piensan que no son la panacea, y que las unidades de disco venían ya en la caja de Pandora, de la mano de la facilidad de escribir sobre un dato cada vez a alguien se le ocurra (o lo que es peor, cuando a nadie se le ocurre). Aparece entonces un nuevo tipo de catastrofe: las catástrofes por errores de software o acceso no restringido, que presentan los peores dolores de cabeza en el manejo de bases de datos.

#### IV.- LIMITACIONES DE LOS SISTEMAS TRADICIONALES.

Los sistemas de organizacion y manejo de archivos tradicionales, es decir, basados en las organizaciones de archivos que hemos visto, tienen una serie de limitaciones:

- Es necesario tener un archivo para cada necesidad, con lo cual tiende a proliferar la cantidad de archivos mas alla de limites practicos.
- Esto produce que diferentes archivos tengan informacion similar, creando una gran **redundancia de datos**
- La interrelacion de datos contenidos en distintos archivos, crea diseños complejos y programas sumamente complicados que deben ser capaces de conocer y manejar varios archivos en forma simultanea y por lo general con organizaciones diferentes.
- Si se decide cambiar el diseño o la organizacion de un archivo, es necesario corregir **todos** los programas que lo usan e incluso, reprogramarlos.
- Las actualizaciones en un archivo pueden causar **inconsistencia** en la informacion, si no se actualizan **todos** los archivos que tambien contienen los datos a actualizar, y esto se deberia hacer en forma simultanea, cosa que raras veces es posible.

Esto ha llevado a la computacion a evolucionar hasta la bases de datos:



## V.- BASES DE DATOS.

Una base de datos, al igual que un archivo, constituye una representación de un subconjunto del espacio de la información.

Se ha definido de muchas formas lo que es una base de datos, sin embargo, es claro que el concepto de base de datos es más sofisticado que el concepto de archivo.

Veamos entonces algunas diferencias notables entre estos dos sistemas:

- En una base de datos conviven diferentes tipos de registros con múltiples relaciones entre ellos.
- Una base de datos exige un soporte que le permita acceso directo y gran capacidad de almacenamiento, esto es, discos magnéticos. A diferencia de los archivos, una base de datos no se puede soportar en cinta magnética.
- En una base de datos se centralizan todos los registros que se mantenían diversificados en diferentes archivos, y por supuesto, solo se utiliza una sola copia de los datos repetidos y se establecen relaciones con el resto de los registros.
- Se admite el acceso concurrente (es decir, simultáneo) de distintos programas que pueden trabajar sobre parte o todos los datos de la base.
- Al no haber duplicación de datos, se evita la redundancia y la inconsistencia, ya que la actualización de un registro se hace en un solo lugar y queda automáticamente disponible para cualquier otro programa.

Veamos un cuadro comparativo, que resume estos aspectos:

CUADRO COMPARATIVO ARCHIVOS & BASES DE DATOS

=====

ARCHIVOS

BASES DE DATOS

Distintos programas mantienen distintos archivos.

Un conjunto central de datos para **todos** los programas.

Informacion duplicada (redundante)

No hay informacion redundante, se elimina espacio, esfuerzo y riesgo.

Un proceso de informacion puede exigir varias fuentes de datos (archivos).

Una sola fuente de datos para todos los procesos de informacion.

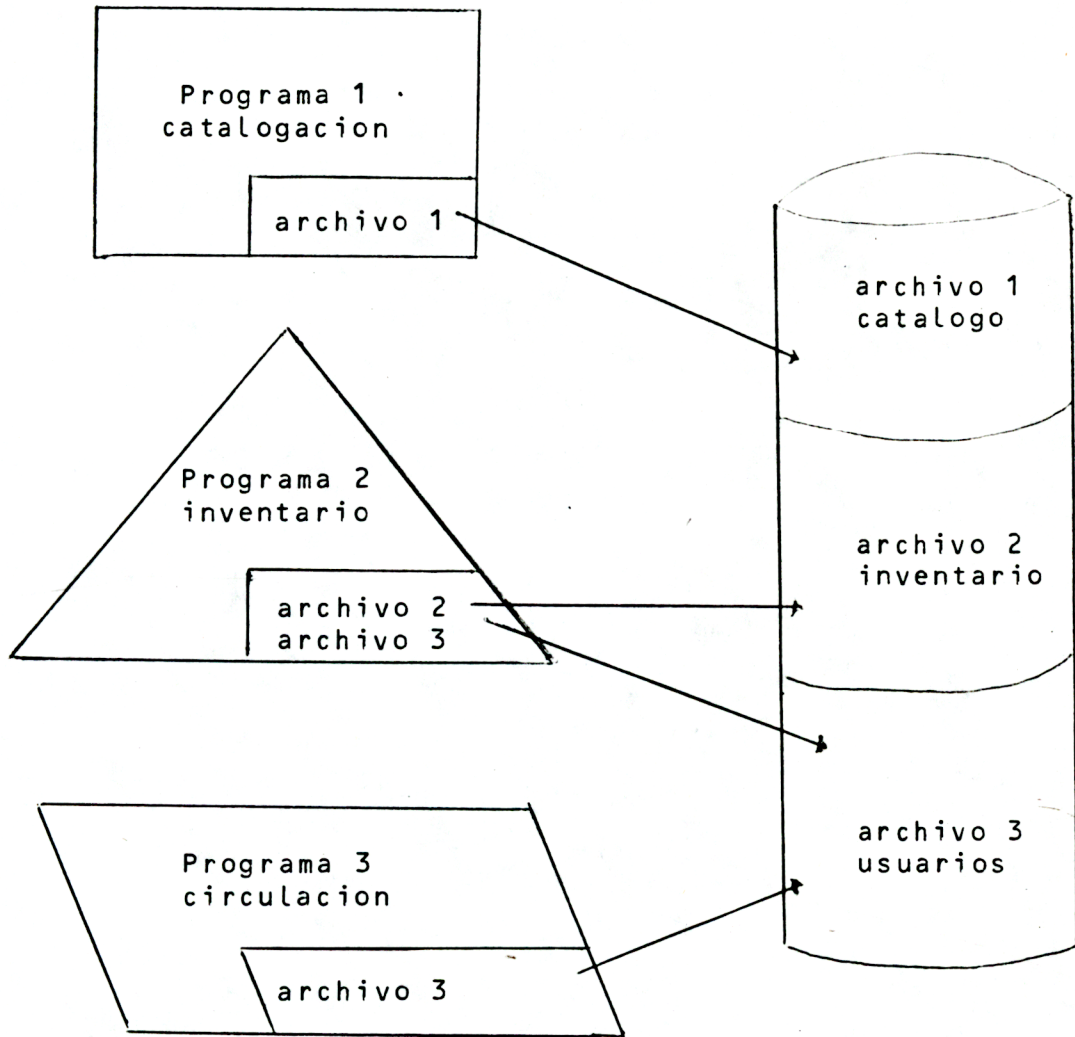
Multiples actualizaciones pueden causar inconsistencia.

Hay consistencia, cada actualizacion se realiza una sola vez y en un solo lugar.

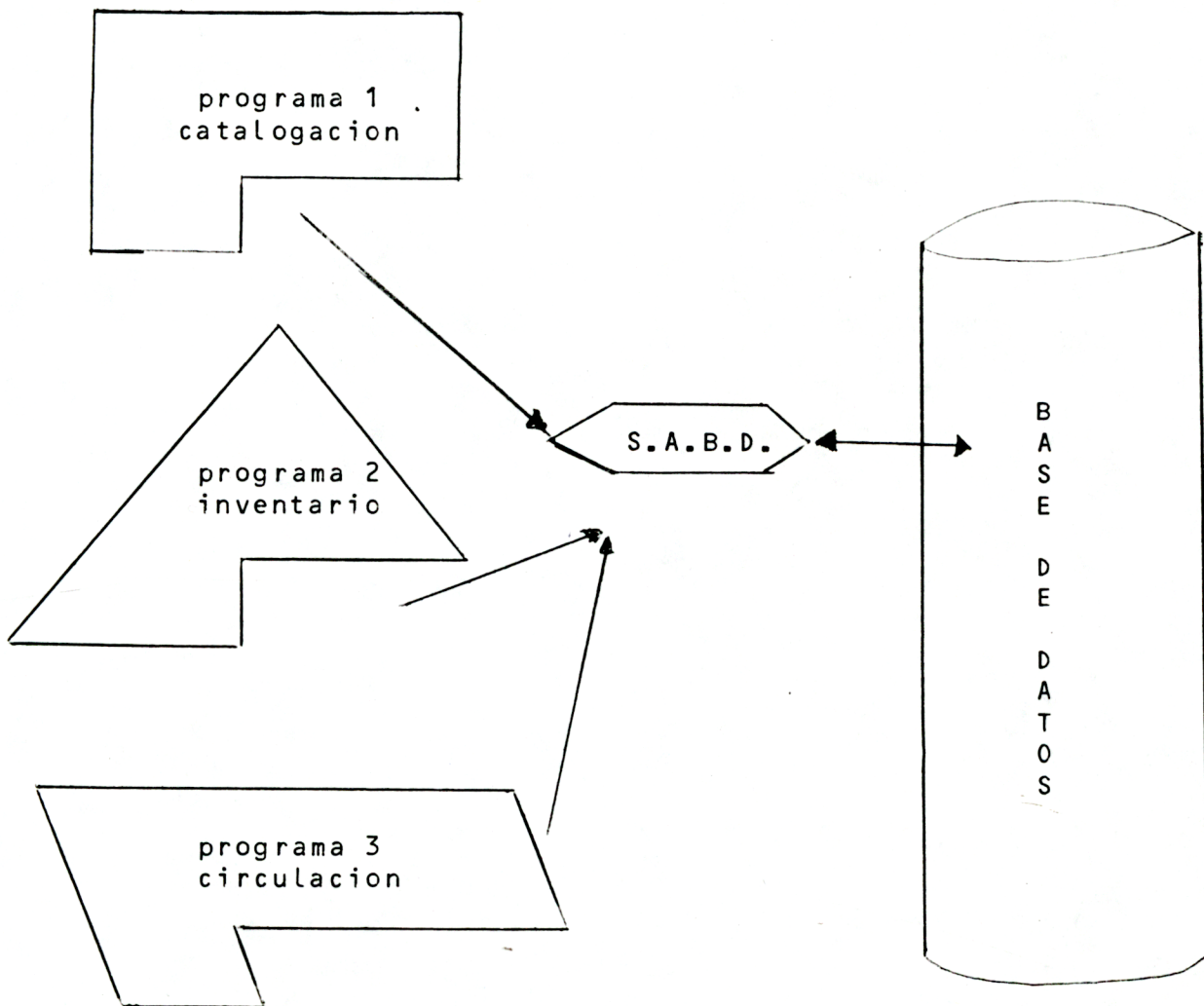
Los programas son funcion de la organizacion de los archivos (si cambia la organizacion hay que cambiar el programa)

Los programas se refieren a los datos requeridos, y no a la forma en que estan; es decir, hay **independencia** entre los datos y los programas.

ENFOQUE TRADICIONAL



## USANDO BASES DE DATOS



Solo el sistema de administracion de bases de datos (S.A.B.D.) conoce la forma en que estan almacenados los datos, los programas interactuan con el S.A.B.D. sin tocar la base de datos.

## SISTEMAS DE ADMINISTRACION DE BASES DE DATOS

Un sistema de administracion de bases de datos (S.A.B.D.o D.B.M.S) permite a multiples usuarios **independientes**, tener acceso concurrente a un receptaculo central de datos o base de datos.

Los objetivos de un S.A.B.D. son:

- Hacer disponible un conjunto integrado de datos a una gran variedad de usuarios, ofreciendo diferentes grados de privacidad de estos datos en caso de ser necesario.
- Proveer integridad a los datos.
- Permitir un control centralizado de la base de datos.
- Permitir independendencia de los datos respecto de los programas que los manejan.

## VI.- SISTEMAS DE INFORMACION

=====

Como su nombre lo indica, esperamos que un sistema de informacion sea un **sistema**, y que este sistema nos permita "manejar" **informacion**.

Hemos visto, al comienzo, lo que es la informacion, por lo tanto, como es de esperar, un sistema de informacion tendra algo que ver con esto.

Existen muchas definiciones de sistema de informacion. Aqui lo estudiaremos como una parte organizada de la realidad, o un **modelo** de la realidad.

En terminos intuitivos, podemos definir un sistema de informacion como una abstraccion de una parte del mundo real que cumple con lo siguiente:

- Los componentes de el sistema se pueden especificar claramente, es decir las partes del mundo real que logicamente pertenecen al sistema.
- Los limites del sistema dentro del mundo real son distinguibles.
- Que entra y que sale, es decir, los intercambios entre el sistema y el resto del mundo.
- Los recursos y procesos necesarios para transformar lo que entra en lo que sale.
- Los objetivos del sistema.
- Las "reglas del juego" dentro de este sistema.

Hemos mencionado que un sistema es un modelo; veamos entonces, para clarificar la idea de un sistema de informacion, lo que es un modelo.

## MODELO.

Se podría decir que un modelo es una "imagen abstracta de la realidad".

Con esto podemos decir de inmediato, que un modelo (cualquiera que sea) no abarca toda la realidad, sino solo una parte, y corresponde, como máximo, a lo que podemos **percibir** de esta realidad.

Estamos conscientes de que nuestra percepción del mundo real es muy limitada, por lo que los modelos son también limitados como representación de esta realidad, y nos vemos obligados muchas veces a corregirlos y hasta a desecharlos, a medida que perfeccionamos herramientas que nos permiten mejorar nuestra percepción.

Como un modelo es una **imagen** abstraída de la realidad, también esta imagen puede estar distorsionada por diversos problemas que ocurren en el proceso de abstracción; este proceso exige un **comprensión** de la realidad percibida, y esta comprensión puede ser errada o muy parcial.

De lo anterior se desprende que existen muchos niveles de modelos, desde un modelo a partir de la realidad percibida, hasta lo que llamaremos un **modelo de datos**.

Si examinamos un poco nuestra propia percepción de la realidad, nos damos cuenta de que inconscientemente o conscientemente, "filtramos" el conjunto de elementos que estamos percibiendo. Así, filtramos los ruidos ambientales, la visión periférica, olores, etc., restringiendo nuestra "visión" de la realidad a determinados aspectos que nos interesan. Este efecto de filtro aumenta a medida que nos **concentramos**.

Este proceso, posterior a la percepción, constituye una **selección** de lo percibido, con el cual creamos un modelo de un nivel diferente, al cual llamaremos **descripción informal** de la realidad.

Nuevamente, para poder manejar adecuadamente la realidad, debemos filtrarla, en una segunda etapa. Hasta el momento, hemos seleccionado un conjunto de percepciones, pero necesitamos relacionar estas percepciones con percepciones anteriores, y notar similitudes y semejanzas entre ellas. En otras palabras, introduciremos un poco de formalidad a nuestra descripción informal primitiva de la realidad. Como resultado de este proceso, tendremos un **universo de discurso no estructurado**.

El siguiente paso en la modelacion de la realidad es estructurar nuestro universo de discurso. En este proceso clasificaremos los distintos elementos de el universo de discurso, es decir representaciones abstractas que nos hemos dado luego de agrupar percepciones que nos parecen semejantes.

En este proceso, vamos a tomar determinadas convenciones para **nombrar** estos elementos, los clasificaremos de diferentes formas y estructuraremos nuestro universo de discurso, hasta obtener lo que llamaremos un **modelo conceptual** de la realidad.

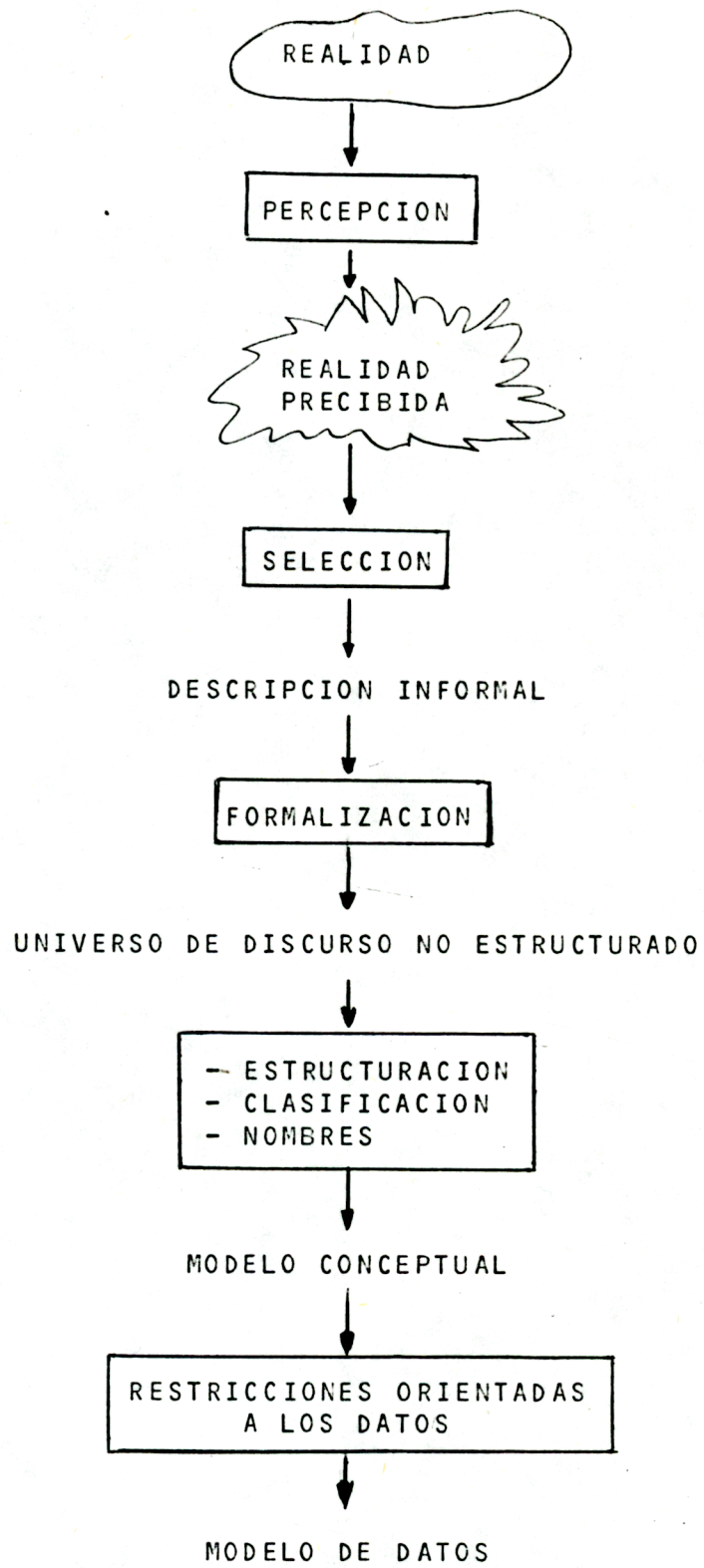
En este punto, nuestro modelo ya no tiene relacion directa con nuestra percepcion del mundo real, hemos conceptualizado esta percepcion, y hemos creado un mundo imaginario mucho mas simple que el mundo real, pero **organizado** (el mundo real es totalmente desorganizado). Esto nos permite "predecir" con alguna exactitud (muy poca en general) el comportamiento del mundo real, y actuar de acuerdo con ello. Sin este proceso podriamos caer facilmente en la paranoia.

Finalmente, tomando en cuenta la definicion de informacion que vimos en la primera parte, podriamos introducir restricciones a este modelo conceptualizado en la tres coordenadas del espacio de informacion. Podremos entonces, limitar tanto las clases de objetos, como los atributos y valores que permitiremos en nuestro modelo, y podremos fijar determinadas representaciones para estos elementos de informacion aceptables. Diremos entonces que tenemos un **modelo de datos**.

En cada uno de estos niveles de modelos, esperamos que exista una correspondencia entre las construcciones abstractas de nuestro modelo y el mundo real. Si descubrimos que tal correspondencia no es adecuada, debemos perfeccionar el modelo. Esto es sin duda bastante subjetivo, aunque existen metodos mas o menos precisos que sirven como herramienta para perfeccionar modelos. Uno de estos metodos que es importante mencionar es la **simulacion**.

Al simular un modelo, nosotros vamos a observar el comportamiento de nuestro modelo en determinadas situaciones, y compararemos los resultados con observaciones de la realidad.

DIFERENTES NIVELES DE MODELOS



## MODELOS EN UN SISTEMA DE INFORMACION

Al modelar la realidad para un sistema de informacion, debemos llegar en primer lugar a un modelo conceptual.

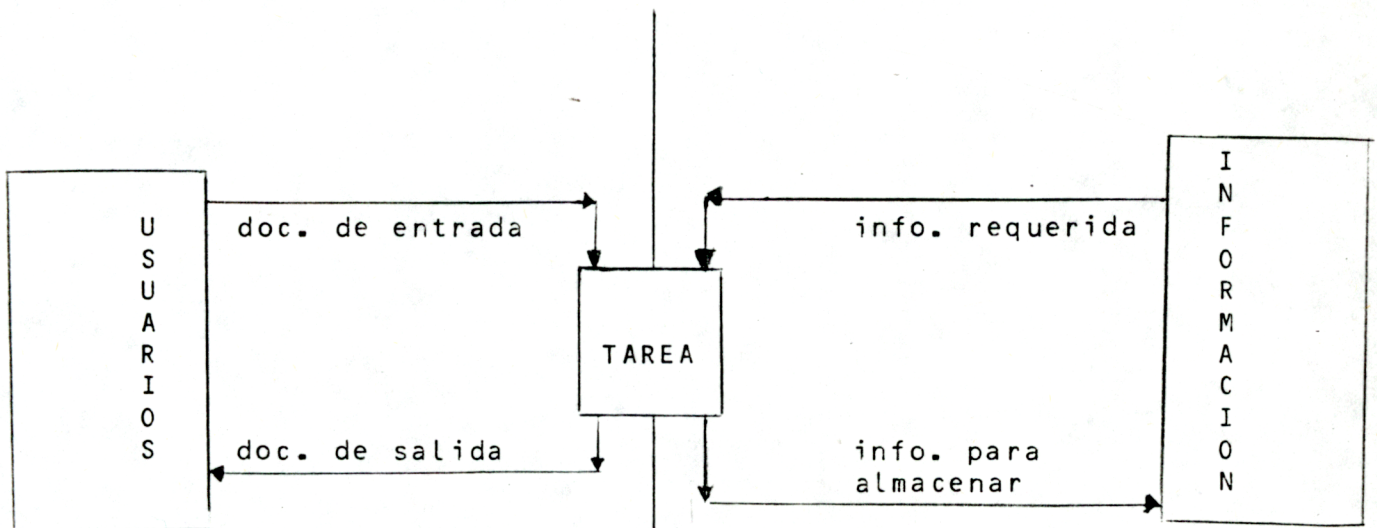
En este modelo debemos distinguir cuatro elementos importantes, que son:

- Los Usuarios
- Las Tareas
- La Informacion almacenada en el sistema.
- Los flujos de informacion entre las tareas, los usuarios y el sistema.

Una tarea es un proceso que recibe y/o produce la informacion intercambiada entre los usuarios y el sistema.

Una tarea puede ser manual o automatizada, pero es repetitiva para cada intercambio (usamos la palabra "tarea" para no confundirla con el concepto de proceso, que es mucho mas amplio).

La tarea va a transformar la entrada en salida de acuerdo a reglas claras y pre-definidas.



## ¿QUE ES UN FORMATO?

=====

Cuando un procedimiento bibliotecario quiere ser automatizado, es necesario proveer al computador con:

- a) Los datos o información a ser procesados
- b) Una lista de instrucciones de como procesar los datos

Las instrucciones para procesar los datos serán hechos por el programador.

El bibliotecario participará en la decisión sobre que información se va a procesar y como se va a hacer.

Cuando hablamos del "como", cabe hablar de "formato".

Formato es un método de organizar datos para que un registro o parte de él pueda ser identificable. El formato contiene:

- Datos : Es la unidad más baja de información. Ej.: ISBN
  - Campo : Colección de datos. Ej.: Autor = nombre, título nobiliario, fecha de nacimiento. Un campo puede ser un sólo dato. Ej.: ISBN
- Registro - colección de campos.

### Ejemplos:

La Biblioteca quiere hacer un archivo automatizado de sus usuarios. El bibliotecario decide incluir nombre, número y comuna de cada lector

Solución 1	—	formato fijo
Nº	—	6 caracteres
Nombre	—	20 caracteres
Comuna	—	16 caracteres

```
4591XXPEREZXMARIAXXXXXXXXXXMAIPUXXXXXXXXXXXXX
65478XGALLEGUILLOSXMANUELXSANXBERNARDOXXXXX
34XXXXLOPEZXDEXSANTAXMARIAXLASXCONDESXXXXX
```

En el primer y segundo caso hay espacios sobrantes. En el tercer caso el registro ha sido truncado. A pesar de que es difícil estimar el máximo o minimum de caracteres para cada campo es más fácil para el computador operar con formato fijo.

Solución 2 - Formato variable  
Un caracter define el inicio o final de cada campo.

Ejemplo:

- @ Iniciar número de lector
- \$ Iniciar nombre de lector
- \* Iniciar comuna

@ 4591\$PEREZXMARIA\*MAIPU  
@ 65478\$GALLEGUILLOSXMANUEL\*SANXBERNARDO  
@ 34\$LOPEZXDEXSANTAXMARIAXJOSE\*LASXCONDES

Esta es una solución más atractiva desde el punto de vista bibliotecológico, pero requiere un programa más complicado.

Solución N° 3 - Formato con directorio.

La posición del caracter de inicio de cada campo, esta especificada en el directorio al comienzo del registro.

La posición del caracter de inicio de cada campo esta especificada en el directorio al comienzo del registro.

1	5	16
---	---	----

451PEREZXMARIAMAIPU

1	6	25
---	---	----

65478GALLEGUILLOSXMANUELSANXBERNARDO

Esta es una mezcla de campos fijos y variables. Este tipo de formato ha sido adoptado por varias bibliotecas

M A R C (Machine Readable Cataloguing)  
=====

En la actualidad con el desarrollo que ha alcanzado el MARC el nombre es anacrónico, debería llamarse "Machine Readable bibliographic description".

Se señala como nacimiento del MARC a un informe sobre automatización en la Biblioteca del Congreso en USA hecho por G.W. King y otro en 1963. (Automation and the Library of Congress : a survey sponsored by the Council of Library Resources, Library of Congress Washington. 1963)

La conclusión más importante de este informe fue que el sistema bibliográfico de la L.C. podría ser automatizado dentro de 10 años.

En 1965 se preparó un informe preliminar sobre los problemas de conversión. Como resultado el Council of Library Resources dió un grant para un plan piloto de distribución de cintas magnéticas en formato MARC.

A fines de 1966 se repartían semanalmente registros en cinta magnética a 16 Bibliotecas de todos los tipos. 50.000 registros de monografías en inglés fueron distribuidos a las Bibliotecas participantes entre Noviembre 1966 y Junio 1968, fecha en que finalizó el proyecto piloto.

El formato de esta etapa, conocido como MARC I estaba estrechamente ligado a la estructura del catálogo en fichas de la L.C. Hay una diferencia significativa entre un registro de catálogo en cinta magnética y un archivo bibliográfico automatizado.

En 1968 se origina el formato MARC II

La decisión de reconsiderar el formato MARC I coincidió con la decisión de OSTI (U.K. office of Scientific and Technical Information) de financiar un estudio de factibilidad para un registro en el U.K. ligible por la máquina.

Así fue que hubo una estrecha colaboración entre BNB y la LC en el diseño del nuevo formato conocido por MARC II y hoy más simplemente como MARC.

La esencia del MARC II es que un sistema de comunicación o un lenguaje común de intercambio entre sistemas que desean intercambiar información bibliográfica.

El sistema MARC está basado en los siguientes principios:

- 1.- Ser un sistema de intercambio internacional
- 2.- Ofrecer flexibilidad de uso
- 3.- Ser compatible con una gran variedad de sistemas de computo
- 4.- Ofrecer un eficiente factor de búsqueda por dato
- 5.- Aceptar standares donde existen y adoptar nuevos standares cuando se creen.

1969 se inició la distribución de registros en el formato MARC II.

1970 se llevó a cabo un interesante proyecto en microbibliografía. Las cintas MARC estadounidenses y británicas se juntaron y se extrajeron los registros en inglés. Esto se hizo en COM el que fue luego microfilmado usando una técnica llamada Photo-Chronic Micro Image (PCMI).

El resultado fue una transparencia de 6 x 4 pulgadas, conteniendo 2.380 cuadros que podrán ser leídos con un lector especial. El experimento fue un éxito y desde 1972 es una publicación conocida como Books in English. Es la más grande bibliografía actual en inglés.

Se le agregan alrededor de 80.000 registros anualmente.

Prontamente las Bibliotecas usuarias de las cintas MARC se dieron cuenta de la necesidad de contar con registros MARC para toda la colección antes de producir un catálogo automatizado completo. Así se iniciaron los proyectos de conversión retrospectivos.

En 1970 se inició en USA el proyecto RECON (Retrospective Conversion)

Se basó en la L.C. pero incluyó numerosas otras Bibliotecas en U.S.A. En U.K. se ha hecho un archivo MARC retrospectivo que incluye desde 1950.

1973 Recomendación a OEA para tener MARC en español.

1976 Edición preliminar MARCAL

1978 Inicio implementación MARCAL.

En la actualidad el MARC está implementado en México. Se está implementando en Venezuela a través de un convenio con una universidad americana. (North Western de Chicago), y se está implementando en Chile.

Como decíamos el objetivo del formato MARC es la comunicación de información bibliográfica entre Bibliotecas que usaran esta información de diversas maneras. Por esto, muchos de los detalles del registro no son necesarios y para algunas Bibliotecas que deben reformatear el registro para su uso local. Este hecho ha sido criticado por muchas Bibliotecas. El tamaño promedio del registro U.K. MARC es de 780 caracteres.

A pesar de que el formato MARC fue diseñado para ser internacional a medida que más países lo usan mayores variaciones se le han introducido. Para solucionar este problema y permitir el intercambio a nivel internacional se ha diseñado un nuevo formato conocido con el nombre de UNIMARC.

Las organizaciones nacionales de los diferentes países producirán registros de acuerdo a los estándares nacionales y luego lo reformatearan al UNIMARC para intercambio. Nombres previos al UNIMARC, fueron SUPERMAC e INTERMARC.

El formato MARC está hecho de acuerdo con la norma ISO 2709 de intercambio de información bibliográfica en cintas magnéticas. También se han incorporado las reglas Anglocamericanas de catalogación ISBD(M) International Standard Bibliographic Description.

¿Como es el formato MARC?  
=====

En el formato MARC los campos están identi  
ficados por las etiquetas (tags). Etiqueta es un código de  
3 caracteres que siempre son números y van del 000-999.

El registro MARC consiste básicamente en 4  
partes.

1.- Líder, guía o cabecera

Es fijo de 24 caracteres y proporciona información sobre  
las características del registro.

- Longitud total del registro (Nº total de caracteres)

- Estado del registro

n = nuevo

c = registro corregido o modificado

d = registro eliminado

a = completado previamente un registro parcial

- Tipo de material

a = material impreso etc.

2.- Directorio

Esta compuesto de una serie de entradas de longitud fija  
12 caracteres cada uno, que incluyen:

Etiqueta: 3 caracteres

Longitud  
del campo: 4 caracteres

Posición del  
caracter inicial: 5 caracteres

Ejemplo: La etiqueta del campo de autor personal en el  
formato MARC es 100

100002600137

100 autor personal

26 longitud del nombre

137 caracter de inicio del nombre en el registro



1 9 6 6 c 1 9 6 5

c 1 9 6 6 1 9 6 5

s.a.

n ∅ ∅ ∅ ∅ ∅ ∅ ∅ ∅

1 9 6 6 c 1 9 5 0 Publicada en Dublin 1741

r 1 9 6 6 1 7 4 1

1 9 6 6 - 1 9 6 7 v. 1, 1 9 6 7

m 1 9 6 6 1 9 6 7

1 9 6

q 1 9 6 0 1 9 6 9

↑  
Se pone el inicio de la década

1 9

q 1 9 0 0 1 9 8 0 ← Año de catalogación

- Código de país de publicación (3 caracteres)

Se hace de acuerdo a las normas ISD (pag. 7)

- Códigos de ilustración (4 caracteres alfabéticos)

Hay una lista larga

a = ilustraciones

b = mapas

c = fotografías

d = cartas marítimas

e = planos

...

...

...

...

∅ = sin ilustraciones

Cuando hay más de 4 términos aplicables la secuencia de la lista determina los 4 que deben registrarse

- Código de nivel intelectual (1 caracter)
- Código de forma de reproducción (1 caracter)
- Código de forma de contenido (4 caracteres)

Se usa principalmente para las obras de referencia.

Puesto que algunas obras incorporan varios tipos de materiales, se pueden especificar hasta 4 tipos.

- b - bibliografías
- c - catálogos
- i - Indices
- a - resúmenes analíticos
- d - diccionarios
- e - enciclopedias
- r - directorios
- s - estadísticas
- h - manuales
- p - libros de texto programado
- ∅ - no es ninguna de las formas especificadas

- Indicadores de publicaciones gubernamentales
- Indicadores de Congreso o asamblea
- Indicadores de Homenaje

Un código de un caracter muestra cuando la obra es un homenaje

∅ = No es un homenaje

1 = Es un homenaje

- Código de idioma (3 caracteres)
- Código de origen de la catalogación
  - a = LC y BNA (NAL)
  - b = LC Y NLM
  - c = LC en cooperación con otra Biblioteca

4.- Campos Variables Contienen toda la información bibliográfica del registro.

Fuera de la etiqueta los campos variables tienen:

- a) Dos indicadores numéricos que indican características del campo. El significado está estrechamente ligado a la etiqueta, con lo cual varía de campo a campo.
- b) Delimitador general es el signo \$ , es un interruptor para que el computador reconozca los diferentes elementos de un campo.
- c) Código de sub-campo, va junto al delimitador e indica uno de los elementos del campo

Ejemplo:

Pie de imprenta - consta de -

Etiqueta : 260

Indicadores : 1<sup>er</sup> indicador 0 - editorial no es la entrada principal

1<sup>er</sup> indicador 1 - editorial si es la entrada principal.

2<sup>a</sup> indicador MARCHAL está en blanco

U.K. MARC

0 pie de imprenta relacionado con el trabajo

1 pie de imprenta relacionado a 1 volumen o parte del trabajo.

Sub-Campos

\$a lugar de publicación

\$b nombre de la editorial

\$c fecha de publicación

MARCAL \$d lugar de impresión

\$e nombre de la imprenta

U.K. MARC full address information

Se puede decir por lo tanto que la estructura del formato MARC se puede dividir en dos secciones:

La primera sección consta del líder, el directorio y los campos fijos de control. Esta sección contiene información que describe la información bibliográfica contenida en la 2<sup>da</sup> sección de los campos variables que es la manipulable por el computador.

LIDER	DIRECTORIO	CAMPOS FIJOS	CAMPOS VARIABLES
-------	------------	--------------	------------------

CAMPOS VARIABLES

INDICADORES	CODIGO DE SUB-CAMPO	DATO 1	CODIGO DE SUB-CAMPO
-------------	---------------------	--------	---------------------

DATO 2	CODIGO DE SUB-CAMPO	DATO 3	FIN DEL CAMPO
--------	---------------------	--------	---------------

La etiqueta de cada campo variable se incluye en el directorio del registro y no en el campo que lo identifica.

## PROGRAMAS COMPUTACIONALES DISPONIBLES

E. Alejandro Burgos M.

En esta sesion examinaremos los programas que estan disponibles, en este momento, (Octubre de 1981) en el Departamento de Catalogacion. Los siguientes son los diferentes programas:

- \* un programa de busqueda de registros MARC en base de datos
- \* un programa de inventario de libros
- \* un programa para catalogar segun el formato MARC
- \* un conjunto de programas de impresion de registros MARC

### PROGRAMA PARA LA BUSQUEDA DE REGISTROS MARC EN BASE DE DATOS

El programa permite consultar la base de datos para saber si el libro que se va a catalogar esta registrado o no. La busqueda se efectua por:

- autor
- titulo
- autor-titulo
- clasificacion
- numero MARC

El programa permite copiar un registro MARC desde la base de datos a un archivo temporal. Este registro puede ser editado mediante una funcion del programa de catalogacion ( esto permite aprovechar ese registro para catalogar otro libro: por ejemplo otra edicion del mismo titulo u otro titulo con un contenido similar).

Si la busqueda no tiene exito, entonces el programa entrega un numero MARC nuevo, si la busqueda es exitosa entonces puede suceder una de las siguientes cosas:

- el texto que se desea catalogar es una copia del registro que se encontro, entonces hay que anotar el numero MARC del registro encontrado.
- si el texto que se desea catalogar no es una copia exacta del registro que se encontro (es otra edicion o es otro titulo con un contenido intelectual semejante), pero se desea aprovechar ese registro para hacer la catalogacion, entonces hay que anotar el numero MARC del registro que nos interesa y copiarlo a un archivo mediante una de las funciones de este programa (volviendo al menu principal).

Por todo lo anteriormente dicho, queda claro que el numero MARC es indispensable para ingresar al programa de inventario, y debe ser obtenido mediante el programa de busqueda.

#### PROGRAMA DE INVENTARIO

La funcion del programa de inventario es consignar informacion acerca del item fisico que se va a catalogar.

El programa interroga al usuario que lo ejecuta y le pide la siguiente informacion:

- numero MARC
- numero de volumen fisico
- biblioteca
- unidad (propietario)

El numero de copia es asignado por el programa y es correlativo para todo el sistema, por cuanto este inventario es global y no por unidad o biblioteca.

La informacion recogida por este programa se utilizara en las siguientes actividades:

- subsistema de circulacion automatizada
- toma de inventarios
- impresion de marbetes y etiquetas para cada libro

## PROGRAMA DE CATALOGACION

Basicamente puede decirse que la importancia de este programa reside en que permite trabajar con registros MARC. Esto significa, en este momento, las siguientes tres cosas:

- que se pueden crear registros MARC, mediante una funcion del programa que permite invocar los campos y subcampos del formato MARC-UC, hasta conseguir la descripcion completa de un libro.
- que se puede editar un registro MARC para revisarlo, corregirlo y finalmente grabarlo en un archivo (este archivo sera supervisado y luego los registros ingresaran a la base de datos).
- que se puede leer un registro MARC ya creado (tomandolo desde un archivo creado por el programa de busqueda, por ejemplo) y usarlo como base para catalogar otro libro similar u otra edicion del mismo libro.

## FILOSOFIA DEL PROGRAMA DE CATALOGACION

Es un programa de ayuda para el catalogador; a continuacion veremos el alcance y la forma de esta ayuda:

- no es necesario conocer el formato MARC-UC de memoria o depender del Manual de Codificacion,
- permite comenzar a digitar un registro por el campo que el catalogador desee,
- avisa si el catalogador olvido digitar algun campo o subcampo obligatorio, o algun campo de entre una serie de campos alternativos,
- impide duplicar informacion que no es repetible (dejando la decision final al catalogador, este rasgo es interesante de destacar, por cuanto es caracteristico de todo el programa: existe una serie de restricciones dentro de la codificacion de un registro, sin embargo estas restricciones pueden ser forzadas si el catalogador asi lo desea),
- permite revisar y corregir el registro digitado antes de grabarlo.

Lo interesante de analizar es el como brinda esta ayuda el programa. Lo hace a traves de 3 instancias:

- help (invocado mediante el comando hel;)

provee informacion sobre:

\* informacion general

\* teclas de funcion

\* codigos especiales

- mensajes sobre algunas restricciones del programa, por ej.:

\* falta informacion

\* informacion no repetible

\* codigos equivocados

- menus o recordatorios sobre:

\* funciones del programa

\* campos del formato MARC-UC

\* subcampos de cada campo del formato

\* help

Todas estas ayudas estan analizadas con mas detalle en el llamado Manual de Digitacion.

## PROGRAMAS DE IMPRESION

Existen diferentes programas de impresion que ayudan a complementar el trabajo efectuado en el Departamento de Catalogacion, los que ahora nos interesan son los siguientes:

- programa de impresion de registros MARC en una forma adecuada para una posible supervision,
- programa de impresion del set de fichas (puntuacion ISBD, entre campos y subcampos, en forma automatica),
- programa de impresion de marbetes y etiquetas para el material de cada libro del sistema.

\* \* \*