

Conceptos Básicos de la Computación

Conceptos Básicos

■ Computadora:

- Dispositivo electrónico capaz de interpretar y ejecutar comandos programados para entrada, salida, cálculo y operaciones lógicas.

■ Informática (Ciencias de la Computación):

- Disciplina que busca establecer una base científica para diversos temas, como el diseño de computadores, la programación de computadores, el proceso de información, la elaboración de algoritmos para resolver problemas y el proceso algorítmico en sí.

■ Sistema Computacional:

- Sistema que se basa en la computadora. Un sistema sólo tiene cuatro componentes fundamentales: *entrada*, *procesamiento*, *salida* y *almacenamiento*.
- La computadora es el sistema de *procesamiento* y se combina con los otros tres componentes para formar el sistema computacional.

Entrada



Salida



Procesamiento



Almacenamiento



■ Hardware (HW):

- Cualquier parte de la computadora que puede ser tocado.
- Consiste en dispositivos electrónicos interconectados.

■ Software (SW):

- Conjunto de instrucciones electrónicas que dicen al HW lo que debe hacer. Conocidos también como **programas**, cada uno con un propósito específico.

■ Datos:

- Es la expresión simbólica de elementos que no tiene un significado por sí mismo.
- Son la materia prima de la que se deriva la información.

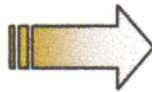
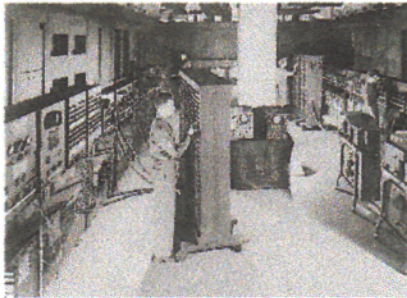
■ Información:

- Conjunto de datos que por medio de convenciones previamente conocidas tienen un significado.

■ Procesamiento:

- Cualquier operación que se lleve a cabo sobre los datos o sobre la información para obtener datos o información.

Historia de la Computación

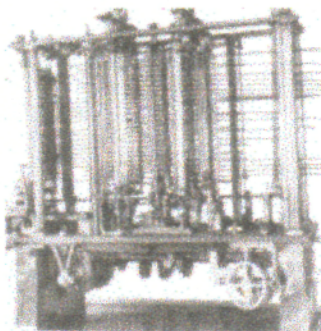


■ El ábaco (hace 5000 años)

- Primer dispositivo mecánico de contabilidad.

■ Pascalina (1642)

- Inventor: Blaise Pascal (1623-1662).
- Sumadora para calcular impuestos. (Ayudar a su padre).
- Funcionaba a base engranes y ruedas.
- los datos se representaban mediante las posiciones de los engranajes, y los datos se introducían manualmente estableciendo dichas posiciones finales de las ruedas.



■ Telar de Jacquard (1801)

- Inventor: Joseph-Marie Jacquard.
- Telar que podía reproducir automáticamente patrones de tejidos leyendo la información codificada en patrones de agujeros perforados en tarjetas de papel rígido.
- Tarjetas perforadas dirigían aguja, hilo y tela.

■ Máquina de Babbage (1834)

- Inventor: Charles Babbage
- Máquina de diferencias (calculaba tablas matemáticas) -> Máquina analítica (computadora de propósito general).
- Podía +, -, * y / en secuencia automática a una velocidad de 60 sumas por minuto.
- Tenía un tamaño de una cancha de fútbol.
- En 1843 Lady Ada Augusta Lovelace (primera programadora) sugirió la idea de que las tarjetas perforadas pudieran adaptarse de manera que propiciaran que el motor de Babbage repitieran ciertas operaciones.

■ Tabuladora de tarjetas perforadas Hollerith (1890)

- Inventor: Herman Hollerith.
- Sumadora de datos introducidos manualmente en tarjetas perforadas.
- Utilizada por la Oficina de Censos de los EEUU (Ahorró \$5 000 000.).

■ ABC (1942) Atanasoff-Berry Computer

- Inventor: John V. Atanasoff-Berry
- Primera computadora digital electrónica.
- Realizaba cálculo de física.
- Utilizaba tubos al vacío.

■ Mark I (1944)

- Inventor: Howard Aiken.
- 1era. Computadora electromecánica.
- 17m Longitud x 2,5m altura



1946 - ENIAC (Univ de Pennsylvania)

■ ENIAC (1946) Electronical Numerical Integrator and Computer

- Inventores: John Mauchly, Presper Eckert.
- Fue desarrollada con el propósito de calcular tablas de trayectoria para el ejército de los EEUU.
- Mil veces más rápida que sus antecesoras. 450m2. 30ton. 18000 tubos

Máquina	Inventor	Año	Propósito
UNIVAC I	Mauchly Eckert	1951	Censo EEUU. Tubos al vacío. Comp.Dig. Comercial.
Honeywell	Honeywell	1959-1964	Transistores
System 360	IBM	1964	Circuitos Integrados
PC	IBM	1981	Miniaturización de circuitaría – Diseños de HW y SW.

I Generación:
Tubos al vacío.

II Generación:
Transistores.

III Generación:
Circuitos Integrados.

IV Generación:
Microprocesador.

Hardware



1. Unidad Central de Procesamiento (CPU)

Consta de los circuitos electrónicos capaces de desarrollar las funciones de control y de cálculo aritmético y lógico (microprocesador).

1. Unidad de Control (UC):

- Es el centro lógico de la computadora. Se puede ver como la policía de tránsito dirigiendo el flujo de datos.

2. Unidad Aritmético-Lógica (ALU):

- Encargada de hacer todas las operaciones aritméticas y lógicas. Es decir, es la encargada de la manipulación de los datos.

3. Registros:

- Celdas de memoria especiales, son utilizadas para el almacenamiento temporal de los datos.
 - Suplen entradas a los circuitos de las ALU
 - Almacenan resultados
 - Guardan datos de aplicación inmediata



■ Ejemplos de CPUs:

- Intel 8086
- Intel 80486
- Pentium III
- Pentium IV
- Motorola 6800
- G5

2. Unidades de Memoria

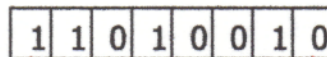
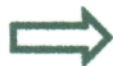
Celdas donde se almacena información. Tienen dirección específica para ser localizada por la unidad central. La capacidad de memoria se mide en **Bytes**.

Bit: Representa un estado electrónico :

- 1- Encendido, True
- 0- Apagado, False

Byte: Unidad de almacenamiento de caracteres.

1 Byte = 8 bits

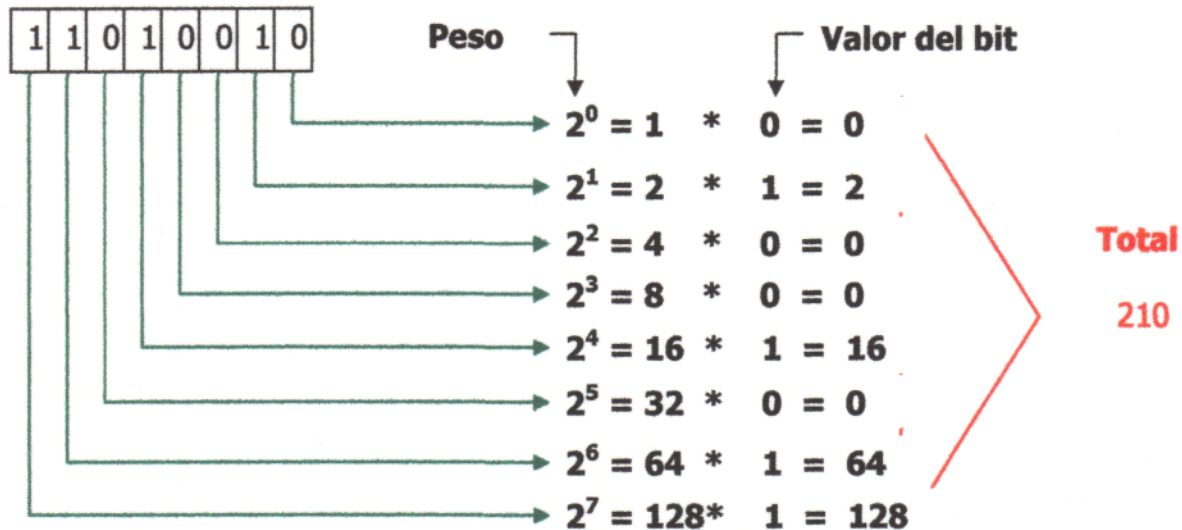
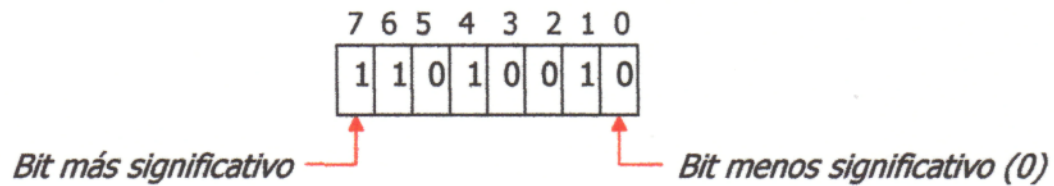


Bit más significativo

Bit menos significativo

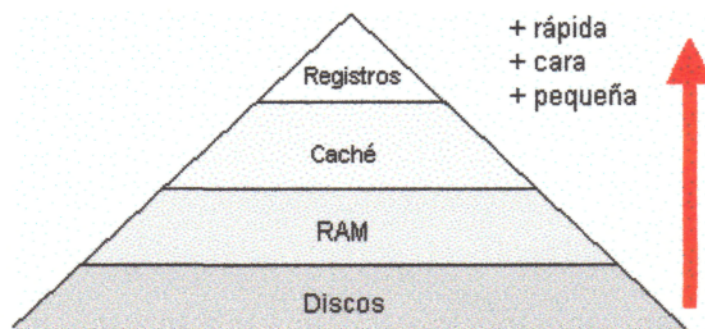
1 KB (kilobyte)	=	1024 B	ó	2^{10}	Bytes
1 MB (megabyte)	=	1024 KB	ó	2^{20}	Bytes
1 GB (gigabyte)	=	1024 MB	ó	2^{30}	Bytes

■ Sistema Binario



■ Tipos de Memoria

1. **RAM (Random Access Memory):** Se pueden leer y grabar datos, es volátil. Todo programa o dato se transfiere a la RAM. 128 MB, 256 MB, 512 MB, etc.
2. **ROM (Read Only Memory):** De sólo lectura. Es no volátil. Su contenido se guarda durante su construcción y no se puede cambiar.
3. **Caché:** En ella se almacenan los datos que se utilizan con mayor frecuencia. Más rápida, más cara, más pequeña que la RAM. Alivia la diferencia de velocidad entre la CPU y la RAM



3. Reloj

Es un oscilador, un circuito electrónico que genera una señal con una frecuencia determinada, a fin de sincronizar otros circuitos.

La velocidad del procesador se mide por su frecuencia de oscilación
ciclos de reloj por segundo.

Número de

Se mide en MHz (Megahertz) Millones de ciclo de reloj.

$$\tau_{\text{ciclo de reloj}} = \frac{1}{\text{Frecuencia}}$$

Entre más pequeño es el ciclo de reloj, más rápido es el procesador.

4. Buses

Líneas trazadas en la tarjeta madre para transportar bits.

Tamaño de palabra: Número de bits que se maneja como unidad en un sistema de computación.

Ejemplo:

32 bits de palabra



El bus puede transmitir 32 bits a la vez entre el procesador, la RAM y los dispositivos periféricos.

■ Tipos de buses

- Bus de Datos
 - Ruta eléctrica que conecta la CPU, la memoria y otros dispositivos de HW en la tarjeta madre.
 - Grupo de cables paralelos, donde cada cable transporta un bit a la vez.
- Bus de Instrucciones
 - Juego de cables similar a al bus de datos que conecta la CPU y la RAM y lleva las direcciones de memoria.
 - El número de cables en este determina la velocidad máxima de direcciones en memoria.

5. Tarjeta Gráfica

Dispositivos que permiten incorporar gráficos en los programas.

En desuso

MDA (*Monochrome Display Adapter - 1980*): modo texto representando 25x80 líneas en pantalla. Con una memoria de 4Kbytes.

CGA (*Color Graphics Adapter - 1981*): capaz de trabajar tanto en modo texto como en modo gráfico. Presentaba 4 colores con una resolución de 320 x 200 píxeles. Equipada con cuatro veces más de memoria, 16 Kbytes.

HGC (*Hércules Graphics Card*): con una resolución de 720x348 píxeles. Combina la legibilidad en modo texto de la MDA con las capacidades gráficas de la CGA. Compuesta de una memoria de 643 Kbytes.(No era capaz de mostrar color).

EGA (*Enhanced Graphics Adapter*): Con una resolución en el modo gráfico de 640x350 píxeles, se podían representar 16 colores diferentes de una paleta de 64 colores, trabajando en modo texto con 25 filas y 80 columnas. También se aumentó la RAM de vídeo hasta 256KB,

Más Utilizados

VGA (*Video Graphics Array*): representar 256 colores a escoger de una paleta de 262.144 tonalidades, con una resolución de 640 x 480 píxeles en modo gráfico y 720x400 píxeles en modo texto. Detecta si el monitor que tiene conectado es color o monocromo, y en caso de que sea monocromo, convierte la información de color en tonos de grises.

SVGA (*Super Video Graphics Array*)

3D: para juegos, películas, imágenes.

6. Dispositivos Periféricos (E/S)

Entrada



Mouse



Teclado



Scanner



Lápiz
Lumínico



Webcam



Bola
Palmar



Joystick



Cámara
Digital

Salida



Monitor



Parlantes

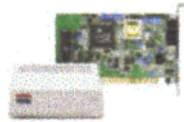


Impresora

Entrada/Salida



Discos



Módem



Monitor Táctil



Software

Es el conjunto de todos los componentes lógicos de un sistema computacional que permiten que el HW funcione.



Persona con problemas

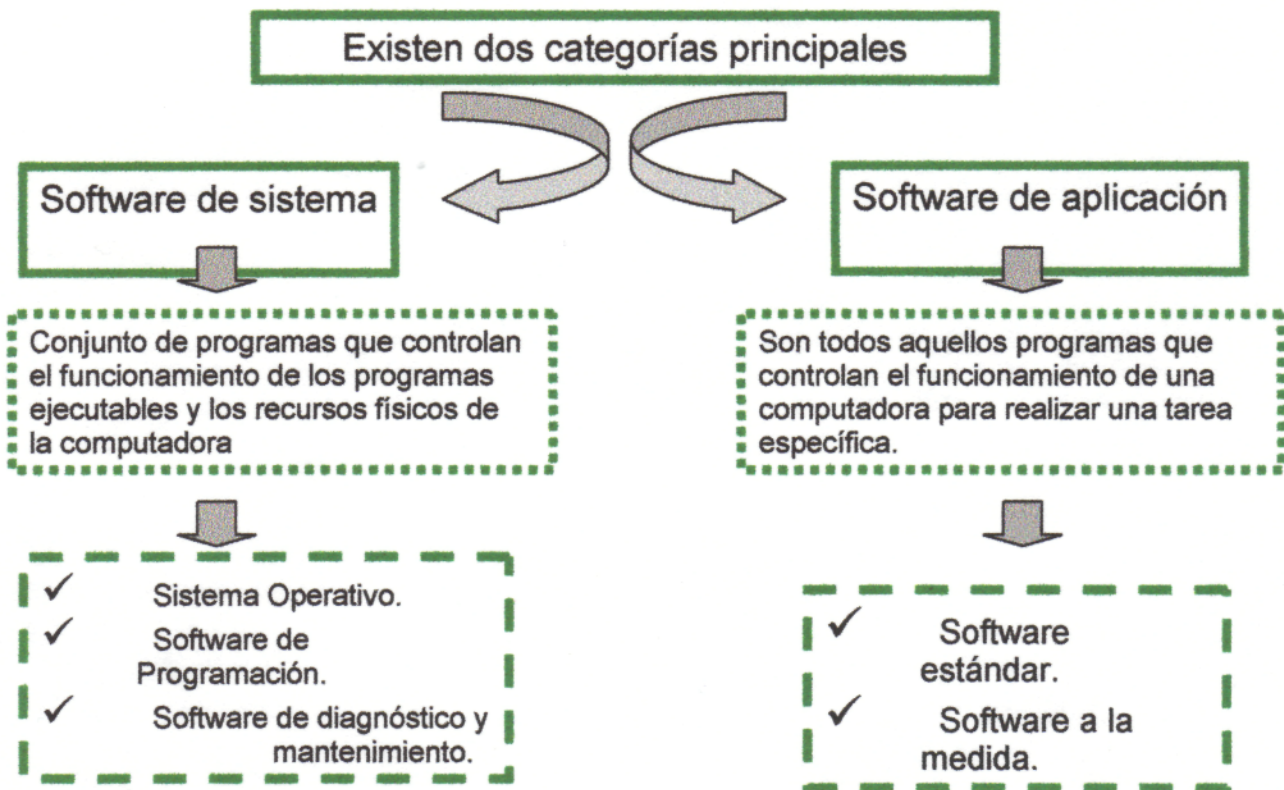


Permite el enlace entre el ser humano y las computadoras. Enlaza, además sistemas con sistemas.



Entiende solamente 1 y 0

TIPOS DE SOFTWARE



Software de sistema

1) El Sistema Operativo

Es el conjunto de programas necesarios para administrar los recursos de la computadora: HW y SW. Además, facilita la comunicación entre el hombre y la computadora evitándole conocer detalles técnicos.

- ✓ Controla los dispositivos periféricos
- ✓ Coordina el procesamiento
- ✓ Administra la memoria
- ✓ Administra los programas
- ✓ Administra el Hardware
- ✓ Da seguridad al sistema de cómputo
- ✓ Da acceso al usuario

1.1) Tipos de sistemas operativos

- **De tiempo real**
 - Sistemas industriales, instrumentos científicos, telecomunicaciones. Para operaciones con tiempo predecible.
- **Monousuarios, tarea única**
 - Palms. Solo un usuario ejecutando una sola tarea a la vez.
- **Monousuarios, tareas múltiples**
 - Windows 98, 2000, XP y MacOS.
- **Multiusuarios (tareas múltiples)**
 - Linux, UNIX (FreeBSD)

1.2) Algunos Sistemas Operativos disponibles

- **MS-DOS** (Microsoft Disk Operating System): se podía cargar un programa a la vez.
- **Windows 95/98**: SO multitareas con interfaz gráfica.
- **Windows NT/2000/XP**: Estaciones de trabajo y servidores(NT).
- **OS/2** (Microsoft / IBM): SO multitarea para microprocesadores INTEL.
- **Macintosh** (Apple Computer): enfocado actualmente para el diseño gráfico.
- **UNIX (AT & T)** (FreeBSD): el más antiguo. Multiusuario y multiprocesamiento.
- **Linux** (Red Hat, Susse, Debian, entre otros): más utilizado para servidores de red.

2) El Software de Programación

- Son todos los programas y utilitarios que facilitan la construcción de aplicaciones de usuarios.
- Incluye:
 - Los intérpretes
 - Los compiladores
 - Los lenguajes para programar
 - Los utilitarios para depurar los programas.
- Ejs: Visual Studio, Borland, Turbo C, Java, Delphi, entre otros.

3) Software de Diagnóstico y mantenimiento

- Es el software utilizado por el personal que se encarga de poner a funcionar los equipos.
- Ayuda a localizar averías de un equipo o encontrar el funcionamiento de un paquete de software.
- Ej: Herramientas administrativas de Windows.

Software de Aplicación

- Son todos aquellos programas que controlan el funcionamiento de una computadora para realizar una tarea específica.
 1. **Software estándar:** son aquellas aplicaciones de uso general como: procesador de palabras, hojas electrónicas, administrador de bases de datos, herramientas de comunicación, entre otros. Ejs: MS Internet Explorer, Netscape, Ópera, MS Office, 2000, XP, OpenOffice (gratis).
 2. **Software a la medida:** constituido por aquellas aplicaciones específicas para usuarios concretos. Ej: Aplicaciones de contabilidad.

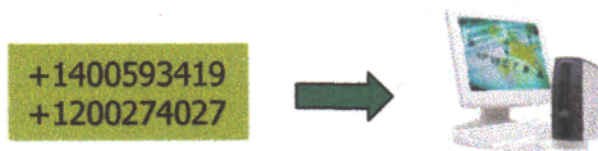
Lenguajes de programación

Un lenguaje de programación es un conjunto de símbolos junto a un conjunto de reglas para combinar los símbolos que se usa para expresar programas.



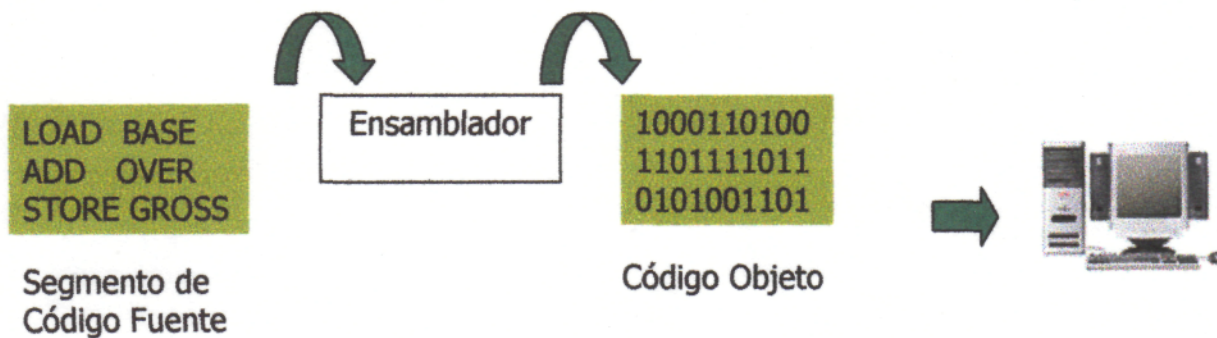
1) Lenguaje Máquina:

- Único lenguaje que la máquina entiende directamente. (0's y 1's)
- Lenguaje natural de cualquier computadora.
- Una computadora sólo puede entender su propio lenguaje máquina.
- Consiste de cadenas de números.
- No se pueden documentar. (ilegibles)
- Programación eficiente se aprovechan al máximo los recursos. (tiempo de ejecución y memoria)



2) Lenguaje ensamblador:

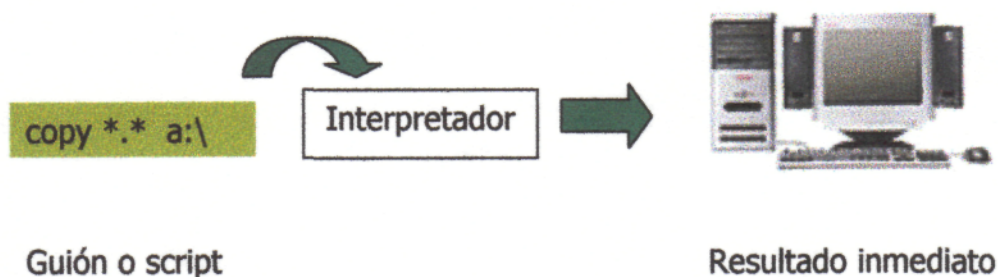
- Abreviaturas semejantes al inglés para representar operaciones elementales de la computadora. Ej: ADD
- **Ensambladores:** Programas de traducción que convierten los programas de lenguaje ensamblador a lenguaje máquina.
- También necesitan de muchas instrucciones para realizar tareas simples.



3) Lenguajes de Alto Nivel

- Con ellos se escriben enunciados simples para llevar a cabo tareas complejas.
- **Interpretores:** Programa que ejecuta las órdenes que se le brinden en uno o varios archivos.
- Traduce el programa fuente en un solo paso, cada instrucción se traduce y se ejecuta. El programador corrige de una vez cada error.

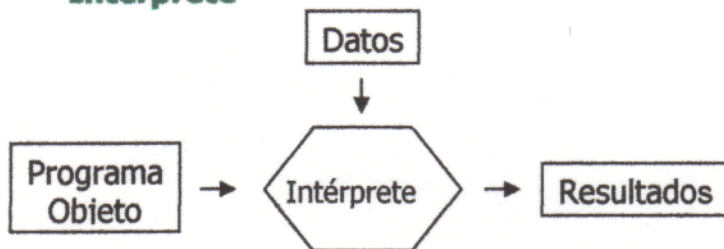
Ejemplo: intérprete de comandos (*shell* o consola)



Compilador



Intérprete



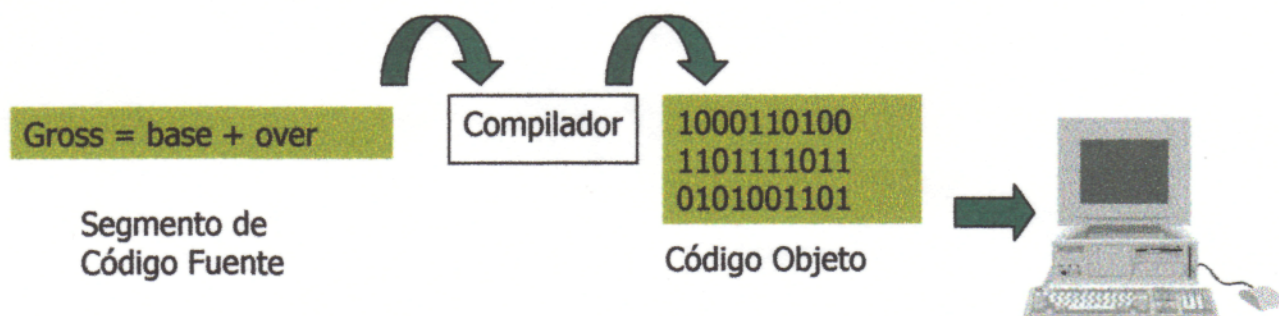
Ejemplo para shell de windows 2000

```

PATH
md algo020820
cd algo020820
dir
md algo1
dir algo1 > algo1.txt
type algo1.txt
cd ..
del algo020820
  
```

- Con ellos se escriben enunciados simples para llevar a cabo tareas complejas.
- **Compiladores:** Programas de traducción que convierten los programas de lenguaje de alto nivel a lenguaje máquina.

Ejemplo: compilador de C.



Ejemplo en C:

```
#include <stdio.h>

main( ){
    int n, i;
    float x=0, resultado=0, suma = 0;
    /* Leer el valor de n */
    printf("¿Cuántos números?: ");
    scanf("%d",&n);
    /* Leer los números*/
    for(i = 0; i < n; i++){
        printf("x = ");
        scanf("%f", &x);
        suma += x;
    }
    resultado = suma / n;
    printf("\El resultado es: %f.\n", resultado);
}
```

Lenguajes de programación de alto nivel de propósito general Compilados

Características generales (I)

- Independiente de la plataforma
- 1 instrucción -> N instrucciones de máquina.
 - Instrucción con datos de entrada y/o datos de salida.
- Permiten construcción por partes o módulos reutilizables.
- Código fuente->código objeto->código de máquina.
 - Compilador y Enlazador o Linker

Características generales (II)

- Datos de varios tipos
 - char, short, long, double, String
- Creación de tipos de datos abstractos y estructuras
- Operaciones aritméticas básicas y de comparación
- Control del flujo del programa
- Operaciones lógicas
 - AND, OR, NOT

Operaciones lógicas**1- Verdadero****2- Falso****AND (&&)**

p	q	p && q
1	1	1
1	0	0
0	1	0
0	0	0

Ejemplo:

- ☐ Como pan → Verdadero
- ☐ Bebo vino → Verdadero
- ☐ Tomo agua → Falso

- 1- Todas las mañanas tomo agua y como pan → La expresión es falsa ya que tomo agua es falso.
- 2- Como pan y bebo vino → ambas afirmaciones son verdaderas, por lo tanto la expresión es verdadera.

AND (||)

p	q	p q
1	1	1
1	0	1
0	1	1
0	0	0

Ejemplo:

- ☐ Como pan → Verdadero
- ☐ Bebo vino → Verdadero
- ☐ Tomo agua → Falso

- 1- Todas las mañanas tomo agua o como pan → A pesar de que la es falso que tomo agua, la expresión es verdadera. Esto porque al menos una es verdadera.
- 2- Como pan o bebo vino → ambas afirmaciones son verdaderas, por lo tanto la expresión es verdadera.

NOT (!)

p	!p
1	0
0	1

Ejemplo:

- ☐ Como pan → Verdadero
- ☐ Bebo vino → Verdadero
- ☐ Tomo agua → Falso

- 1- No como pan → La afirmación es falsa.
- 2- No tomo agua → La afirmación es verdadera.

■ La interpretación de una fórmula queda completamente determinado por los valores de verdad de las variables proposicionales que dicha interpretación asigna a las letras enunciativas que aparecen en esa fórmula. Una vez que conocemos el valor de verdad que la interpretación asigna a cada variable proposicional y tenemos presentes las definiciones de los conectivos resulta fácil determinar el valor de verdad que le corresponde a la fórmula completa.

■ El procedimiento de determinación requiere ir por pasos, estableciéndolos valores correspondientes a los diferentes niveles de subfórmulas (indicados por los paréntesis) hasta alcanzar el nivel de la fórmula completa. Así obtenemos una tabla de verdad para la fórmula en cuestión.

Ejemplo:

- O estás seguro y lo que dices es cierto o mientes como un bellaco.

s estar seguro.

d decir la verdad.

m mentir como un bellaco.

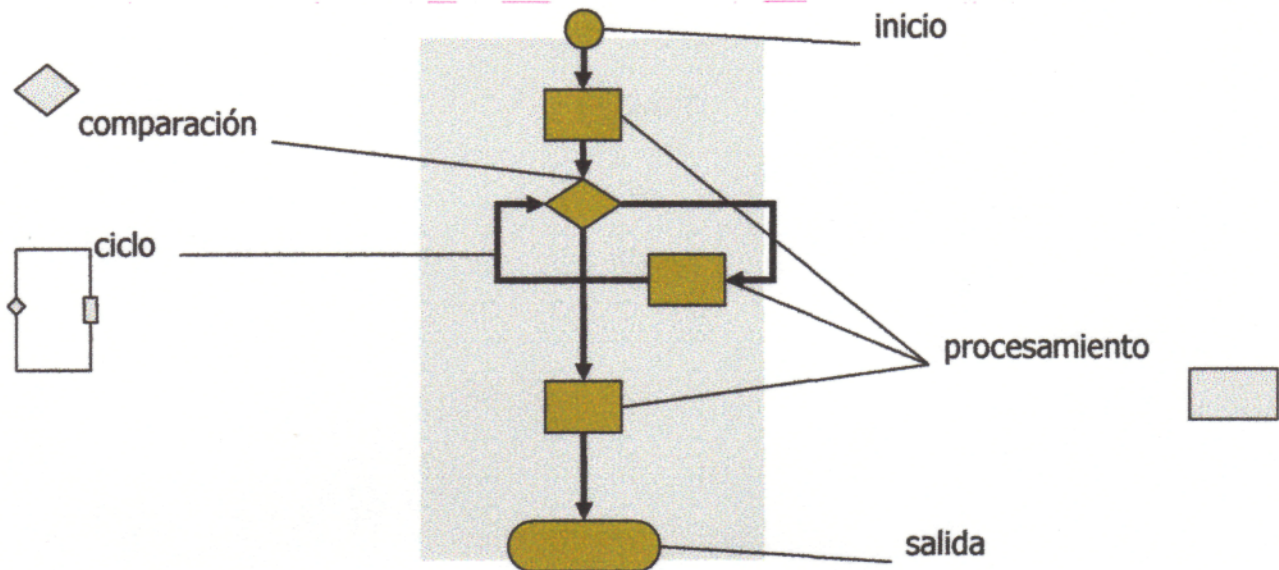
(s && d) || m

TABLA DE VERDAD:

s	d	M	s && d	(s && d) m
1	1	1	1	1
1	1	0	1	1
1	0	1	0	1
1	0	0	0	0
0	1	1	0	1
0	1	0	0	0
0	0	1	0	1
0	0	0	0	0

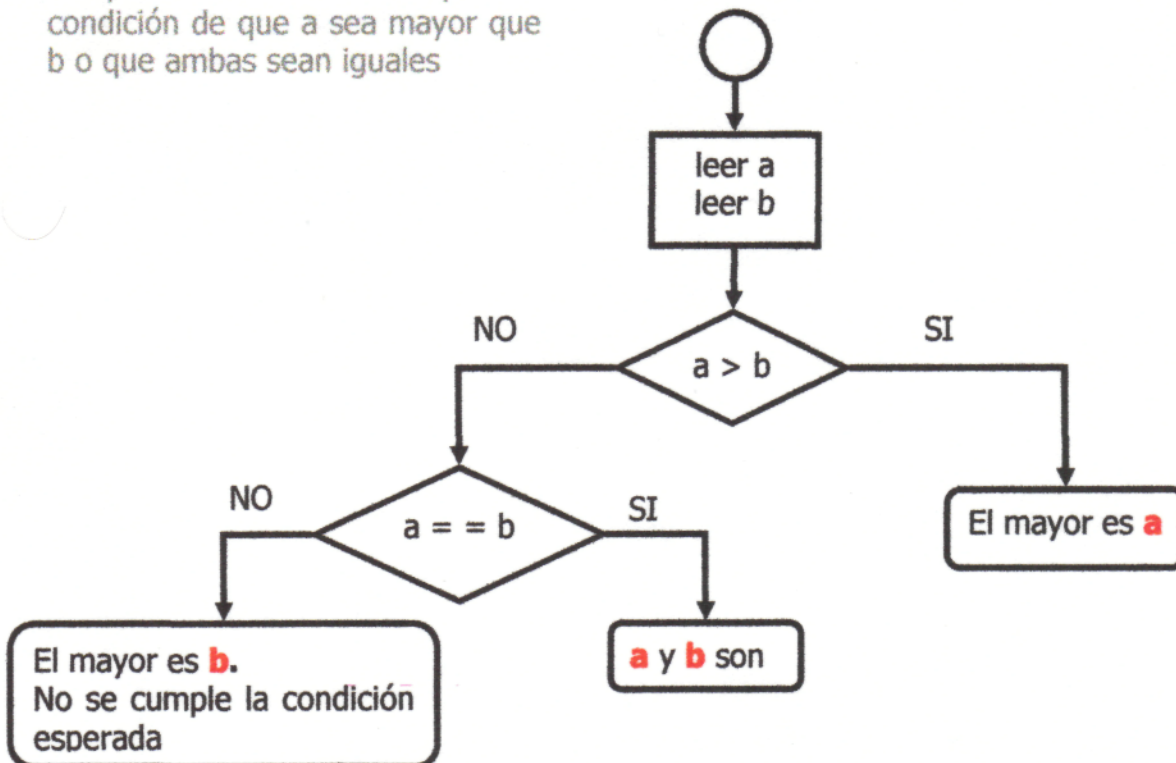
Control de flujo

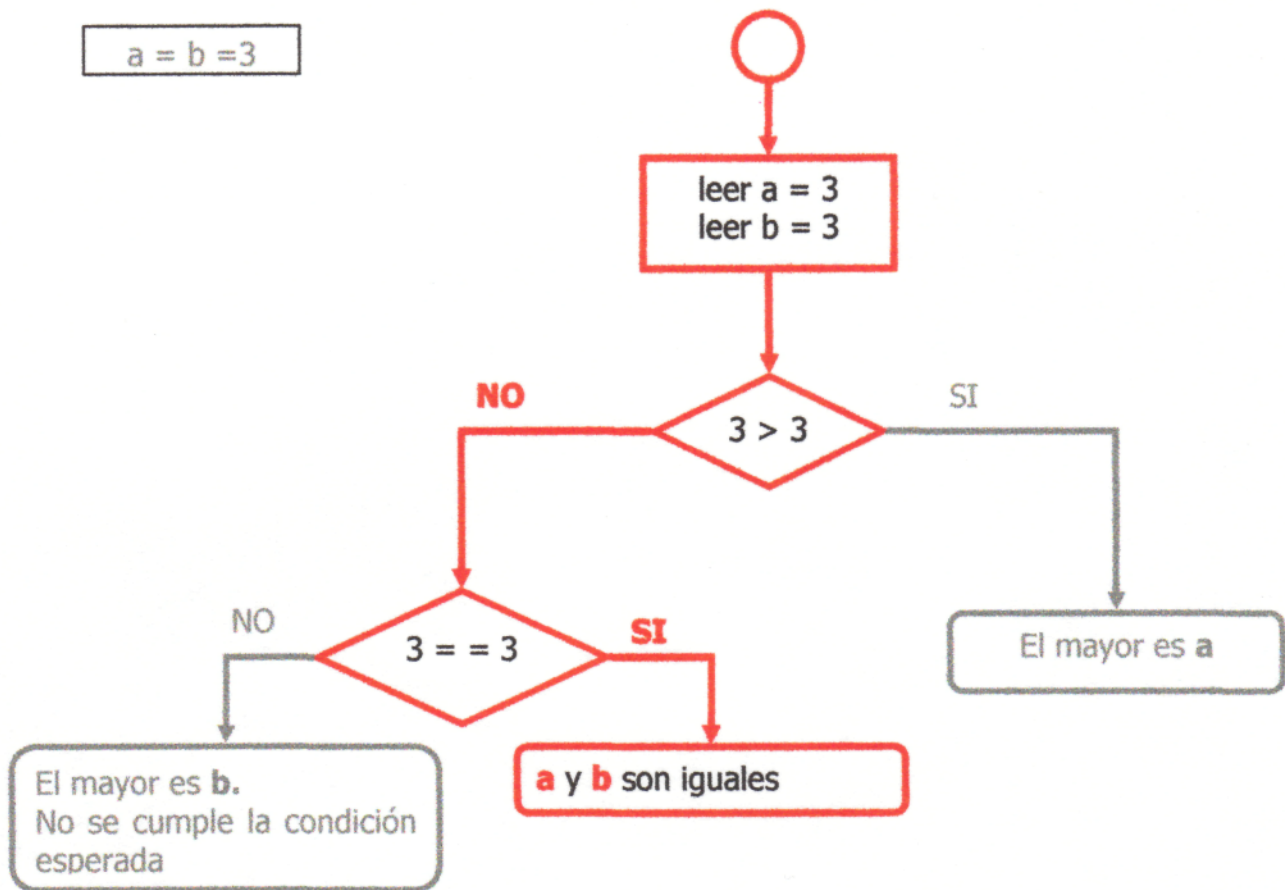
Gobierna la secuencia de las instrucciones y toma decisiones realizando comparaciones



$a \geq b$

Antes de iniciar debemos leer el valor de a y de b para poderlos comparar. Se debe cumplir la condición de que a sea mayor que b o que ambas sean iguales



$a \geq b$ $a = b = 3$ 

Se ingresa un número **n**, se suman todos los números desde 1 hasta **n-1** y el resultado se almacena en la variable **resultado**.

Ej.:

n = 5

