

TEMA 1

CONCEPTOS BÁSICOS DE INFORMÁTICA

1 DEFINICIONES GENERALES

Veamos algunas definiciones:

Informática:

- Ciencia encargada del tratamiento automático de la información
- Conjunto de conocimientos científicos y técnicos que hacen posible el tratamiento automático de la información por medio de ordenadores (RAE).
- Ciencia responsable del tratamiento automático y racional de la información considerada como soporte del conocimiento de la sociedad y las comunicaciones en los campos social, económico y técnico.
- Área de conocimiento que reúne todos los aspectos necesarios para el diseño y uso de los ordenadores.

Ordenador:

- Es una máquina electrónica digital que realiza operaciones aritmético-lógicas con los datos de entrada hasta que obtiene el resultado.



- Un ordenador es, por tanto, una máquina que procesa información para obtener unos resultados o salida. La ejecución de un programa implica la realización de unos tratamientos, según especifica un conjunto ordenado de instrucciones (es decir, un programa) sobre unos datos. Para que el ordenador ejecute un programa, es necesario darle información de dos tipos:
 - Instrucciones que forman el programa (órdenes).
 - Los datos con los que debe operar el programa.

Hardware y software:

- El soporte físico o hardware de un ordenador es la máquina en sí, es decir, el conjunto de circuitos electrónicos, cables, carcasas, dispositivos electromecánicos y otros elementos físicos que lo forman.
- Conjunto de componentes que integran la parte material de una computadora (RAE).
- El soporte lógico o software es el conjunto de programas (sistema operativo, de utilidades y de los usuarios) ejecutables en el ordenador.
- Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora (RAE).

Memoria:

- La memoria del ordenador es la parte del hardware que retiene, durante cierto período de tiempo, información para datos, resultados parciales, instrucciones que constituyen los programas que necesita el sistema para funcionar correctamente.
- Dispositivo físico, generalmente electrónico, en el que se almacenan datos e instrucciones para recuperarlos y utilizarlos posteriormente (RAE).

En un ordenador doméstico, la memoria generalmente toma las formas de:

- Disco duro: el cual almacena los archivos y programas del usuario. Mantiene la información incluso si el ordenador es apagado, y tiene una gran capacidad de almacenamiento.

Ejemplo:

En el disco duro guardaremos nuestros programas para que en la siguiente sesión podamos acceder a ellos, reeditarlos, etc.

- Memoria RAM: usada para almacenar cosas tales como programas e información de manera temporal mientras el ordenador está siendo utilizado. Esto se hace porque este tipo de memoria es rápido y es útil para acceder a datos sin tener que leerlos de un soporte de almacenamiento más lento, como un disco duro. La información se pierde cuando el ordenador es apagado (es volátil).
- Memoria caché: es generalmente encontrada en los microprocesadores, discos duros u otros aparatos. Es una pequeña cantidad de memoria de muy alta velocidad dedicada a que partes importantes del ordenador puedan trabajar a la máxima velocidad sin necesidad de pedir información constantemente a partes más lentas del resto del sistema.
- Memoria secundaria: es un conjunto de dispositivos periféricos para el almacenamiento masivo de datos de un ordenador, con mayor capacidad que la memoria principal, pero más lenta que ésta. El disquete, el disco duro o disco fijo, las unidades ópticas, las unidades de memoria flash o los discos ZIP pertenecen a esta categoría. Estos dispositivos periféricos quedan vinculados a la memoria principal, o memoria interna, conformando el subsistema de memoria del ordenador. Soportes de memoria secundaria: CD, CD-R, CD-RW DVD, DVD-/+R, DVD-/+RW, Disquete, Disco duro, Cinta magnética, Pen-drives.

Ejemplo:

En la memoria secundaria guardaremos nuestros datos si queremos portarlos a otro equipo para que en otra sesión y en otro equipo podemos acceder a ellos, reeditarlos, etc.

Periféricos:

- El ordenador es una máquina que no tendría sentido si no se comunicase con el exterior, es decir, si careciese de periféricos. Se denominan periféricos tanto a los dispositivos a través de los cuales el ordenador se comunica con el mundo exterior, como a los sistemas que almacenan información, sirviendo de memoria auxiliar de la memoria principal, independientemente de que se encuentre en el interior o en el exterior de la carcasa.
- Podemos distinguir entre:
 - o Unidad(es) de entrada, a través de la(s) cual(es) introducirle los programas que queremos que ejecute y los datos correspondientes. (Teclado, ratón)
 - o Unidad(es) de salida, con la(s) que el ordenador da los resultados de los programas (pantalla, impresora)
 - o Memoria masiva o auxiliar, que facilite su funcionamiento u utilización (pendrive)

Instrucción:

- Una instrucción es un conjunto de símbolos que representan una orden de operación o tratamiento para el ordenador.
- Expresión formada por números y letras que indica, en una computadora, la operación que debe realizar y los datos correspondientes (RAE).

Ejemplo:

```
printf ("Hola");
```

Programa:

- Un programa es un conjunto ordenado de instrucciones que se da al ordenador
- Conjunto unitario de instrucciones que permite a un ordenador realizar funciones diversas, como el tratamiento de textos, el diseño de gráficos, la resolución de problemas matemáticos, el manejo de bancos de datos, etc. (RAE)

Ejemplo:

```
#include <stdlib.h>
#include <stdio.h>
void main()
{
    char a,b;
    scanf("%c,%c",&a,&b);
    printf ("%d",a);
    printf ("%d",b);
    system("PAUSE");
}
```

Memoria, entrada, salida, instrucción:

Despacho del Sr. Controlador



El Sr. Controlador ordena los datos que le llegan por la ventanilla de entrada y los coloca en el archivador para utilizarlas



El Sr. Controlador ejecutando una orden



El Sr. Controlador envia los resultados a través de la ventanilla de salida

Lenguaje de programación:

Las instrucciones se construyen siguiendo unas reglas precisas y con símbolos de un determinado repertorio. Un lenguaje de programación es artificial y está compuesto por el conjunto de reglas, símbolos y palabras reservadas que se utilizan en la construcción de un programa y que permiten la comunicación con un ordenador.

Lenguaje de alto nivel:

Lenguaje que facilita la comunicación con un computador mediante signos convencionales cercanos a los de un lenguaje natural. Hay un alto nivel de abstracción entre lo que se pide a la computadora y lo que realmente comprende. Existe también una relación compleja entre los lenguajes de alto nivel y el código máquina. Los lenguajes de alto nivel son normalmente fáciles de aprender porque están formados por elementos de lenguajes naturales, como el inglés.

Ejemplo:

```
if (contador == 0)
    printf ("El valor es cero");
```

En un lenguaje de programación podemos hablar de **sintaxis** o de conjunto de normas para la escritura de sentencias (una sentencia que no sigue las normas es ERRÓNEA) y **semántica** o significado de las sentencias (una sentencia semánticamente incorrecta cumple las normas pero no hace lo que queremos porque no hemos utilizado la sentencia correcta).

Lenguaje máquina:

Conjunto de instrucciones codificadas que una computadora puede interpretar y ejecutar directamente. Estos lenguajes están formados por diferentes combinaciones de 0 y 1, que es lo único que un ordenador puede entender. En lenguaje máquina están escritos los programas ejecutables.

¿Por qué sólo 0 y 1? Porque el ordenador no es más que un conjunto de circuitos digitales, y los circuitos lo único que entienden es un sistema binario (es decir, 1 = corriente, 0 = no corriente).

Archivo ejecutable:

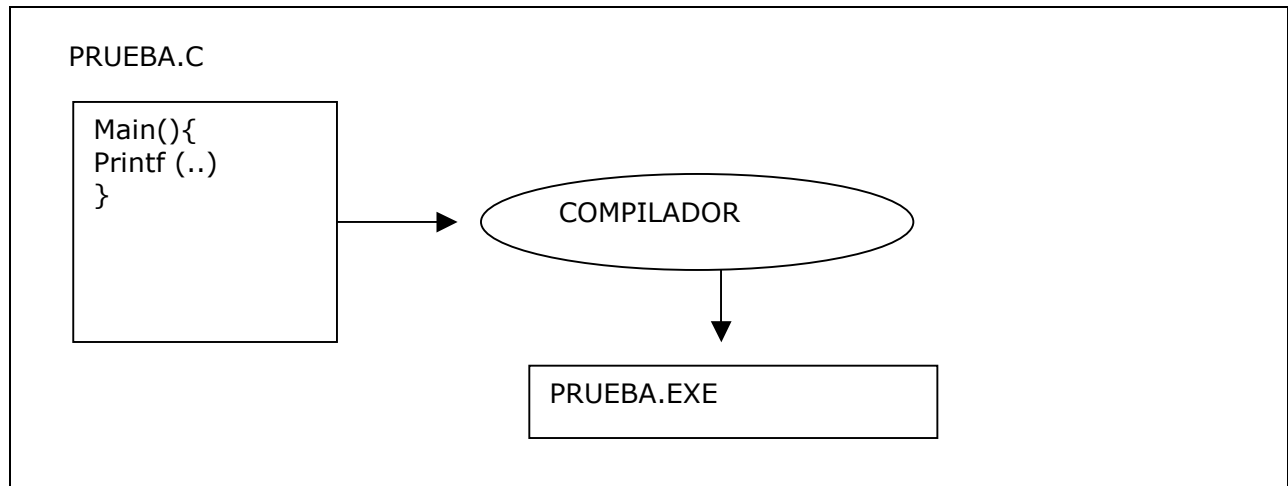
Un ejecutable, o archivo ejecutable, en informática, es un archivo cuyo contenido se interpreta por el ordenador como un programa. Generalmente contiene instrucciones en código máquina de un procesador en concreto, pero también puede contener bytecode que requiera un intérprete para ejecutarlo. Además suele contener llamadas a funciones específicas de un sistema operativo (llamadas al sistema). Dependiendo del tipo de que se traten las instrucciones, hablaremos de ejecutables portables (se pueden ejecutar en varias plataformas) y no portables (destinado a una plataforma concreta). Por ejemplo, un ejecutable Java es portable, ya que utiliza un bytecode no asociado a un procesador concreto.

Compilador:

Programa que convierte el lenguaje informático empleado por el usuario en lenguaje propio del computador.

Ejemplo:

Es la aplicación que convierte un programa escrito en lenguaje de programación **C** en código ejecutable. Nosotros editamos un programa en lenguaje C como el que aparece anteriormente y obtenemos lo que se denomina ejecutable o programa que directamente podremos ejecutar.

**Sistema operativo:**

Programa o conjunto de programas que efectúan la gestión de los procesos básicos de un sistema informático, y permite la normal ejecución del resto de operaciones. Sus funciones principales son el reparto del tiempo de ejecución, la gestión de Entrada/Salida, gestión de la memoria y gestión de la información. Las características de un buen sistema operativo son la eficiencia, fiabilidad, facilidad de mantenimiento y la utilización de pocos recursos.

Ejemplos:

MS-DOS, Windows, UNIX, LINUX, OS/2, OS/400, AIX

Aplicaciones:

Programas preparados para una utilización específica, como procesadores de texto, hojas de cálculo, bases de datos, aplicaciones gráficas, el pago de nóminas, formación de un banco de términos léxicos, etc.



Algoritmo:

- Secuencia de pasos finita, ordenada y no ambigua que describe la resolución de un problema.
- Conjunto ordenado y finito de operaciones que permite hallar la solución a un problema (RAE).
- Las características de un algoritmo son:
 - Es preciso.
 - Debe expresar el orden entre los pasos.
 - Siempre se comporta de la misma forma en iguales condiciones.
 - Debe finalizar.
 - Es independiente del lenguaje de programación.

Un ejemplo no informático:

Hacer una tortilla de patatas para 6 personas

INGREDIENTES:	UTENSILIOS:
6 huevos frescos	plato, sartén
1 Kg de patatas	cuchillo, tenedor

PREPARACIÓN

1. Pelar y trocear las patatas, dejándolas en el plato
2. Batir los huevos en un bol
3. Mezclar las patatas con los huevos batidos
4. Calentar una sartén con un poco de aceite
5. Verter los huevos con patatas en la sartén...

Cada persona podría expresar esto mismo en distintos pasos o de distinta manera (con otras palabras, etc.). Es por ello que utilizaremos para describir un algoritmo lo que se denomina **pseudocódigo**, que es un sistema de signos convencionales adoptado para especificar los pasos u operaciones a realizar.

2 ¿CÓMO TRABAJAREMOS NOSOTROS?

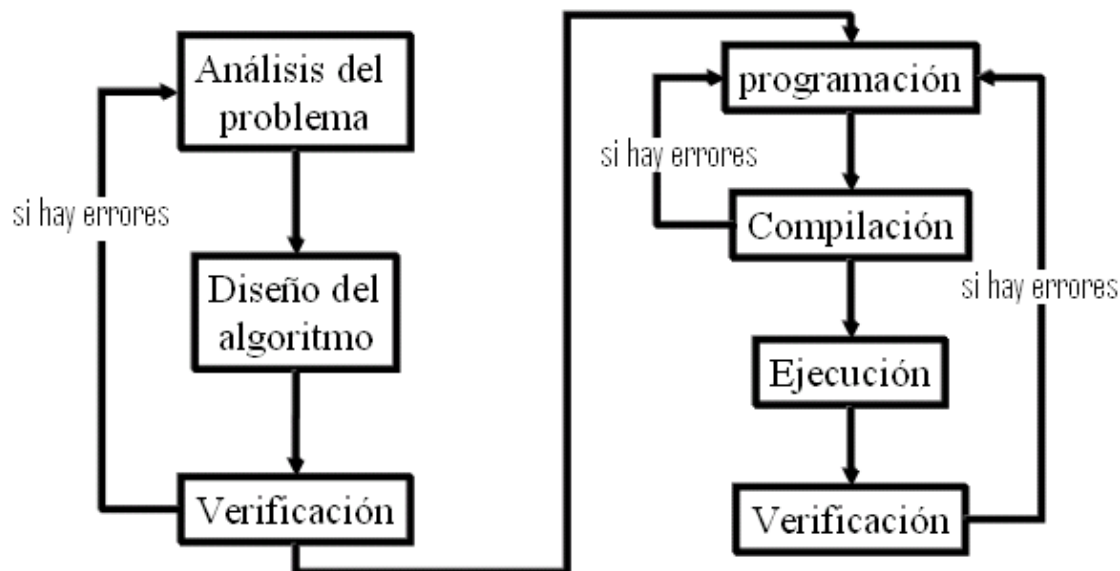
Resumiendo, a nosotros se nos dará un problema a resolver utilizando lenguaje C, para lo cual tendremos que:

1. Pensar el algoritmo de resolución y escribir el pseudocódigo.
2. Traducir esa solución a un programa en un lenguaje de alto nivel que será lenguaje C (datos + órdenes). Necesitaremos saber las normas del lenguaje C, lo mismo que para explicar algo en inglés necesitamos saber las palabras y normas que no podremos saltarnos si queremos que el ordenador nos entienda.
3. Escribiremos el programa en lenguaje C utilizando el programa que luego compilará, es decir, traducirá este programa a lenguaje máquina (a 0 y 1) y generará un ejecutable para que el ordenador pueda ejecutar las instrucciones.



En este proceso puede haber errores que nos obliguen a volver atrás si vemos que el algoritmo propuesto no es correcto, o que la compilación nos da errores de sintaxis (cuando escribimos en lenguaje C el compilador nos dice que "no entiende" porque no hemos seguido las normas) o bien en la ejecución observamos que nos hemos equivocado en la resolución porque el programa no hace lo que esperábamos en todos los casos (puede que el significado o semántica de lo que habíamos escrito fuera incorrecto).

Este proceso, más en detalle indicando las verificaciones, validaciones y las vueltas atrás aparecen en el siguiente gráfico:



Para todo esto será imprescindible conseguir el compilador, en nuestro caso será **DevC ++**, e instalarlo correctamente en el ordenador. A partir de este momento ya estaremos en situación de empezar a aprender C para poder escribir nuestro primer programa.

