

A faint, stylized illustration of a laptop and a tablet. The laptop screen displays the FACYT logo and the text 'FACYT'. The tablet also shows the FACYT logo.

# Aspectos Introdutorios

Tema 01  
(Parte 3)

# Aspectos Introductorios

## Dato e Información

### Dato

La palabra proviene del latín *datum* que significa “lo que se da”.

Es una representación simbólica (numérica o alfabética) de un atributo o una característica de un hecho o una entidad.

Constituye la representación convencional de una información bajo la forma conveniente para su tratamiento por un computador.

El dato no tiene sentido (valor semántico) en sí mismo, pero si recibe un tratamiento (procesamiento) apropiado, se puede utilizar en la realización de cálculos o toma de decisiones.

### Ejemplos

1. 15

2. 25º C

3. 25 mts

4. 31-12-1983

5. Febrero

6. María Pérez

## Dato e Información

### Información

Los datos organizados o procesados adquieren significado y proporcionan conocimiento o desencadenan un comportamiento sobre los hechos u objetos que lo originaron.

Por lo tanto se tiene que:

**Datos + Significado = Información**

Un grupo de datos es información:

Datos:

19 – Marzo – María – 20 puntos.

Información:

El 19 de marzo se publicó en acta la calificación de 20 puntos para María.

El principal objetivo de la información consiste en aumentar el conocimiento o reducir la incertidumbre.

### Ejemplos

1. Juan tiene 15 años.
2. 25º C fue la temperatura de esta mañana.
3. El edificio “Robles” mide 25 m de altura.
4. La fecha de hoy es 31-12-1983.
5. Febrero tiene 28 días y 29 si el año es bisiesto.
6. María Pérez es un nombre común.

## Problema

Es un asunto o un conjunto de cuestiones que se plantean para ser resueltas.

La naturaleza de los problemas varía con el ámbito o el contexto:

- Problemas matemáticos,
- problemas computacionales,
- problemas químicos,
- problemas filosóficos,
- problemas religiosos,
- entre otros.

Para abordar un problema es importante tener una descripción simple y precisa del mismo, de lo contrario resultaría complejo modular, simular, o programar su solución en un computador.

Un programador es ante que nada una persona que resuelve problemas, y para llegar a ser un programador eficaz se necesita aprender a resolver problemas de un modo riguroso y sistemático.

## Algoritmo

### Etimología

Proviene del nombre del matemático persa Mohammed Al-khowanzmi, cuyo apellido traducido al latín es Algorismus.

### Definición

Es un conjunto finito de pasos y acciones que especifican de forma clara y concisa la secuencia de operaciones a realizar para procesar adecuadamente una serie de datos con el objetivo de resolver un problema.

**Nota:** Los algoritmos son independientes tanto del lenguaje de programación como del computador que los ejecuta.

### Características

1. **Entrada:** Un algoritmo puede tener cero o mas entradas.
2. **Salida:** Un algoritmo puede tener una o mas salidas.
3. **Finito:** Un algoritmo siempre debe terminar después de un número de pasos conocido.
4. **Definido:** Tiene un propósito en particular. Si se sigue un algoritmo dos veces se debe obtener el mismo resultado.
5. **Efectivo:** Debe resolver el problema para el cual fue concebido.



### Ejemplo: Cambiar el caucho de un carro.

1. Buscar las herramientas, caucho de repuesto y triángulo.
2. Ubicar el triángulo en el lugar adecuado.
3. Ir al lugar del caucho averiado.
4. Sacar las tuercas.
5. Colocar el gato.
6. Levantar el carro.
7. Sacar el caucho.
8. Colocar el caucho de repuesto.
9. Colocar las tuercas.
10. Apretar las tuercas.
11. Bajar el carro.
12. Sacar el gato.
13. Guardar las herramientas, caucho averiado y triángulo.



## Programa

Es la codificación de un algoritmo en un computador, realizado en un lenguaje de programación reconocido por la máquina.

### Fases para el Desarrollo de un Programa



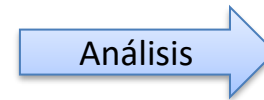
## Programa

### Fases para el Desarrollo de un Programa

1. **Análisis:** Es el estudio detallado del problema con el fin de obtener una serie de documentos (especificación) en los cuales quede definido el proceso a seguir en la automatización.



Estudio Detallado

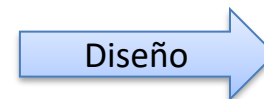


Documentos de Especificación

2. **Diseño:** Consiste en la realización del algoritmo que resuelve el problema de acuerdo a la especificación dada en la fase anterior. El algoritmo se representa en una notación pseudoformal que también se conoce como pseudocódigo.



Documentos de Especificación



**Algoritmo PRUEBA**  
**Variables**

a, b, c: entero

**Inicio**

leer(a, b)

$c \leftarrow a + b$

escribir (c)

**Fin**

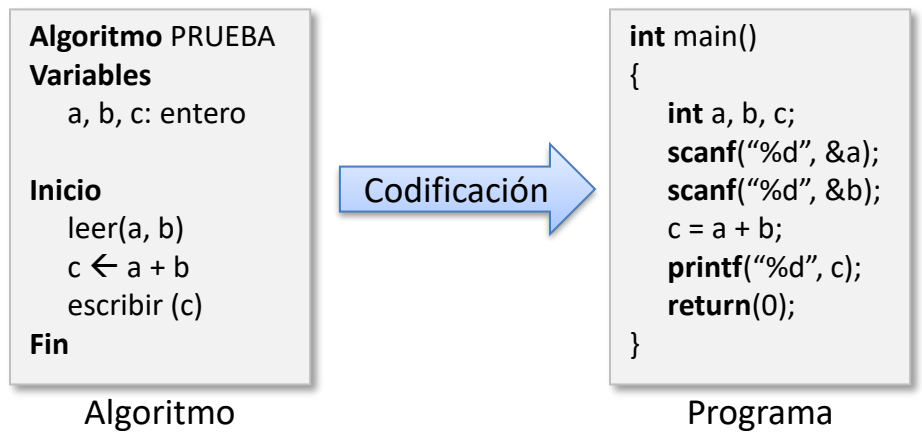
Algoritmo



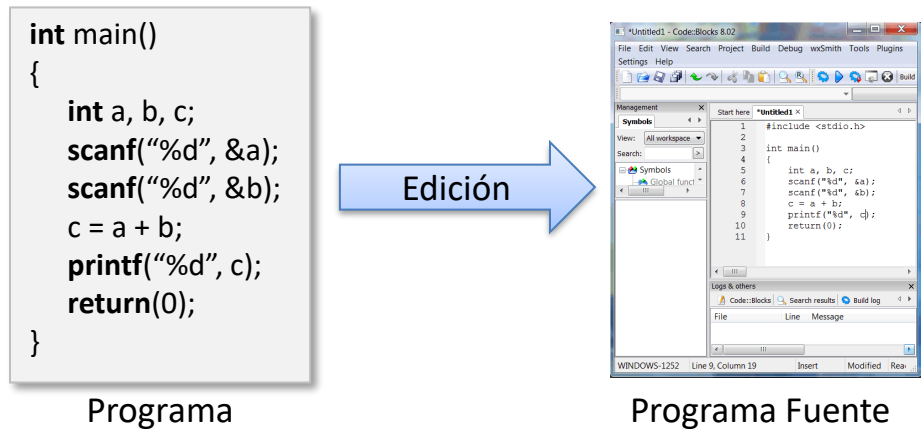
## Programa

### Fases para el Desarrollo de un Programa

**3. Codificación:** Consiste en la traducción del algoritmo a un programa escrito en un lenguaje de programación.



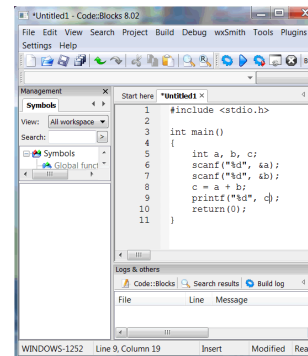
**4. Edición:** Consiste en la transcripción del programa al computador, generalmente por medio de un editor de programas o procesador de textos, obteniendo un programa fuente.



## Programa

### Fases para el Desarrollo de un Programa

**5. Compilación:** Consiste en obtener el programa objeto a partir del programa fuente, mediante el traductor de lenguaje, el cual además de efectuar la traducción, incluye un análisis sintáctico.



```

1 #include <stdio.h>
2
3 int main()
4 {
5     int a, b, c;
6     scanf("%d", &a);
7     scanf("%d", &b);
8     c = a + b;
9     printf("%d", c);
10    return(0);
11 }
    
```

Programa Fuente



```

01001011 01101001
01100001 01110010
01100001 00100000
01000001 00101110
00100000 01001111
01110100 01110100
01101111 01100111
01100001 01101100
01101100 01101001
00100000 01000110
    
```

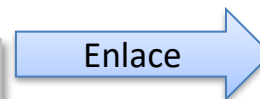
Programa Objeto

**6. Enlace:** Consiste en la inclusión de determinadas rutinas internas del lenguaje y en el caso de programación modular, se enlazan los distintos módulos.

```

01001011 01101001
01100001 01110010
01100001 00100000
01000001 00101110
00100000 01001111
0
0
0
0
0
01001011 01101001
01100001 01110010
01100001 00100000
01000001 00101110
00100000 01001111
01110100 01110100
01101111 01100111
01100001 01101100
01101100 01101001
00100000 01000110
    
```

Programa Objeto +  
Rutinas



```

01001011 01101001
01100001 01110010
01100001 00100000
01000001 00101110
00100000 01001111
01110100 01110100
01101111 01100111
01100001 01101100
01101100 01101001
00100000 01000110
01100001 01101100
01101100 01101001
00100000 01000110
    
```

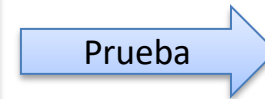
Programa Ejecutable

### Fases para el Desarrollo de un Programa

7. **Prueba:** Consiste en determinar si el programa funciona correctamente, los datos de entrada que se utilizan para la prueba, deben ser incorrectos.

```
01001011 01101001
01100001 01110010
01100001 00100000
01000001 00101110
00100000 01001111
01110100 01110100
01101111 01100111
01100001 01101100
01101100 01101001
00100000 01000110
01100001 01101100
01101100 01101001
00100000 01000110
01100001 01101100
01101100 01101001
00100000 01000110
```

Programa Ejecutable



Aplicación

## Otros Conceptos

### Lenguaje

Es un conjunto de reglas y convenciones que se utilizan para comunicar información.

### Lenguaje Pseudoformal

Es un lenguaje que se utiliza para especificar un algoritmo independientemente del lenguaje de programación en el cual éste se vaya a codificar. La ventaja de este lenguaje es que el programador puede concentrarse en la lógica y en la estructura de control y no preocuparse de las reglas de un lenguaje

### Lenguaje de Programación

Es un lenguaje que permite la traducción (codificación) de un algoritmo para que sea entendible por el computador. Son los lenguajes utilizados para escribir programas de computadoras. Viene definido por su sintaxis y su semántica.

Sintaxis: Consta de unas definiciones que especifican la secuencia de símbolos que forman una frase del lenguaje. Dicen si la frase esta bien escrita o no.

Semántica: Define el significado de las reglas sintácticas del lenguaje, expresiones y tipos de datos utilizados.

## Otros Conceptos

### Lenguajes Máquina

Son aquellos que están escritos en un lenguaje directamente inteligible por la máquina, ya que sus instrucciones son cadenas binarias (0's ,1's) que especifican una operación.

Las instrucciones en lenguaje máquina dependen del Hardware de la computadora, difiere de una PC a otra.

La ventaja de los lenguajes máquina es que ofrecen mayor velocidad de ejecución.

Existen algunas desventajas: dificultad y lentitud en la codificación, poca fiabilidad, los programas no son portables.





## Otros Conceptos

### Lenguajes de Bajo Nivel

Son lenguajes que permiten escribir programas con instrucciones similares al lenguaje humano.

Son más fáciles de utilizar que los lenguajes de máquina, pero éstos también dependen de la máquina en particular.

Un programa escrito en lenguaje ensamblador requiere una fase de traducción.

Entre las desventajas tenemos: que este lenguaje depende de la máquina y demanda una mayor exigencia para los programadores, ya que deben conocer tanto las técnicas de programación así como el interior de la máquina

**Ejemplo:** El lenguaje ensamblador.

## Otros Conceptos

### Lenguajes de Alto Nivel

Son los más utilizados por los programadores.

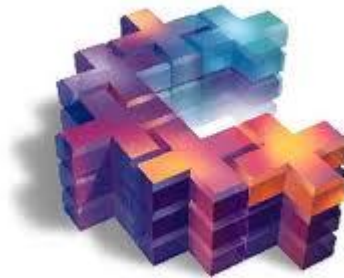
Son independientes de la máquina.

Los programas escritos en estos lenguajes son portables.

Aumento de la ocupación de la memoria.

Las estructuras de los programas se basan en reglas sintácticas.

**Ejemplos:** Borland C, C++, Java, Pascal, Ada, Prolog, Smalltalk, Visual Basic.



## Otros Conceptos

### Programación

Disciplina dentro de la computación dedicada a la resolución de problemas mediante el computador, cuyo producto final son los programas. Es la actividad de expresar un algoritmo en forma de programa

### Paradigmas de Programación

Son filosofías, modelos o patrones para programar.

#### 1. Imperativo

La unidad de trabajo es la sentencia o acción. Lenguajes: Pascal, C, PHP, Perl.

#### 2. Funcional

La unidad de trabajo es la función. Consiste en combinar funciones para conseguir funciones mas complejas hasta llegar a la función que es el programa. Lenguajes: Lisp, Scheme.

#### 3. Orientado a Objetos

La unidad de trabajo es el objeto. Agrupa las estructuras de datos con los algoritmos. Lenguajes: C++, Java, Smalltalk.

#### 4. Lógico

Se especifican los hechos y propiedades que definen el problema. El sistema utiliza esa información para llegar a la solución. Lenguaje: Prolog.

## Otros Conceptos

### **Acción**

Es un suceso que dura un tiempo finito y produce un resultado o efecto perfectamente definido y previsto. Opera sobre un objeto y es posible reconocer el efecto de tal acción por los cambios de estado presentados por el objeto.

### **Instrucción**

Es la especificación de una acción determinada, expresada en un lenguaje de programación.

### **Método de Calculo**

Procedimiento que tiene todas las características de un algoritmo excepto que posiblemente carezca de ser finito.

**Programa Fuente:** es el código del programa original escrito en un lenguaje de programación. También conocido como código fuente

**Programa Objeto:** es un programa en lenguaje de máquina que resulta de la compilación de un programa fuente.

## Otros Conceptos

### Traductor

Es un programa que traduce los programas fuentes escritos en lenguajes de alto nivel a código máquina. Estos se dividen en compiladores e intérpretes.

### Compilador

Es un traductor que transforma programas fuentes a programas objetos. En este proceso de transformación se debe utilizar como paso intermedio un programa llamado enlazador (*linker*) el cual da como resultado un programa en lenguaje máquina directamente ejecutable.

### Intérprete

Es un traductor que toma un programa fuente, lo traduce y a continuación lo ejecuta. (Basic, Smalltalk).

### Ensamblador

Programas que transforman programas escritos en lenguaje ensamblador a lenguaje máquina y ejecutable directamente.



## Otros Conceptos

### Sistema Operativo

Es un conjunto de programas de control hecho en lenguaje de bajo nivel, compuesto por un conjunto de normas y procedimientos para facilitar el uso de la computadora y conseguir que ésta se ejecute eficientemente.

Su principal labor es el control sistemático de los recursos de la computadora al momento de ejecutar un proceso, es decir, el SO es el encargado de supervisar, administrar, gestionar y asignar los recursos de Hardware al usuario: La CPU, la memoria, los discos duros y otros periféricos.

Permite gestionar y administrar eficientemente los recursos de *Hardware* de la computadora, permitiendo ejecutar de forma concurrente varios programas sin que haya conflicto en el acceso a los recursos que se requieren y sin que ningún programa monopolice un recurso determinado.



### Tipos de Sistemas Operativos

Se clasifican de acuerdo a la forma de ofrecer los servicios:

1. Sistema operativo por lotes
2. Sistema operativo de multiprogramación
  - 2.1. Sistema operativo de tiempo compartido
  - 2.2. Sistema operativo de tiempo real
  - 2.3. Sistema operativo de combinado
3. Sistema operativo distribuido

### Tipos de Sistemas Operativos

#### 1. Sistema operativo por lotes

- Se podrían considerar como los primeros SOs reales.
- El SO estaba almacenado en memoria.
- Cargaba un único trabajo en memoria (desde el lector de tarjetas).
- Ejecutaba el trabajo (generaba su salida).
- Cargaba el siguiente trabajo.
- Las tarjetas de control indicaban qué hacer al SO.

#### Problemas

- Largas esperas entre lotes de trabajo.
- La dificultad de manejar la concurrencia entre E/S y ejecución en CPU ayudó a estimular el desarrollo de los SO multiprogramados.
- Carecen de interactividad entre el usuario y los trabajos que se ejecutan.

### Tipos de Sistemas Operativos

#### 2. Sistema operativo de multiprogramación

- Objetivo: Aprovechar los tiempos de espera de un trabajo en la CPU para ejecutar instrucciones de otro trabajo.
- Método:
  - ✓ Mantener los trabajos simultáneamente en memoria.
  - ✓ Elegir el trabajo a conmutar.
- Se optimiza la productividad (*throughput*) del sistema.
- Se clasifican en:
  - 2.1. Sistema operativo de tiempo compartido.
  - 2.2. Sistema operativo de tiempo real.
  - 2.3. Sistema operativo combinado.

### Tipos de Sistemas Operativos

#### 2. Sistema operativo de multiprogramación

##### 2.1. Sistema operativo de tiempo compartido

- Objetivo: Permitir la interacción entre el usuario y el trabajo que se está ejecutando.
- Método:  
Utilizar las técnicas de multiprogramación y planificación de la CPU para proporcionar a cada usuario una pequeña proporción del tiempo de CPU.
- Características:
  - ✓ Cada usuario tiene la impresión de disponer de la máquina completa.
  - ✓ Se intenta optimizar el tiempo de respuesta.
  - ✓ Basados en la asignación de fracciones de tiempo (se divide el tiempo de CPU de forma equitativa entre los procesos).



### Tipos de Sistemas Operativos

#### 2. Sistema operativo de multiprogramación

##### 2.2. Sistema operativo de tiempo real

- El factor tiempo se convierte en una variable fundamental.
- El sistema operativo debe responder siempre dentro de los márgenes requeridos por el sistema controlado, si se sale de esos márgenes, se tendrán que buscar otras soluciones.
- Existen actualmente lenguajes de programación (Ada) que nos dan grandes posibilidades para controlar procesos en tiempo real.
- Se usan en: Sistemas de control de procesos, sistemas de control de vuelo, sistemas de control de automóviles, sistemas de defensa, sistemas de vigilancia intensiva, sistemas de telecomunicación, entre otros.

##### 2.3. Sistema operativo combinado

- Combina las características de los SO de tiempo compartido y tiempo real.

### Tipos de Sistemas Operativos

#### 3. Sistema operativo distribuido

- Los usuarios pueden acceder a recursos remotos de la misma manera en que lo hacen para los recursos locales.
- Permiten distribuir trabajos, tareas o procesos, entre un conjunto de procesadores. Puede ser que este conjunto de procesadores esté en un equipo o en diferentes, en este caso es transparente para el usuario.
- Los sistemas distribuidos deben de ser muy confiables, ya que si un componente del sistema se descompone otro componente debe de ser capaz de reemplazarlo.

## Otros Conceptos

### Proceso

Es básicamente un programa en ejecución. Consta del programa ejecutable, sus datos y pila, contador y otros registros, además de toda la información necesaria para ejecutar el programa.

### Multiprogramación

Modo de operación en cual se incrementa la utilización del CPU, puesto que éste se conmuta de una tarea a otra para lograr que las mismas avancen, mientras se mantienen en uso los dispositivos periféricos. Estas tareas se encuentran en memoria principal.

### Grado de Multiprogramación

Mide el uso de la CPU desde un punto de vista probabilístico. Normalmente es el rendimiento que se obtiene del CPU en función del número de procesos activos que mantiene un sistema.