

INDICE

Concepto computador básico.

Algoritmo.

Programa.

El proceso de la creación de un programa (Programación).

Sistemas de numeración.

Concepto computador básico

La computadora es un dispositivo electrónico diseñado para procesar información.

- La información es representada con números decimales a través de una cadena de dígitos binarios.
- Almacena (**memory o memoria**) y procesa datos (**Microprocessor o microprocesador**) (*operaciones lógicas y aritméticas (ALU)*), típicamente en forma binaria, de acuerdo con las instrucciones que se le da desde un programa.
- El **dispositivo de entrada (input Device)** se utiliza para proporcionar datos y señales de control a un sistema de procesamiento de información. (Ej: Teclado, mouse ..)
- El **dispositivo de salida (output unit)** es el dispositivo que recibe la información procesada. (Ej: pantalla, impresora...)

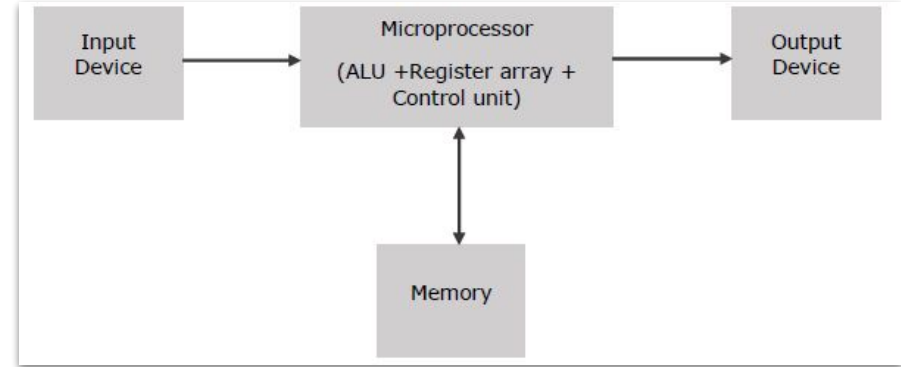
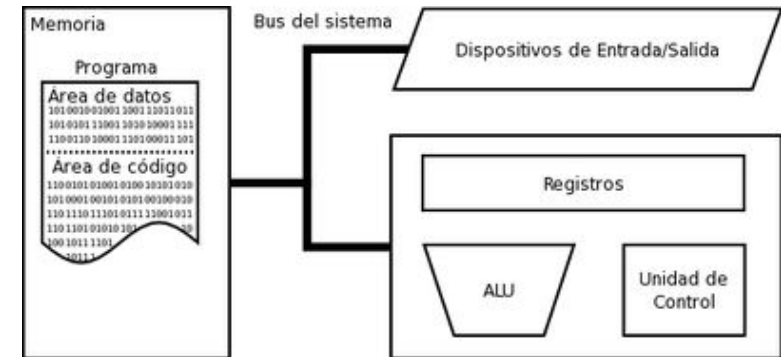
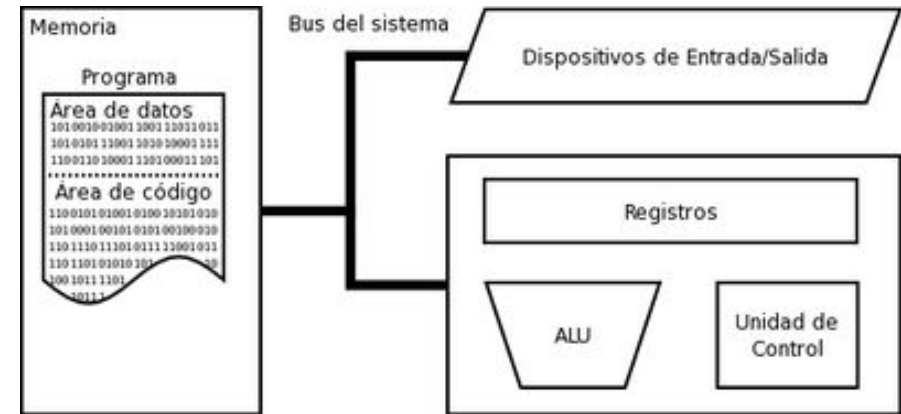


Diagrama de bloques de un computador básico (1)



Concepto computador básico

- Formada físicamente por numerosos circuitos integrados (Hardware) y logicamente por programas (software).
 - Hardware:** Se refiere a las partes físicas, tangibles, de un sistema informático; sus componentes eléctricos, electrónicos, electromecánicos y mecánicos. Ej: cpu, memoria, dispositivos de entrada, gabinete, pantalla, entre otros.
 - Software:** Se refiere a la parte intangible (lógica) del sistema informático. Incluye a los programas, datos, información, señales digitales para uso interno. Ej: sistema operativos, aplicaciones como navegador, procesador de texto, entre otros.

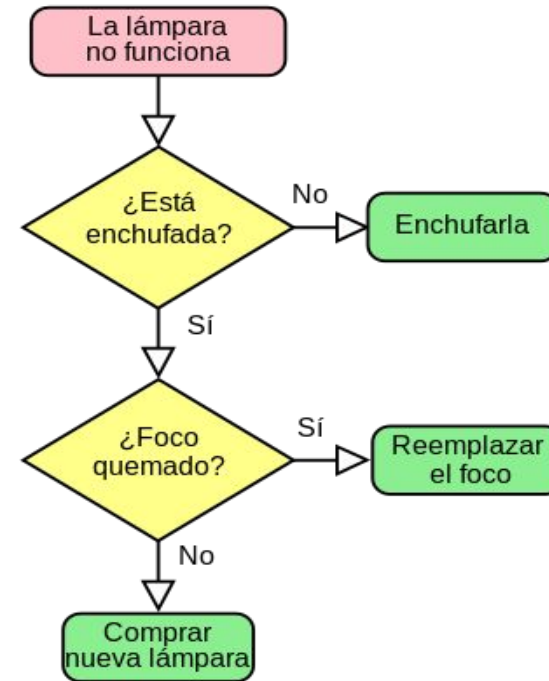


Algoritmo

Un algoritmo es una secuencia de pasos ordenada, finita y general para alcanzar un objetivo previamente establecido.

En nuestro contexto: el algoritmo describe la solución a un problema en términos de:

1. Los datos necesarios para representar la instancia del problema.
2. El conjunto de pasos necesarios para producir el resultado deseado.



Ejemplo de un algoritmo, en diagrama de flujo.

Programa

Un programa es la implementación de un algoritmo; por lo que está escrito en un lenguaje de programación y constituye un conjunto de instrucciones.

Un **lenguaje de programación** es un lenguaje formal que proporciona una serie de instrucciones que permiten a un programador escribir secuencias de órdenes a modo de controlar el comportamiento lógico de una computadora. Los símbolos del lenguaje de programación es texto legible por el humano.

```
def fib(n):  
    a, b = 0, 1  
    while a < n:  
        print(a, end=' ')  
        a, b = b, a+b  
    print()
```

Ejemplo de código de programación en lenguaje Python

El proceso de la creación de un programa (Programación)

La resolución de un problema mediante una computadora consiste en el proceso que a partir de la descripción de un problema, expresado habitualmente en lenguaje natural y en términos propios del dominio del problema, permite desarrollar un programa informático que resuelva dicho problema.

Para desarrollar un programa informático presentamos un método con una serie de pasos que facilitarán el proceso de la programación:

1. Identificar y comprender el problema (Análisis).
2. Planificar la solución (Diseño).
3. Codificar el programa (Implementación).
4. Probar el programa (Probar o Test).
5. Documentar el programa (Documentar).

El proceso de la creación de un programa (Programación)

Identificar y comprender el problema (Análisis): - Comprender lo que debe hacer el programa . -

Es el proceso de Análisis del problema, en donde se debe:

- **Identificar "Entrada":** Datos que son necesarios para resolver el problema. Son los datos que el usuario ingresa a la computadora antes de procesar.
- **Identificar "Salida":** Datos que debe proporcionar la resolución del problema. Son los datos que son necesarios mostrar en la salida de la computadora.
- **Definir el problema:** entender el dominio del problema. Hacer breves descripciones, si lo cree necesario, del tratamiento (*proceso*) al cual serán suscriptos los datos.

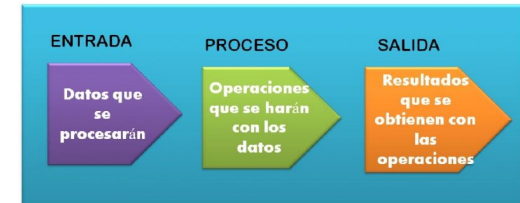


Diagrama Entrada Proceso Salida (EPS)

El proceso de la creación de un programa (Programación)

Planificar la solución (*Diseño*): - Descubrir cómo resolver el problema paso a paso. -

Se debe crear el algoritmo que dará la solución al problema planteado. Se establecen las especificaciones del programa, es decir, qué debe hacer y cómo lo debe hacer. Los algoritmos pueden ser expresados de muchas maneras, a través de **representaciones internas (mentales)** o a través de representaciones externas como ser: lenguaje natural, [pseudocódigo](#) o [diagrama de flujo](#)) y lenguajes de programación entre otros.

*Las descripciones en lenguaje natural tienden a ser ambiguas y extensas. El usar pseudocódigo y diagramas de flujo evita muchas ambigüedades del lenguaje natural. Dichas expresiones son formas más estructuradas para representar algoritmos; no obstante, se mantienen independientes de un lenguaje de programación específico. **Los programadores con experiencia utilizan directamente el lenguaje de programación para expresar el algoritmo. Nosotros adoptaremos esta metodología.***

El proceso de la creación de un programa (Programación)

Codificar el programa (*Implementación*): - Escribir la solución en un lenguaje de programación. -

El programador traduce la lógica del diagrama de flujo o pseudocódigo o de su representación mental a un lenguaje de programación. Para que el programa funcione, los programadores deben seguir exactamente las reglas y la sintaxis del lenguaje que están utilizando.

En nuestro caso utilizaremos el lenguaje de programación Python.

El proceso de la creación de un programa (Programación)

Probar el programa (*Probar o Test*): - Ejecutar el programa en la computadora y llevar adelante un conjunto de pruebas. -

- Probar si el programa funciona, esto quiere decir si "corre (ejecuta)". Si el programa no se ejecuta y el IDE ("*El software que nos ayuda a programar*") muestra un error, entonces estamos ante un problema de sintaxis.
- Si el programa ejecuta, entonces debemos probar con un conjunto amplio de datos de entrada y verificar que los datos de salida (*para dichos datos de entrada*) sean los esperados. Si los datos de salida no son los esperados (es decir son incorrectos) entonces estamos ante un error del tipo lógico.

Tipos de errores:

1. **Error de sintaxis:** Es cualquier violación a las reglas del lenguaje de programación. Es decir se escriben instrucciones que no pertenecen al lenguaje.
2. **Error de ejecución:** No tiene error de sintaxis. Se produce un error (*y el programa deja de ejecutar*) cuando el programa no puede ejecutar alguna instrucción de forma correcta en tiempo donde se está ejecutando. Típico ejemplo es la división por cero.
3. **Error lógico:** No tiene error de sintaxis ni error de ejecución. Se produce un error lógico cuando el programa no hace lo que se espera que haga. De manera que, un error de lógica se produce cuando dado un conjunto de datos de entradas, los resultados obtenidos (*datos de salida*) no son los esperados, es decir el resultado es incorrecto.

El proceso de la creación de un programa (Programación)

Documentar el programa:

Este paso se realiza en conjunto con "Codificar el programa". Escriba explicaciones del programa para que usted u otra persona entienda los códigos del programa. Las explicaciones pueden hacerse tanto a nivel global como a nivel de línea de instrucción (más detallado). Los lenguajes de programación proveen un mecanismo de comentarios que son utilizados para documentar.

Además, el código que escribe debe tener característica de autodocumentado. Esto se refiere que las declaraciones de variables, funciones utilice nombre que se relacione con el uso que se le dará.

Sistemas de numeración

Un sistema de numeración es un conjunto de símbolos y reglas que permiten representar datos numéricos. Los sistemas de numeración actuales son sistemas posicionales, que se caracterizan porque ***un símbolo tiene distinto valor según la posición que ocupa en la cifra.***

Sistemas de numeración

Sistema de numeración decimal:

El sistema de numeración que utilizamos habitualmente es el **decimal**, que se compone de diez símbolos o dígitos (0, 1, 2, 3, 4, 5, 6, 7, 8 y 9) a los que otorga un valor **dependiendo de la posición** que ocupen en la cifra: unidades, decenas, centenas, millares, etc.

En el sistema decimal el número **528**, por ejemplo, significa:

5 centenas + 2 decenas + 8 unidades, es decir:

$$5 \cdot 10^2 + 2 \cdot 10^1 + 8 \cdot 10^0 \text{ o, lo que es lo mismo:}$$

$$500 + 20 + 8 = 528$$

Sistemas de numeración

Sistema de numeración binario.

El sistema de numeración binario utiliza sólo dos dígitos, el **cero** (0) y el **uno** (1).

En una cifra binaria, cada dígito tiene distinto valor dependiendo de la posición que ocupe. El valor de cada posición es el de una potencia de **base 2**, elevada a un exponente igual a la posición del dígito menos uno. Se puede observar que, tal y como ocurría con el sistema decimal, la base de la potencia coincide con la cantidad de dígitos utilizados (2) para representar los números.

De acuerdo con estas reglas, el número binario **1011** tiene un valor que se calcula así:

$$1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0, \text{ es decir:}$$

$$8 + 0 + 2 + 1 = 11$$

y para expresar que ambas cifras describen la misma cantidad lo escribimos así:

$$1011_2 = 11_{10}$$

Sistemas de numeración

Conversión de números decimales y binarios

Para convertir un número decimal al sistema binario hay que realizar **divisiones sucesivas por 2** y escribir los restos obtenidos en cada división **en orden inverso** al que han sido obtenidos. Por ejemplo, para convertir al sistema binario el número 77_{10} haremos una serie de divisiones que arrojarán los restos siguientes:

$$77 : 2 = 38 \text{ Resto: } 1$$

$$38 : 2 = 19 \text{ Resto: } 0$$

$$19 : 2 = 9 \text{ Resto: } 1$$

$$9 : 2 = 4 \text{ Resto: } 1$$

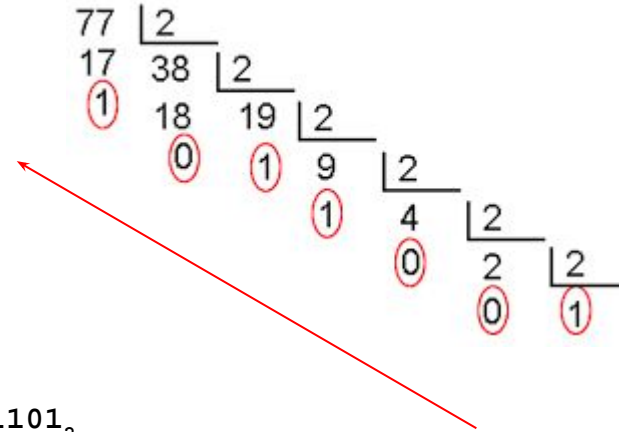
$$4 : 2 = 2 \text{ Resto: } 0$$

$$2 : 2 = 1 \text{ Resto: } 0$$

$$1 : 2 = 0 \text{ Resto: } 1$$

y, tomando los restos en orden inverso obtenemos la cifra binaria:

$$77_{10} = 1001101_2$$



Sistemas de numeración

Conversión de binarios a decimal

El proceso para convertir un número del sistema binario al decimal consta de desarrollar el número, teniendo en cuenta el valor de cada dígito en su posición, que es el de una potencia de 2, cuyo exponente es 0 en el bit situado más a la derecha, y se incrementa en una unidad según vamos avanzando posiciones hacia la izquierda.

Por ejemplo, para convertir el número binario 1010011_2 a decimal, lo desarrollamos teniendo en cuenta el valor de cada bit:

$$1*2^6 + 0*2^5 + 1*2^4 + 0*2^3 + 0*2^2 + 1*2^1 + 1*2^0 = 83$$

$$1010011_2 = 83_{10}$$

Sistema de numeración

Sistema de numeración octal y sistema Hexadecimal

Ver artículo: <http://platea.pntic.mec.es/~lgonzale/tic/binarios/numeracion.html>

FIN