

## **Conceptos básicos de computación y algoritmos**

Emilio Rojas Badillo

Instituto Tecnológico de Querétaro

Fundamentos de Programación

María Luisa Montes Almanza

24 de agosto de 2025

## **1.1 Conceptos básicos**

### **1.1.1 Definición de computadora, hardware y software**

Una computadora es un dispositivo electrónico programable capaz de recibir, procesar, almacenar y transmitir información. Realiza operaciones a través de instrucciones de programas que permiten ejecutar diversas tareas. El hardware se refiere a todos los componentes físicos y tangibles de la computadora, como el procesador, la memoria RAM, el disco duro, entre otros. En cambio, el software es el conjunto de programas, instrucciones y reglas informáticas que gestionan el hardware y permiten al usuario realizar funciones específicas, como sistemas operativos y aplicaciones de usuario.

### **1.1.2 Elementos funcionales de una computadora**

Unidad Central de Procesamiento (CPU): Considerada el cerebro del sistema, la CPU ejecuta instrucciones, controla y coordina todas las operaciones internas, y realiza cálculos lógicos y aritméticos. Incluye la unidad de control y la unidad aritmético-lógica (ALU).

Memoria: Se divide en memoria principal o RAM, donde se almacenan temporalmente las instrucciones y datos en uso, y memoria secundaria o almacenamiento masivo, donde los datos se guardan de forma permanente, como discos duros y memorias USB.

Dispositivos de entrada: Permiten al usuario introducir datos en la computadora, como teclado, mouse y escáner.

Dispositivos de salida: Proporcionan información procesada al usuario, como el monitor, impresora y altavoces.

Dispositivos de almacenamiento: Incluyen hardware dedicado al almacenamiento de datos, como discos duros, unidades SSD y memorias USB.

### **1.1.3 Clasificación del software**

Software de sistema: Gestiona y controla el hardware, proporcionando una plataforma para que el software de aplicación opere. Ejemplos clásicos son los sistemas operativos como Windows, Linux y macOS.

Software de aplicación: Programas diseñados para realizar tareas específicas para el usuario, como procesadores de texto, navegadores web, hojas de cálculo y juegos.

Software de desarrollo: Herramientas utilizadas por programadores para crear, depurar y mantener software, incluyendo compiladores, editores de código, y entornos de desarrollo integrados (IDEs).

### **1.1.4 Clasificación del hardware**

Dispositivos de entrada: Teclado, mouse, escáner, cámara.

Dispositivos de salida: Monitor, impresora, altavoces.

Dispositivos de almacenamiento: Disco duro, unidad de estado sólido (SSD), memorias USB.

Unidad de procesamiento: CPU y GPU, que realizan la ejecución de instrucciones y procesamiento gráfico respectivamente.

## **1.2 Representación de algoritmos: gráfica y pseudocódigo**

### **1.2.1 Definición de algoritmo, diagrama de flujo y pseudocódigo**

Un algoritmo es un conjunto finito de pasos o instrucciones ordenadas y definidas que permiten resolver un problema o realizar una tarea específica. Estas instrucciones deben ser claras, precisas, finitas y ejecutarse en un orden determinado para obtener un resultado concreto a partir de una entrada (input).

El diagrama de flujo es una representación gráfica que ilustra la secuencia de pasos de un algoritmo mediante formas geométricas estándar conectadas por flechas que muestran el orden de ejecución. Facilita la visualización del proceso y su lógica antes de codificarlo. El pseudocódigo es una forma de representar un algoritmo mediante un lenguaje semiestructurado que simula instrucciones de programación en un formato cercano al lenguaje humano, para facilitar su comprensión y traducción a cualquier lenguaje de programación .

### **1.2.2 Características principales de los algoritmos (Propiedades)**

Finitud: Tiene un número limitado de pasos para terminar.

Precisión: Instrucciones claras y sin ambigüedad.

Entrada: Recibe uno o más datos iniciales para procesar.

Salida: Produce al menos un resultado o solución.

Orden: Los pasos se deben ejecutar en secuencia lógica.

Generalidad: Debe servir para resolver problemas de una clase o tipo.

### **1.2.3 Principales símbolos de los diagramas de flujo**

Óvalo: inicio y fin del proceso.

Rectángulo: instrucción o proceso a realizar.

Paralelogramo: entrada o salida de datos.

Rombo: decisión o condición que determina el flujo según respuestas (sí/no).

Flechas: indican la dirección del flujo del proceso .

### **1.2.4 Reglas para elaborar diagramas de flujo**

- Debe tener un solo punto de inicio y un solo punto de fin.
- Las instrucciones deben estar dentro de símbolos adecuados.
- El flujo debe ser claro y seguir flechas sin cruces innecesarios.
- Debe evitar ambigüedades, decisiones claras y rutas definidas.
- Simplificar el diagrama evitando exceso de detalles sin importancia .

### **1.2.5 Reglas para elaborar pseudocódigos**

- Lenguaje claro y estructurado, cercano al humano.
- Uso de palabras clave (INICIO, FIN, SI, MIENTRAS, PARA).
- Identación para bloques de código.
- Evitar ambigüedad.
- Incluir todas las operaciones importantes para facilitar la codificación .

## **1.3 Diseño de algoritmos**

### **1.3.1 Definición de Identificador, Variables, Constantes y Operador**

Identificador: Es el nombre simbólico que se asigna a elementos del programa como variables, constantes, funciones o procedimientos para referenciarlos. Deben ser únicos y seguir ciertas reglas para ser válidos.

Variable: Es un espacio en memoria que puede almacenar datos que pueden cambiar durante la ejecución del programa. Se les asigna un identificador para interactuar con el valor que contienen.

Constante: Es un valor fijo que no cambia durante la ejecución del programa. Al igual que las variables, tienen un identificador y representan información permanente.

Operador: Son símbolos o palabras que indican una operación a realizar sobre uno o más operandos, como suma, resta, asignación, comparación, etc..

### **1.3.2 Reglas para nombrar identificadores**

Deben comenzar con una letra (A-Z o a-z) o un guion bajo (\_) pero no con un número.

Solo pueden contener letras, números y guiones bajos.

No deben coincidir con palabras reservadas del lenguaje de programación.

No deben contener espacios ni caracteres especiales.

Ser descriptivos para facilitar la comprensión del código, pero breves.

### **1.3.3 Tipos de datos**

Numéricos: Enteros (int), reales o decimales (float, double).

Caracteres: Caracteres individuales (char) o cadenas de texto (string).

Booleanos: Variables que solo pueden tener dos valores: verdadero (true) o falso (false).

Otros: Tipos avanzados como arreglos, listas, objetos, dependiendo del lenguaje.

### **1.3.4 Tipos de operadores**

- Aritméticos: +, -, \*, /, %.

- Relacionales: ==, !=, <, >, <=, >=.

- Lógicos: AND, OR, NOT.

- Asignación: =, +=, -=.
- Otros: [información faltante].

### **1.3.5 Jerarquía de operadores**

Paréntesis () — se evalúan primero.

Exponenciación y radicación (si aplica).

Multiplicación (\*), división (/), módulo (%).

Suma (+) y resta (-).

Operadores relacionales.

Operadores lógicos (NOT, AND, OR).

Asignación.

### **1.3.6 Estructuras de control y su clasificación**

Las estructuras de control son mecanismos que permiten modificar el flujo secuencial de un algoritmo según condiciones o repeticiones. Se clasifican en:

Estructuras de selección o decisión: permiten ejecutar código condicionalmente.

Ejemplos: IF... ELSE, SWITCH.

Estructuras de repetición o bucles: permiten ejecutar un bloque de código varias veces. Ejemplos: WHILE, FOR, DO... WHILE. Estructuras de salto: desvían el flujo del programa a otro punto, como BREAK y CONTINUE.



## Referencias

Equipo editorial, Etecé. (2025, 20 agosto). Algoritmo en informática - Concepto, partes y ejemplos. Concepto.

<https://concepto.de/algoritmo-en-informatica/>

Formación, A. (2024, 20 septiembre). ¿Qué es un operador en programación?

[https://www.adrformacion.com/knowledge/programacion/\\_que\\_es\\_un\\_operador\\_en\\_programacion\\_.html](https://www.adrformacion.com/knowledge/programacion/_que_es_un_operador_en_programacion_.html)

Cuevas, C. (2012, 16 abril). Unidades funcionales de una computadora. Boccaccicuevas.

<https://boccaccicuevas.wordpress.com/2012/04/16/unidades-funcionales-de-una-computadora/>

Centro de Escritura Javeriano. (2020). Normas APA, séptima edición. Pontificia

Universidad Javeriana, seccional Cali.

<https://www2.javerianacali.edu.co/centro-escritura/recursos/manual-de-normas-apa-septima-edicion#gsc.tab=0%C2%A0>