

DEFINICIÓN FORMAL DE ALGORITMO

El **pseudocódigo** (del griego *pseudo*: falso y *código*) es una descripción de alto nivel de un algoritmo computacional que utiliza las convenciones estructurales de un lenguaje de programación real, pero está diseñado para la **lectura humana** en lugar de la lectura por máquina.

En términos sencillos: es escribir lógica de programación usando palabras de nuestro idioma, ignorando las reglas rígidas de "sintaxis" (como los puntos y comas o paréntesis complicados) para enfocarnos exclusivamente en la **solución lógica**.

📖 **Thomas H. Cormen (y el grupo CLRS)**: En su libro *"Introduction to Algorithms"* (considerado la biblia de la algoritmia), utilizan un pseudocódigo muy refinado. Su enfoque dicta que el pseudocódigo debe ser lo suficientemente claro para que cualquier programador lo traduzca a cualquier lenguaje (C++, Java, Python).

📖 **Donald Knuth**: El padre del análisis de algoritmos. En su obra monumental *"The Art of Computer Programming"*, aunque usa lenguajes de bajo nivel, defiende la descripción paso a paso de los procesos antes de la implementación.

📖 **Steven Skiena**: Autor de *"The Algorithm Design Manual"*, quien enfatiza el pseudocódigo como la herramienta principal para la comunicación entre ingenieros.

ALGORÍTMICA: CONCEPTOS FUNDAMENTALES

1 ¿Qué es un algoritmo?

Concepto general

Un **algoritmo** es una secuencia **finita, ordenada y no ambigua** de pasos que permiten resolver un problema o realizar una tarea específica.

Definiciones de autores

- ◆ **Niklaus Wirth (1976)**

Un algoritmo es un procedimiento bien definido que toma uno o más valores como entrada y produce uno o más valores como salida.

- ◆ **Thomas H. Cormen et al. (2009)**

Un algoritmo es cualquier procedimiento computacional bien definido que toma un valor o conjunto de valores como entrada y produce un valor o conjunto de valores como salida.

- ◆ **David Harel (2004)**

Un algoritmo es una descripción precisa de una estrategia de resolución de problemas, expresada de manera que pueda ser ejecutada mecánicamente.

Características de un algoritmo

Todo algoritmo debe cumplir con:

1. **Finitud:** debe terminar en un número limitado de pasos.
2. **Precisión:** cada paso debe ser claro y exacto.
3. **Entrada:** recibe uno o más datos.
4. **Salida:** produce al menos un resultado.
5. **Efectividad:** cada operación debe ser realizable.

2 ¿Qué es la algorítmica?

Concepto

La **algorítmica** es la rama de la informática que estudia el **diseño, análisis y representación** de algoritmos para la resolución de problemas.

Autor

- ◆ **Brassard y Bratley (1997)**

La algorítmica se ocupa del estudio sistemático de los métodos de resolución de problemas mediante procedimientos finitos y bien estructurados.

¿Qué es el pseudocódigo?

Concepto general

El **pseudocódigo** es una forma **intermedia de representación** de algoritmos que utiliza lenguaje natural estructurado, independiente de cualquier lenguaje de programación.

Definiciones de autores

- ◆ **Joyanes Aguilar (2008)**

El pseudocódigo es una herramienta que permite expresar algoritmos mediante una mezcla de lenguaje natural y estructuras lógicas de programación.

- ◆ **G. M. Schneider & J. L. Gersting (2010)**

El pseudocódigo describe algoritmos usando una notación cercana al lenguaje humano, facilitando su comprensión sin depender de la sintaxis de un lenguaje específico.

◆ **Luis Joyanes Aguilar (2013)**

El pseudocódigo facilita el diseño lógico de soluciones antes de su implementación en un lenguaje de programación.

Ventajas del pseudocódigo

- ✓ No depende de un lenguaje
- ✓ Facilita el análisis lógico
- ✓ Mejora la comprensión del problema
- ✓ Sirve como base para diagramas de flujo y programas

4 Relación entre algoritmo y pseudocódigo

Algoritmo

Es la solución lógica

Abstracto

Independiente del lenguaje

Puede expresarse de varias formas

Pseudocódigo

Es la representación escrita

Semi-formal

Independiente del lenguaje

Forma estructurada y clara

Todo pseudocódigo representa un algoritmo, pero no todo algoritmo está necesariamente escrito en pseudocódigo.

5 Tipos de algoritmos

5. 1 Según su estructura de control

◆ **Algoritmos secuenciales**

Ejecutan instrucciones una tras otra.

Autor: Deitel & Deitel

Los algoritmos secuenciales siguen una ejecución lineal sin bifurcaciones.

◆ **Algoritmos selectivos (condicionales)**

Utilizan decisiones lógicas.

Autor: Joyanes Aguilar

Permiten elegir entre diferentes caminos de ejecución según una condición.

- ◆ **Algoritmos repetitivos (iterativos)**

Repiten un conjunto de instrucciones.

Autor: Cormen et al.

Los algoritmos iterativos ejecutan bloques de instrucciones múltiples veces hasta que se cumple una condición.

5. **2** Según su forma de ejecución

- ◆ **Algoritmos determinísticos**

Siempre producen el mismo resultado con la misma entrada.

Autor: Knuth (1997)

- ◆ **Algoritmos no determinísticos**

Pueden producir resultados diferentes.

5. **3** Según su finalidad

- ◆ Algoritmos de búsqueda
- ◆ Algoritmos de ordenamiento
- ◆ Algoritmos matemáticos
- ◆ Algoritmos de control
- ◆ Algoritmos de procesamiento de datos

6 Importancia del pseudocódigo en la enseñanza

✦ En formación informática:

- Desarrolla **pensamiento lógico**
- Evita errores de sintaxis
- Facilita el paso a cualquier lenguaje
- Es clave en **análisis de sistemas**

Autor: Louden (2015)

El pseudocódigo permite al estudiante concentrarse en la lógica del problema sin distraerse con detalles sintácticos.

7 Frase docente para cerrar la clase

*Un buen programador escribe código;
un buen analista diseña algoritmos;
y un buen informático domina el pseudocódigo.*

EJEMPLO DE PSEUDOCODIGO.

EJEMPLO 1:

Promedio y clasificación de notas

Enunciado

Leer las calificaciones de **N estudiantes**, calcular el **promedio general** y determinar cuántos estudiantes están **aprobados** y **reprobados**.

Pseudocódigo

Inicio

Leer N

suma \leftarrow 0

aprobados \leftarrow 0

reprobados \leftarrow 0

Para $i \leftarrow 1$ Hasta N Hacer

Leer nota

suma \leftarrow suma + nota

Si nota ≥ 10 Entonces

aprobados \leftarrow aprobados + 1

Sino

reprobados \leftarrow reprobados + 1

FinSi

FinPara

promedio \leftarrow suma / N

Escribir "Promedio general:", promedio

Escribir "Aprobados:", aprobados

Escribir "Reprobados:", reprobados

Fin

Conceptos aplicados:

- Ciclo Para
- Condicional Si
- Contador y acumulador

EJEMPLO 2:**Control de ventas diarias****Enunciado**

Registrar las ventas diarias de una tienda hasta que el usuario ingrese **0**, calcular el **total vendido**, el **promedio** y determinar la **venta mayor**.

Pseudocódigo

Inicio

total \leftarrow 0

contador \leftarrow 0

mayor \leftarrow 0

Repetir

Leer venta

Si venta \neq 0 Entonces

total \leftarrow total + venta

contador \leftarrow contador + 1

Si venta > mayor Entonces

mayor \leftarrow venta

FinSi

FinSi

Hasta Que venta = 0

promedio \leftarrow total / contador

Escribir "Total vendido:", total

Escribir "Promedio de ventas:", promedio

Escribir "Venta mayor:", mayor

Fin

Conceptos aplicados:

- Ciclo Repetir – Hasta Que

- Condición de parada
- Comparación de máximos

EJEMPLO 3:

Conteo de números pares e impares

Enunciado

Leer una cantidad **N de números**, contar cuántos son **pares** y cuántos **impares**, y mostrar los resultados.

Pseudocódigo

Inicio

Leer N

pares \leftarrow 0

impares \leftarrow 0

Para $i \leftarrow 1$ Hasta N Hacer

Leer num

Si num MOD 2 = 0 Entonces

pares \leftarrow pares + 1

Sino

impares \leftarrow impares + 1

FinSi

FinPara

Escribir "Cantidad de pares:", pares

Escribir "Cantidad de impares:", impares

Fin

Conceptos aplicados:

- Operador módulo
- Clasificación de datos
- Contadores