

Algoritmia Elemental

1.2. Eficiencia de los algoritmos

¿Qué es la eficiencia de un algoritmo?

La eficiencia mide qué tan bien un algoritmo utiliza los recursos disponibles para resolver un problema.

Los dos principales recursos considerados son:

- Tiempo: Cantidad de operaciones necesarias para obtener el resultado.
- Espacio: Cantidad de memoria utilizada para almacenar datos temporales durante la ejecución.

Tipos de eficiencia

- Eficiencia temporal: Evalúa la rapidez del algoritmo para producir una solución.
- Eficiencia espacial: Evalúa cuánta memoria requiere el algoritmo para su ejecución.

Medición de eficiencia

Se utiliza la notación Big O para describir la eficiencia en función del tamaño de la entrada (n):

Ejemplos:

- $O(1)$: Tiempo constante.
- $O(\log n)$: Tiempo logarítmico.
- $O(n)$: Tiempo lineal.
- $O(n^2)$: Tiempo cuadrático.
- $O(2^n)$: Tiempo exponencial.

Esta notación permite comparar la eficiencia de diferentes algoritmos de manera formal y predecir su comportamiento para entradas de gran tamaño.

1.3. Caso medio

Definición

El caso medio analiza el comportamiento promedio de un algoritmo considerando entradas aleatorias.

Es una estimación de cuánto tiempo o espacio utilizará el algoritmo en circunstancias típicas, no extremas.

Relación con otros casos

- Mejor caso: El algoritmo se ejecuta en el menor tiempo posible.
- Peor caso: El algoritmo tarda el mayor tiempo posible.
- Caso medio: El algoritmo tiene un tiempo de ejecución promedio considerando todas las posibles entradas.

Ejemplo

Búsqueda lineal en una lista de n elementos:

- Mejor caso: El elemento está en la primera posición (1 intento).
- Peor caso: El elemento está en la última posición o no existe (n intentos).
- Caso medio: En promedio se necesitan $n/2$ intentos para encontrar el elemento.

Importancia del caso medio

El análisis del caso medio es crucial para entender cómo se comportará el algoritmo en situaciones reales, ya que ni siempre se presentan los mejores ni los peores casos.