



# *Programación II*

# *Análisis de Costos*

*2C 2023 TN – Ing. Elizabeth Barrera*



# *Temas*

👑 *Eficiencia*

👑 *Costos*

👑 *Clasificación*

👑 *Ejemplos*

👑 *Aclaraciones*



# *Análisis de Costos*

# Eficiencia

Una característica esencial de la *calidad de software* es el uso eficiente de los recursos. Uno de los principales recursos de que disponemos es el *tiempo* de uso del procesador. Otro recurso importante es la cantidad de *memoria* disponible.

# Costos

*Cuando queremos analizar o comparar diferentes implementaciones necesitamos algún parámetro o variables en la cual basar esa comparación, esos parámetros pueden ser: cantidad de operaciones llevadas a cabo o utilizadas, cantidad de memoria utilizada.*

*En el primer caso hablamos de análisis de **complejidad temporal**, y en el segundo de **complejidad espacial**.*



# Clasificación

Para el análisis de costos de **de complejidad temporal** se consideran tres criterios a través de los cuales clasificaremos el costo de ejecución de un método:

- *Costo constante (C)*: el costo **no depende de la cantidad de elementos de entrada. Es el mejor de los costos posibles.**

# Clasificación

- *Costo lineal (L)*: el costo depende linealmente de la cantidad de elementos de entrada, por cada elemento de entrada hay una cantidad constante de operaciones que se ejecutan. Es un costo bueno.
- *Costo polinómico (P)*: el costo es mayor a lineal. Un ejemplo sería el lineal anidado, el cuadrático. Es un costo no tan bueno, pero tratable.

# Ejemplos

- *Costo constante* (operaciones elementales):
  - operaciones aritméticas
  - evaluación de condiciones booleanas
  - accesos directos a memoria (por ejemplo, a una posición de un arreglo)
  - asignaciones de valores a variables



```
for (int i=0; i < n; i++) {  
    /* bloque de operaciones  
       elementales */  
}
```

# Ejemplos

- *Costo lineal:*

Un ciclo dentro del cual se posee una secuencia de operaciones elementales. Estas poseen un costo constante, pero se repite varias veces, por lo que el costo total del ciclo termina siendo lineal.

```
for (int i=0; i < n; i++) {  
    for (int j=0; j < n; j++) {  
        /* bloque de  
        operaciones elementales */  
    }  
}
```

# Ejemplos

- *Costo polinómico:*

Un ciclo anidado, en el cual el ciclo interno posee una complejidad lineal, pero el mismo se repite varias veces durante la ejecución del ciclo externo (por cada iteración tenemos una vuelta, la cual se repite linealmente), con lo cual se multiplican las operaciones. Estos ciclos anidados tienen costo cuadrático (un tipo de costo polinómico).

# Aclaraciones

- En este curso haremos el *análisis de complejidad temporal*, realizando un análisis simplificado.
- La eficiencia de un algoritmo es uno de los factores principales para **determinar la velocidad con que un programa se ejecuta**. Es incluso más importante que el hardware subyacente.



*¡Muchas Gracias!*



# *Bibliografía*

- 👑 *Programación II – Apuntes de  
Cátedra – V1.3 – Cuadrado  
Trutner – UADE*
- 👑 *Programación II – Apuntes de  
Cátedra – Wehbe – UADE*