

Tarea # 1

Algoritmos y Estructura de Dato Avanzados / Ingeniería Civil Informática
Sistemas de Información

Universidad del Bío-Bío

Prof: Rodrigo Torres A.

Otoño 2025

Problema # 1. Búsqueda Galopante o Búsqueda Exponencial. Sea A un arreglo de enteros ordenados de menor a mayor. El algoritmo $BusquedaBinaria(A, li, ls, x)$ permite verificar si $x \in A$ en tiempo $O(\log_2 n)$. La Búsqueda Galopante (BG) consiste de dos etapas. En la primera se determina un rango donde la clave x debería encontrarse (si es que $x \in A$); y en la segunda se realiza una búsqueda binaria en el rango determinado en la primera etapa (ver Algorithm 1). El límite inferior ($linf$) del rango corresponde a la mayor posición (o mayor valor del índice de A) tal que $A[linf] < x$ y que es potencia de 2 (ver líneas 6-8 de Algoritmo 1).

Algorithm 1 Algoritmo Búsqueda Galopante o Exponencial

```
1:  $BusquedaGalopante(A, size, x)$ 
2: if ( $size = 0$ ) then
3:   return -1
4: else
5:    $linf = 1$ 
6:   while ( $linf * 2 < size$  and  $A[2 * linf] < x$ ) do
7:      $linf = linf * 2$ 
8:   end while
9:   return  $BusquedaBinaria(A, linf, size, x)$ 
10: end if
```

- 1) Implemente en su lenguaje favorito la Búsqueda Binaria y la Búsqueda Galopante
- 2) Genere al azar un arreglo A de tamaño 5.000.000.000 de enteros. El rango de los elementos de A es $[1, 10.000.000]$. Ordene el arreglo A (puede usar Merge Sort, aproveche

de implementarlo). Luego genere al azar las cantidades de claves: 1.000, 5.000, 10.000, 50.000 y 100.000 y complete la tabla siguiente:

# de claves	Búsqueda Binaria Tiempo (ms)	Búsqueda Galopante Tiempo (ms)
1.000		
5.000		
10.000		
50.000		
100.000		

- 3) De acuerdo a los datos de la tabla anterior, ¿Cuál algoritmo es mejor ?.
- 4) Jusifique haciendo un estudio teórico de ambos algoritmos.

Problema # 2. Multiplicación de Matrices.

Sean dos matrices, A y B , de dimensiones $n \times n$. La matriz $C = A \times B$ también es una matriz de $n \times n$ cuyo elemento (i, j) se forma multiplicando cada elemento de la i -ésima fila de A por el elemento correspondiente de la j -ésima columna de B y sumando los productos parciales:

$$C_{ij} = \sum_{k=1}^n A_{ik} B_{kj}$$

El cálculo de cada elemento C_{ij} requiere de n multiplicaciones. La matriz C tiene n^2 elementos, así que el tiempo total del algoritmo es $O(n^3)$. El algoritmo anterior, que llamaremos *algoritmo tradicional*, se desprende directamente de la definición de la multiplicación de matrices.

Sin embargo, la multiplicación de 2 matrices cuadradas puede resolverse de las siguientes maneras:

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

1) Propiedad 1:

$$\begin{aligned} C_{11} &= A_{11} \cdot B_{11} + A_{12} \cdot B_{21} \\ C_{12} &= A_{11} \cdot B_{12} + A_{12} \cdot B_{22} \\ C_{21} &= A_{21} \cdot B_{11} + A_{22} \cdot B_{21} \\ C_{22} &= A_{21} \cdot B_{12} + A_{22} \cdot B_{22} \end{aligned}$$

2) Propiedad 2:

$$\begin{aligned} M &= (A_{11} + A_{22})(B_{11} + B_{22}) \\ N &= (A_{21} + A_{22})B_{11} \\ O &= A_{11}(B_{12} - B_{22}) \\ P &= A_{22}(B_{21} - B_{11}) \\ Q &= (A_{11} + A_{12})B_{22} \\ R &= (A_{21} - A_{11})(B_{11} + B_{12}) \\ S &= (A_{12} - A_{22})(B_{21} + B_{22}) \end{aligned}$$

Posteriormente se calculan las submatrices C_{ij} :

$$\begin{aligned} C_{11} &= M + P - Q + S \\ C_{12} &= O + Q \\ C_{21} &= N + P \\ C_{22} &= M + O - N + R \end{aligned}$$

Se pide:

- 1) Demostrar qué técnica de diseño se puede ocupar, dadas las propiedades 1) y 2).
- 2) Usando las ideas anteriores, generar al azar las matrices A y B (considere matrices de enteros) y completar la siguiente tabla con los tiempos de ejecución¹. DR1 usa la primera propiedad, y DR2 usa la segunda (programelos en el lenguaje que estime conveniente).

n	Tiempos		
	Algoritmo Tradicional	DR1	DR2
32			
64			
128			
256			
512			
1024			
2048			
4096			

3) Obtenga al menos dos conclusiones, respecto del rendimiento de los algoritmos.

4) Haga un estudio de comportamiento asintótico de los 2 algoritmos que creó.

Fecha de Entrega: 22 de Mayo de 2025. **Grupos:** 2 o 3 estudiantes

¹Elija la unidad de medida de tiempo que mejor se acomode (milisegundos, segundos, minutos, etc.)