
Práctico – Análisis de Algoritmos

1. Para cada uno de los siguientes fragmentos de pseudocódigo realice:
 - a. Brinde el análisis teórico de tiempo de ejecución Big O
 - b. Implemente en código Java y saque tiempos de ejecución para distintos valores de N, pruebe con valores grandes.
 - c. Compare el análisis teórico con el empírico ¿ se respetaron las cotas superiores ?

Caso 1

```
sum = 0;

for( i = 0; i < n; i++ )

    sum++;
```

Caso 2

```
sum = 0;

for( i = 0; i < n; i++ )

    for( j = 0; j < n; j++ )

        sum++;
```

Caso 3

```
sum = 0;

for( i = 0; i < n; i++ )

    for( j = 0; j < n * n; j++ )

        sum++;
```

Caso 4

```
sum = 0;

for( i = 0; i < n; i++ )
```

```
for( j = 0; j < i; j++ )  
  
    sum++;
```

Caso 5

```
sum=0;  
  
for( i = 0; i < n; i++ )  
  
    for( j = 0; j < i * i; j++ )  
  
        for( k = 0; k < j; k++ )  
  
            sum++;
```

2. Un algoritmo toma 0.5 milisegundos para una entrada de 100 elementos. Cuanto tiempo tardaría si la entrada fuera de 500 considerando que las funciones de Big O son las siguientes:
 - a. N
 - b. $O(N \log N)$
 - c. N^2
 - d. N^3
3. Un algoritmo toma 0.5 milisegundos para una entrada de 100 elementos. Cual sería la cantidad máxima de elementos que podría procesar si tuviera disponible 1 minuto de ejecución, considerando que las funciones de Big O del algoritmo son las siguientes:
 - a. N
 - b. $O(N \log N)$
 - c. N^2
 - d. N^3
4. Para el algoritmo de ordenamiento Burbuja, Selección y Heap Sort deduzca el Orden Big O.
5. Deduzca un algoritmo eficiente desde el punto de vista del tiempo de ejecución para dado una secuencia de números indicar la suma mínima positiva consecutiva. Por ejemplo para $\{3, 1, -3, 4, 5, -2, -1\}$ una de las secuencias mínima es $\{3, 1, -3\}$. Realice una pequeña investigación para ver las distintas soluciones posibles.