

Análisis de algoritmos

Complejidad

1. Para cada uno de los siguientes fragmentos de programas, haga lo siguiente:
 - a. Dé un análisis en notación O del tiempo de ejecución.
 - b. Implemente el código y ejecútelo para diferentes valores N
 - c. Compare su análisis con los tiempos de ejecución reales

// Fragmento #1

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

//Fragmento #2

```
for( int i = 0; i < n; i++)  
    for( int j = 0; j < n; j++)  
        suma++;
```

// Fragmento #3

```
for( int i = 0; i < n; i++)  
    suma++;  
for( int j = 0; j < n; j++)  
    suma++;
```

//Fragmento #4

```
for( int i = 0; i < n; i++)  
    for( int j = 0; j < n*n; j++)  
        suma++;
```

//Fragmento #5

```
for( int i = 0; i < n; i++)  
    for( int j = 0; j < i; j++)  
        suma++;
```

//Fragmento #6

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

2. Determine, para los algoritmos típicos que utiliza para realizar cálculos a mano, el tiempo de ejecución en hacer lo siguiente:
 - a. Sumar dos enteros de N dígitos
 - b. Multiplicar dos enteros de N dígitos
 - c. Dividir dos enteros de N dígitos

Análisis de algoritmos

Complejidad

3. En términos de N , ¿cuál es el tiempo de ejecución del siguiente algoritmo para calcular X^N ?

```
Public static double potencia( double x, int n){  
    double resultado = 1.0;  
  
    for( int i = 0; i<n; i++)  
        resultado *= x;  
    return resultado;  
}
```

4. Escriba un algoritmo eficiente para determinar si existe un entero i tal que $A_j = i$ ($0 \leq j < n$) en un array de enteros ordenados de forma creciente. ¿Cuál es su tiempo de ejecución?
5. Un número primo no tiene divisores aparte de 1 y de él mismo. Se pide escribir un programa para determinar si un entero positivo N es primo. En términos de N . ¿cuál es el tiempo de ejecución en el caso peor de su programa?