

TC2017 Análisis y Diseño de Algoritmos

Tar03 – Complejidad Computacional

Ing. Luis Humberto González G

Forma de Trabajo: Individual.

Nombre: _____

Matricula: _____

Entrega un reporte de solución por escrito a cada uno de los siguientes problemas.

1. (40 puntos) Analizar cada uno de los siguientes segmentos de pseudocódigo, e indicar cuál es la complejidad del algoritmo representado. Suponer siempre que la “instrucción” es una operación básica.

CASO A:

```
for (j = 1; j <= n ; j = j * 2)
    instrucción;
```

CASO B:

```
for (j = 1; j <= n ; j++)
    for (k = 0; k < n/5; k++)
        instrucción;
```

CASO C:

```
for (j = 1; j <= n ; j++)
{
    k = n;
    while (k >= 1)
    {
        instrucción;
        k /= 2;
    }
}
```

CASO D:

```
j = n;
while (j > 0)
{
    for (k = j; k <= n; k = k*2)
        instrucción;
    j /= 2;
}
```

2. (20 puntos) A continuación se muestran 2 casos en los que para cada uno se muestran 2 algoritmos. Ambos algoritmos en cada caso, sirven para resolver el mismo problema.

- Identifica cuál es el problema que está resolviéndose en cada caso.
- Haz un análisis de la complejidad de cada algoritmo.
- Indica para cada caso, cuál es el algoritmo que más convendría elegir para la implementación de la solución al problema, justificando tu respuesta.

CASO A:

Algoritmo A1:

```
s = 0;
for (int x = 1; x <= n; x++)
    for (int y = 1; y <= n; y++)
        if (x == y) then
            s += a[x,y];
```

Algoritmo A2:

```
s = 0;
for (int j = 1; j <= n; j++)
    s += a[j,j];
```

CASO B:**Algoritmo B1:**

```
r = 1;  
for (int j=1; j≤n; j++)  
    r = r * x
```

Algoritmo B2:

```
r = 1;  
while (n > 0)  
{ if (n % 2 != 0) r *= x;  
  x *= x;  
  n /= 2;  
}
```

3. (40 puntos) Desarrolla un algoritmo que dado un arreglo **A** que contiene **n** distintos enteros positivos, genere un arreglo de dos dimensiones **B**, en donde en la posición **B[i][j] = B[j][i] = A[i]+A[i+1]+...+A[j-1]+A[j]**. Se calificará eficiencia.