TC2017 – Análisis y Diseño de Algoritmos ***Tarea #3***
Ing. Luis Humberto González G *Fecha de Entrega: 27-Ago-2013*
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) do
    {  instrucción;
       k = k / 2;
    }
}
```

CASO D:

```
 j = n;
while j > 0  do
   {  for ( k = j; k<=n; k = k*2)
          instrucción;
      j = 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.
   a) Identifica cuál es el problema que está resolviéndose en cada caso.
   b) Haz un análisis de la complejidad de cada algoritmo.
   c) 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 x = 1 to n do
  for y = 1 to n do
     if x = y then
          s = s + a[x,y];
```

Algoritmo A2:

```
s = 0;
for j = 1 to n do
   s = s + a[j,j];
```

CASO B:

Algoritmo B1:

```
r = 1;
fo(int  j=1; j<=n; j++)
   r = r *  x
```

Algoritmo B2:

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

3. *(40 puntos)* Desarrolla un algoritmo que dado un arrelo **A** que contiene **n** distintos enteros positivos, generé 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.