



ANÁLISIS Y DISEÑO DE ALGORITMOS

Tarea 2

Mejor y peor escenario y escenario promedio.

Profesora

Ana Belem Juárez Méndez

Instrucciones Resuelve los siguientes ejercicios.

Ejercicio I. Las siguientes son funciones típicas para expresar complejidad de algoritmos, coloca en una gráfica estas funciones para que observes cómo es su crecimiento. Realiza la gráfica en digital.

Orden de complejidad	Graficar	Nombre
n^0	1	constante
$\log n$	$\log_2 n$	Logaritmica
n	n	Lineal
$n \log n$	$n \log_2 n$	N logaritmica
n^a	n^2	Polinomial
a^n	2^n	Exponencial
$n!$	$n!$	Factorial

Ejercicio II. Describe cuál es el **peor** y el **mejor** escenario del siguiente algoritmo y calcula el costo y tiempo de esos escenarios de cada instrucción. También coloca su función.

Algoritmo de ordenación burbuja mejorado.

```
burbuja_mejorado(A, n)
  for i=0 to i<n-1
    bandera=0
    for j=0 to j<n-i-1
      if (A[j]>A[j+1])
        bandera=1
        AUX=A[j]
        A[j]=A[j+1]
        A[j+1]=AUX
    if (bandera==0)
      return
```

Ejercicio III. Describe cuál es el **peor** y el **mejor** escenario del siguiente algoritmo y calcula el costo y tiempo de esos escenarios de cada instrucción. Además calcula el escenario **promedio**, indicando cuántas y cuáles son las instancias que contemplaste. También coloca la función de cada escenario.

Algoritmo que indica si una palabra es un palíndromo o no.

```
palindromo(palabra, n)
    bandera=1
    i=0
    while(i<(n/2) and bandera==1)
        if(palabra[i]!=palabra[n-1-i])
            bandera=0
    if(bandera==1)
        return 1
    else
        return 0
```

Ejercicio IV. Describe cuál es el **peor** y el **mejor** escenario del siguiente algoritmo y calcula el costo y tiempo de esos escenarios de cada instrucción. También coloca su función.

Algoritmo que revisa si encuentra dos números tales que uno sea múltiplo del otro.

```
algoritmo(A, n)
    merge_sort(A,n)
    for i=0 to n-1
        j=i+1
        while(j<n)
            if(A[i]%A[j]==0)
                return 1
            j++
    return 0
```