

## TAREA 1. ANALISIS DE ALGORITMOS

1. Codificar el algoritmo de **ordenamiento por inserción (INSERTION-SORT)**, y calcular el tiempo de ejecución para diez valores distintos de  $n$ . Realizar lo anterior para un arreglo ordenado en orden creciente (mejor caso), ordenado en forma decreciente (peor caso) y un arreglo aleatorio (caso promedio). Para cada caso calcular los tiempos y graficarlos.
2. Codificar el algoritmo de **ordenamiento por selección (SELECTION-SORT)** y calcular los tiempo de cómputo para diez valores distintos de  $n$ . Realizar lo anterior para un arreglo ordenado en orden creciente (mejor caso), ordenado en forma decreciente (peor caso) y un arreglo aleatorio (caso promedio). Para cada caso calcular los tiempos y graficarlos.
3. Codifique el algoritmo de **ordenamiento por confluencia (MERGE-SORT)** y calcular el tiempo de ejecución para diez valores distintos de  $n$ . Realizar lo anterior para un arreglo ordenado en orden creciente (mejor caso), ordenado en forma decreciente (peor caso) y un arreglo aleatorio (caso promedio). Para cada caso calcular los tiempos y graficarlos.
4. Determinar la función  $T(n)$ , en el mejor y el peor de los casos, del algoritmo de **ordenamiento por selección**.
5. Realizar la graficación de las siguientes funciones:  $f(n) = \log(n)$ ,  $f(n) = n$ ,  $f(n) = n \cdot \log(n)$ ,  $f(n) = n^2$ ,  $f(n) = n^3$ . Todas las curvas deben ser colocadas en la misma gráfica.

## SELECTION\_SORT (A)

```
for  $i \leftarrow 1$  to  $n-1$  do
   $\text{min } j \leftarrow i$ ;
   $\text{min } x \leftarrow A[i]$ 
  for  $j \leftarrow i + 1$  to  $n$  do
    If  $A[j] < \text{min } x$  then
       $\text{min } j \leftarrow j$ 
       $\text{min } x \leftarrow A[j]$ 
   $A[\text{min } j] \leftarrow A[i]$ 
   $A[i] \leftarrow \text{min } x$ 
```