



Instituto Politécnico Nacional Escuela Superior de Cómputo



Análisis de Algoritmos

Práctica 3: Divide y Vencerás: Algoritmo MergeSort.

Profesor: Dr. Benjamín Luna Benoso.

Grupo: _____

Semestre 2018-1

1. Implementar el algoritmo MergeSort.

- i) Mediante gráficas, muestre que el algoritmo Merge tiene complejidad lineal.
- ii) Demuestre analíticamente que el algoritmo Merge tiene complejidad lineal.
- iii) Mediante gráficas, muestre que el algoritmo MergeSort tiene complejidad $\Theta(n \log n)$.
- iv) Demuestre analíticamente que el algoritmo MergeSort tiene complejidad $\Theta(n \log n)$.

Calcular el orden de complejidad de los siguientes algoritmos en el mejor (Ω) y en el peor de los casos (O) (no es necesario hacer el análisis línea por línea, en este caso, pueden aplicar propiedades de los algoritmos vistos en clase):

A.

Funcion1(n par)

- 1. $i = 0$
- 2. mientras $i < n$ hacer
- 3. para $j = 1$ hasta $j = 10$ hacer
- 4. Accion(i)
- 5. $j++$;
- 6. $i++ = 2$;

Suponga que Accion $\in \Theta(1)$.

B.

Funcion2($A[0, \dots, n-1]$, x *entero*)

1. **for** $i = 0$ **to** $i < n$ **do**
2. **if** ($A[i] < x$)
3. $A[i] = \min(A[0, \dots, n-1])$
4. **else if** ($A[i] > x$)
5. $A[i] = \max(A[0, \dots, n-1])$
6. **else**
7. **exit**