

Análisis y Diseño de Algoritmos

Ayudantía

Prof. Gilberto Gutiérrez
Ayudante: Israel Gajardo

Primavera 2018

Problema 1 Asuma un arreglo S de enteros de tamaño n y el siguiente algoritmo ($MinMax()$) para calcular el valor mínimo y el valor máximo almacenados en el arreglo.

```
1: MinMax( $S$ )
2: if  $\|S\| = 1$  then
3:   return  $(a, a)$  {  $S = \{a\}$  }
4: else if  $\|S\| = 2$  then
5:   return  $(\min(a, b), \max(a, b))$  {  $S = \{a, b\}$  }
6: else
7:   Dividir  $S$  en dos subconjuntos  $S_1, S_2$  de aproximadamente el mismo tamaño
8:    $(n_1, m_1) = \text{MinMax}(S_1)$ 
9:    $(n_2, m_2) = \text{MinMax}(S_2)$ 
10:  return  $(\min(n_1, n_2), \max(m_1, m_2))$ 
11: end if
```

1. Asumiendo como operación relevante la “comparación” entre enteros, defina la ecuación de recurrencia en el peor de los casos para el algoritmo $MinMax()$.

Problema 2 El algoritmo $MinMax()$ anterior divide el conjunto inicial en dos subconjuntos; S_1 y S_2 :

1. Diseñe un algoritmo, $MinMax3()$, que divida el conjunto en tres subconjuntos, S_1, S_2 y S_3 .
2. Plantee la ecuación de recurrencia de $MinMax3()$ para el peor caso (cantidad de comparaciones).
3. De manera intuitiva, ¿Cuál de los dos algoritmos cree que es mejor?. ¿Por qué?
4. El algoritmo clásico ($MinMaxIterativo()$) para resolver este mismo problema consiste en tomar el primer elemento de S como el mínimo (\min) y el máximo (\max) a la vez, y luego con cada elemento restante de S mejorar la solución, es decir, los valores \min y \max . Calcule la complejidad de $MinMaxIterativo()$.
5. De manera intuitiva, ¿Cuál de los tres algoritmos cree que es mejor?. ¿Por qué?

Para responder con certeza las preguntas 3 y 5 necesitamos resolver las ecuaciones de recurrencias. Lo veremos en el curso.

Problema 3 Suponga un arreglo de enteros de tamaño n el cual se encuentra ordenado de menor a mayor. Se necesita saber si existen dos números A y B en el arreglo tal que $B = -A$.

1. Diseñe un algoritmo $O(n^2)$ (peor caso considerando como operación relevante la comparación entre enteros) para encontrar A y B .
2. Demuestre que su algoritmo es $O(n^2)$
3. Especifique un algoritmo de complejidad menor que $O(n^2)$ para verificar si existen A y B .
4. Plantee la ecuación de recurrencia de su algoritmo.