

Algoritmia y Complejidad, Laboratorio 3

Diego Quan, UFM

August 16, 2018

1 Problema 1: Avoid Self-destruction

2 Problema 2: Quicksort

2.1 Same Elements

Quicksort tiene un worse case running time de $O(n^2)$ debido a que si el arreglo esta casi totalmente ordenado o que contenga todos los valores iguales, este algoritmo deberá recorrer todo el arreglo más de una vez. Este tendrá que realizar múltiples comparaciones.

2.2 Partitioning Visualization

Las siguientes imágenes muestran como transcurre el algoritmo de **Partitioning**. El elemento del arreglo en color rojo es el *pivot*, mientras que los de color azul son los elementos que son menores al pivot. Finalmente los verdes son los mayores al pivot.

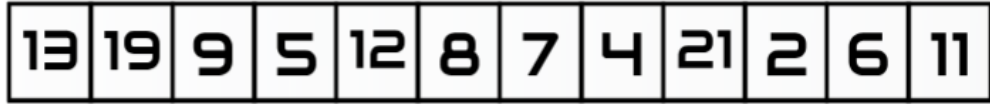


Figure 1: Arreglo sin partición

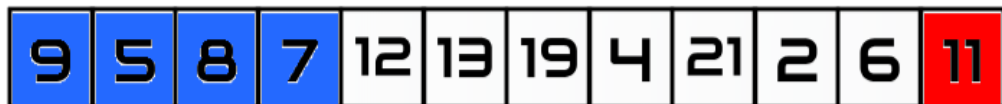
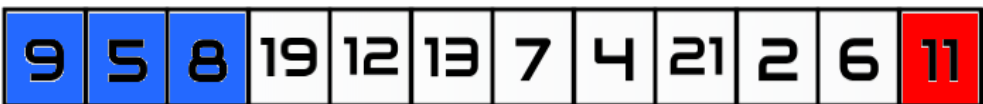
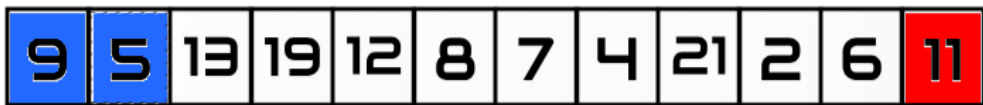
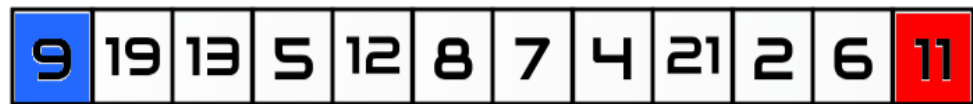




Figure 2: Arreglo particionado

3 Problema 3: "Self"-Partitioning

El código que se encuentra en el archivo **Quicksort.py** realiza la partición del arreglo a través de validaciones de left (lado izquierdo del pivote) y right (lado derecho del pivote). Posteriormente realiza los cambios que sean necesarios para llevar a cabo el *Quicksort*.