Javier Falcón 20165-5265

Ejercicio 1:

1. Dado un arreglo numérico A que contiene $2\mathrm{n}$ elementos, obtenga 2 subarreglos, T_1 y T_2, de tamaño n cada uno, para los cuales se cumple que la suma de todos los elementos de T_1 y la suma de todos los elementos de T_2 son lo más cercanas posibles la una de la otra. Dicho de otra manera, sean $\mathrm{T}_1 \subseteq \mathrm{A}$ y $\mathrm{T}_2 \subseteq \mathrm{A}$, el resultado de:

$$\sum_{i=1}^{n} T_{1_i} - \sum_{i=1}^{n} T_{2_i}$$

   debe acercarse lo más que se pueda a 0, siendo $\mathrm{n} > 0$.

2. Algoritmo que concibe lo planteado en el inciso 1:

   Entrada: Arreglo $\mathrm{A}[1 \dots 2n]$, $arreglo\ vacío\ T_1[1 \dots n]$, $arreglo\ vacío\ T_2[1 \dots n]$.
   Salida: **Arreglo** T_1 $con\ subarreglo\ de\ A$, $arreglo$ T_2 $con\ subarreglo\ de\ A$.

   Procedimiento CreatePartition$(\mathrm{A}, \mathrm{T}_1, \mathrm{T}_2)$
     $A \leftarrow sort(A)$ $\quad \rhd$ *Arreglo ordenado en orden decreciente*
     $n \leftarrow siza(A)$
     $sum_1 \leftarrow 0$
     $sum_2 \leftarrow 0$
     $j \leftarrow 0$
     $k \leftarrow 0$
     para $i \leftarrow 1$ ***hasta*** n ***hacer***
       si $sum_1 < sum_2$ ***entonces***
         $\mathrm{T}_1[j] \leftarrow A[i]$
         $sum_1 \leftarrow sum_1 + A[i]$
         $j \leftarrow j + 1$
       si no
         $T_2[k] \leftarrow A[i]$
         $sum_2 \leftarrow sum_2 + A[i]$
         $k \leftarrow k + 1$
       fin si
     fin para
   fin procedimiento

3. El algoritmo tiene un comportamiento de $n + nlog\, n$ debido a que el método de ordenamiento utilizado, en promedio, consume $O(nlog\, n)$. Dicho esto, una vez ordenado el arreglo, se hace una iteración $2\mathrm{n}$ veces, lo que nos arrojaría un

comportamiento de $O(n)$. Finalmente, teniendo $O(nlog\,n) + O(n)$, nuestro resultado sería $O(nlog\,n)$ por propiedades de la suma entre órdenes de complejidad.