Ejercicio 4

Rojo Mata Daniel

- Algoritmo: mediana
- Entrada: Dos arreglos no ordenados de números enteros y de longitud n cada uno.
- La mediana de los valores

```
1  public static int mediana(int[] arreglo1, int[] arreglo2){
2          int[] arreglo_dimension_2n = Practica_cuatro.arreglo_2n(
3          ejercicio4.bubbleSort(arreglo1),
4          ejercicio4.bubbleSort(arreglo2));
5
6          int longitud = arreglo_dimension_2n.length;
7
8          return arreglo_dimension_2n[longitud/2];
9      }
```

Descripción del algoritmo

Como primer instancia se ordenan los elementos de los arreglos de entrada (*arreglo1* y *arreglo2*) con el método *bubble Sort*, una vez ordenados éstos, se procede a juntar los arreglos con un método llamado *arreglo-2n*[1] el cual retorna un arreglo cuyas entradas son los elementos de los arreglos iniciales ordenados en orden creciente, este último arreglo recibe el nombre de *arreglo-dimension-2n*.

Lo anterior está expresado en las líneas 2, 3 y 4 del código anterior [2]

[1] Este método fue el implementado en la realización de la práctica 4 para resolver el problema dado, véase la figura 1

[2] En realidad la ejecución se realiza en una línea, sin embargo, por cuestiones de este editor de textos se escribe en más líneas para que pueda permanecer dentro de la página el texto, no es un error de sintaxis.

En la línea 6 se define la variable *longitud* como el largo de *arreglo-dimension-2n* mientras que en la línea 8 se accede al elemento de éste que tenga el índice *longitud/2*, teniendo así al elemento que se encuentra a la mitad.

Obtención de $T(n)$

Se hace uso de los tiempos ya calculados para *bubbleSort* y para $arreglo-2n$. Para el primero, la obtención de $T(n)$ se hizo en el salón de clases, mientras que para el segundo la obtención se encuentra en el anexo, los valores de $T(n)$ para estos algoritmos son los siguientes:

$$T_{bubble}(n) = 8n^2 + 12n + 2 \tag{1}$$

$$T_{arreglo-2n}(n) = 28n + 9 \tag{2}$$

En esencia el método *bubble* se aplica dos veces, pues hay dos arreglos como entrada, así que, por la linealidad de $O(f(n))$, hay dos contribuciones de T_{bubble}. Ahora, después de que se aplica *bubble* se aplica el método *arreglo-2n*, por lo que hay una contribución de $T_{arreglo-2n}$.

Al definir *longitud* en la linea 6 se tiene una operación elemental, y en la línea 8 se tendrían dos operaciones elementales, acceder al elemento dado y la división que se realiza.

De esta forma, se tiene:

$$T(n) = 2T_{bubble} + T_{arreglo-2n} + 1 + 2 = 2\left(8n^2 + 12n + 2\right) + 28n + 9 + 3$$

$$T(n) = 16n^2 + 24n + 4 + 28n + 12 = 16n^2 + 52n + 16$$

$$T(n) = 16n^2 + 52n + 16$$

Demostración de la complejidad

Afirmación 0.1. *El algoritmo mediana tiene complejidad en tiempo* $O(n^2)$.

Demostración. Por demostrar: $16n^2 + 52n + 16 \in O(n^2)$, es decir, veamos que existen $c, n \in Z^+$ tales que $16n^2 + 52n + 16 \leq c \cdot n^2$ para toda $n_0 \leq n$.

Sea $n_0 = 1$, es así que $n \geq 1$ por lo que se cumplen las siguientes desigualdades.

$$16n^2 \geq 16n^2$$

$$52n^2 \geq 52n$$

$$16n^2 \geq 16$$

Estas expresiones se cumplen, recuérdese, porque se tomó $n_0 = 1$ y entonces $n \geq 1$. Sumando las expresiones anteriores se tiene el siguiente resultado.

$$16n^2 + 52n^2 + 16n^2 \geq 16n^2 + 52n + 16$$

$$84n^2 \geq 16n^2 + 52n + 16$$

$$16n^2 + 52n + 16 \leq 84n^2$$

$$T(n) \leq 84n^2$$

De esta manera, existen $c = 84$ y $n_0 = 1$, tal que $T(n) \leq 84n^2$ para todo $n_0 \geq n$.
Así pues, es cierto que $T(n) \in O(n^2)$.

□

Anexos

Figura 1: Obtención de $T_{arreglo-2n}(n)$