



## Algoritmos y Estructuras de Datos – Grupo 3

### Parcial 2

1. (3 puntos) Asuma que cuenta con un conjunto de puntos bidimensionales (x,y) almacenados en una estructura lineal (vector, lista, etc.). Debe implementar el ordenamiento de estos puntos por la distancia desde un punto dado (a,b) hasta el punto, de mayor a menor.

Debe utilizar un método de ordenamiento no implementado en clase (no puede ser ninguno de los métodos con complejidad cuadrática, shellsort, quicksort o heapsort), sino dejado de tarea. Si usa uno de los dados en clase, tendrá una deducción de 0.5 puntos.

Ejemplo: Distancia del conjunto de puntos, ordenados de mayor a menos con respecto al punto (12,6)

X	Y	Distancia
1	6	11
6	1	7.81025
9	13	7.615773
14	13	7.28011
6	2	7.211103
11	13	7.071068
14	12	6.324555
6	6	6
14	2	4.472136
8	5	4.123106
9	6	3
10	7	2.236068
12	8	2
14	6	2
12	7	1

(0.5) – Ejecuta sin errores

(0.25) – Define estructuras de datos, genera datos de entrada y muestra por pantalla la lista original

(0.25) – Función de cálculo de la distancia

(0.5) – Función de comparación

(1.0) – Método de ordenamiento

(0.5) – Ordena los datos y los muestra por pantalla



2. (0.5 punto) Calcule la complejidad  $T(N)$  y el  $O(N)$  del siguiente algoritmo. Puede usar las notas de clase para sustentar su respuesta.

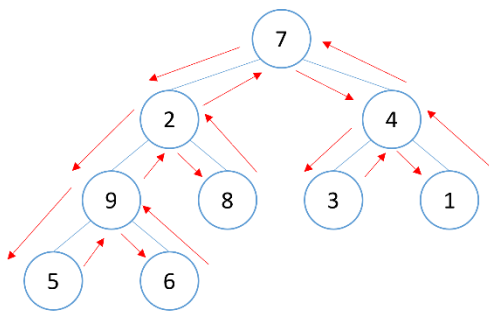
```
for (i = 1; i < n; ++i)
    for (j = 1; j < i * i; ++j)
        if (j % i == 0)
            for (k = 0; k < j; ++k)
                ++sum;
```

(0.3) – Defina y exprese de la manera más sencilla, la sumatoria que representa el  $T(N)$

(0.2) – Defina el  $O()$  de la función  $T(N)$

3. (1.5 puntos) Dado un vector de números enteros, asuma que almacena una estructura de árbol similar a la usada en el algoritmo HeapSort, sin que deban estar ordenados. Diseñe e implemente una función recursiva que permita mostrar los elementos de la siguiente manera:

7	2	4	9	8	3	1	5	6
---	---	---	---	---	---	---	---	---



Orden de escritura en pantalla:

5, 6, 9, 8, 2, 3, 1, 4, 7

Reglas:

- Si el nodo tiene hijos, se exploran los hijos que tenga, primero el izquierdo y luego el derecho.
- Una vez se hayan explorado ambos hijos, se escribe el dato del nodo.
- Si el nodo es hoja, se escribe el dato.

(0.5) – Identifique y explique con sus palabras el caso base del algoritmo recursivo

(1.0) – Implemente el algoritmo en C++ mostrando el vector de prueba y el resultado de su recorrido