

**Instituto**

**Politécnico**

**Nacional**

**Escuela Superior de Cómputo**

**REPORTE DE LA SESIÓN 2 ALGORITMOS BÁSICOS**

**Materia:**

Análisis de algoritmos

**Grupo:**

3CM3

**Integrantes:**

Castro Cruces Jorge Eduardo

**Fecha:**

Viernes, marzo 20, 2020Instrucciones

Ejercicio1:

* Ingresar al tamaño del arreglo A
* Introducir los elementos al arreglo
* Ingresar al tamaño del arreglo B
* Introducir los elementos al arreglo

Ejercicio2:

* Compilar y ejecutar el archivo

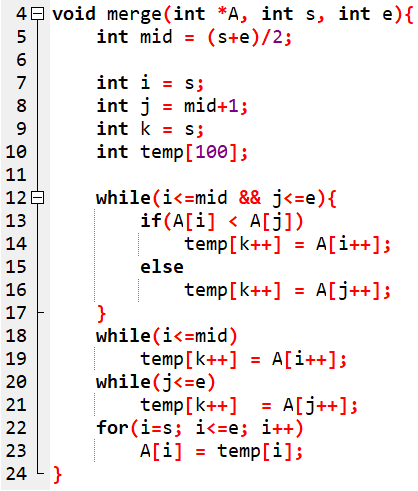
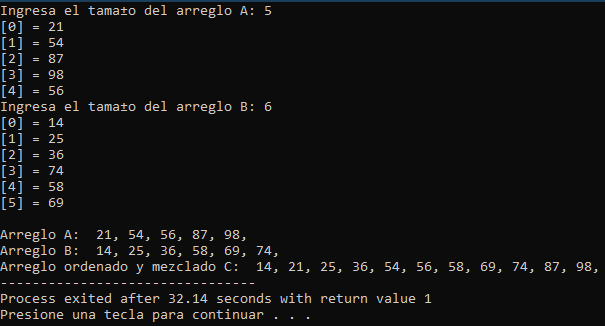
Ejercicio3:

* Ingresar el valor de x
* Ingresar el valor de y

Ejercicio4:

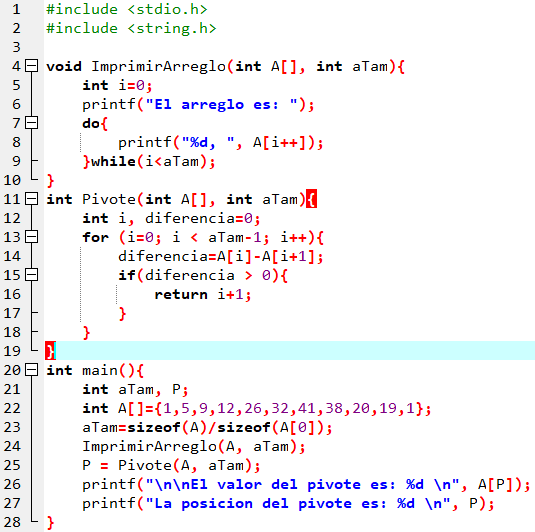
* Ingresar el numero de puntos que contiene el archivo .txt

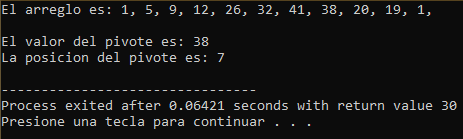
Ejercicio1



Ejercicio 2

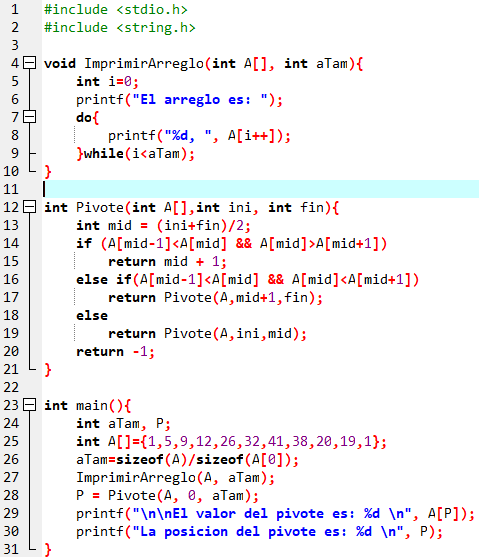
Inciso a)

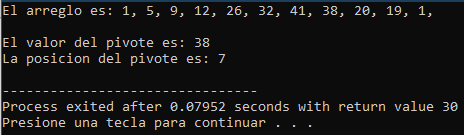
Complejidad: O(n)



En este ejercicio almacenamos un arreglo de tamaño n, y mediante fuerza bruta vamos recorriendo una a una las posiciones del arreglo, hasta hallar p.

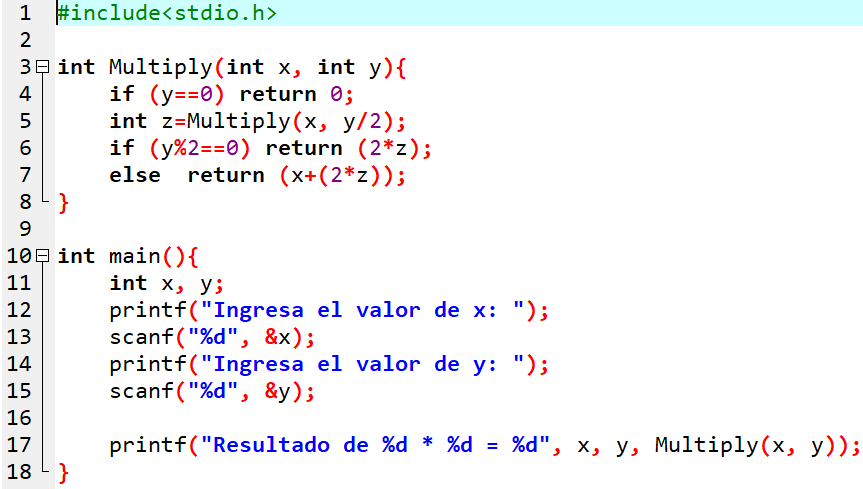
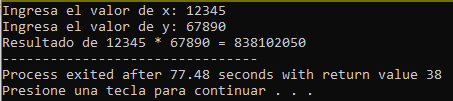
Inciso b)

Complejidad: O(Logn)



La única diferencia con esta nueva función es que, aplica el teorema de divide y veceras, lo que conlleva una disminución del tiempo de ejecución, ya que estamos aplicado la recursividad a nuestro favor. La primera sentencia nos permite corroborar A[mid-1]<A[mid]<A[mid+1] y en caso de que esto sea verdad, retornamos el valor mid+1, la siguiente sentencia nos dice si se encuentra a la derecha del punto medio y en caso de que así sea, de forma recursiva nos desplazamos a la derecha del arreglo, ahora que en caso contrario, nos estaremos moviendo a la derecha igualmente, pero desde el inicio del arreglo. Finalmente si no se cumple ninguna de las anteriores condiciones, retorna -1.

Ejercicio 3



Ejercicio 4

