

Objetivo:

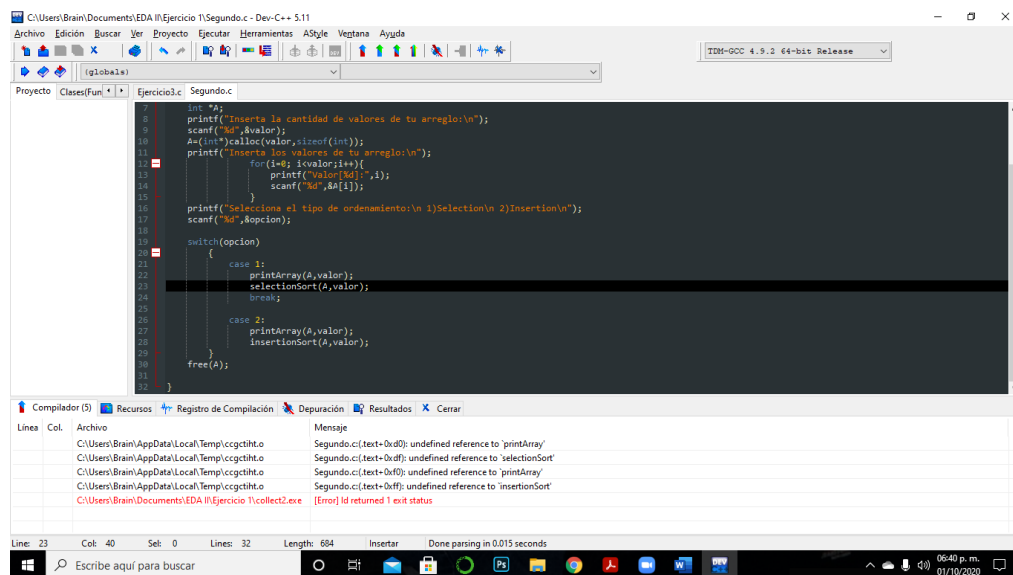
El alumno revisará el uso de bibliotecas como capas de abstracción así como realización de pruebas de los algoritmos para diferentes instancias de colecciones para conocer la complejidad de estos

Primer Ejercicio:

El primer ejercicio consiste en analizar las funciones y los programas, en cuanto a los programas de Utilidades se entiende que son simples funciones que ayudan a intercambiar valores, imprimir valores de un arreglo, y mostrar las modificaciones realizadas en un arreglo en base a los ajustes que se realicen, de acuerdo al algoritmo utilizado.

En cuanto a los programas de ordenamiento se refieren a los programas de Selection e Insertion, los cuales utilizan los valores de un arreglo y al mismo tiempo utilizan las funciones de utilidad para ir implementando los cambios al arreglo y mostrarlo.

Segundo Ejercicio:



```
int main()
{
    printf("Inserta la cantidad de valores de tu arreglo:\n");
    scanf("%d",&valor);
    A=(int*)calloc(valor,sizeof(int));
    printf("Inserta los valores de tu arreglo:\n");
    for(i=0; i<valor;i++){
        printf("Valor[%d]:",i);
        scanf("%d",&A[i]);
    }
    printf("Selecciona el tipo de ordenamiento:\n 1)Selection\n 2)Insertion\n");
    scanf("%d",&opcion);
    switch(opcion)
    {
        case 1:
            printArray(A,valor);
            selectionSort(A,valor);
            break;
        case 2:
            printArray(A,valor);
            insertionSort(A,valor);
            break;
    }
    free(A);
}
```

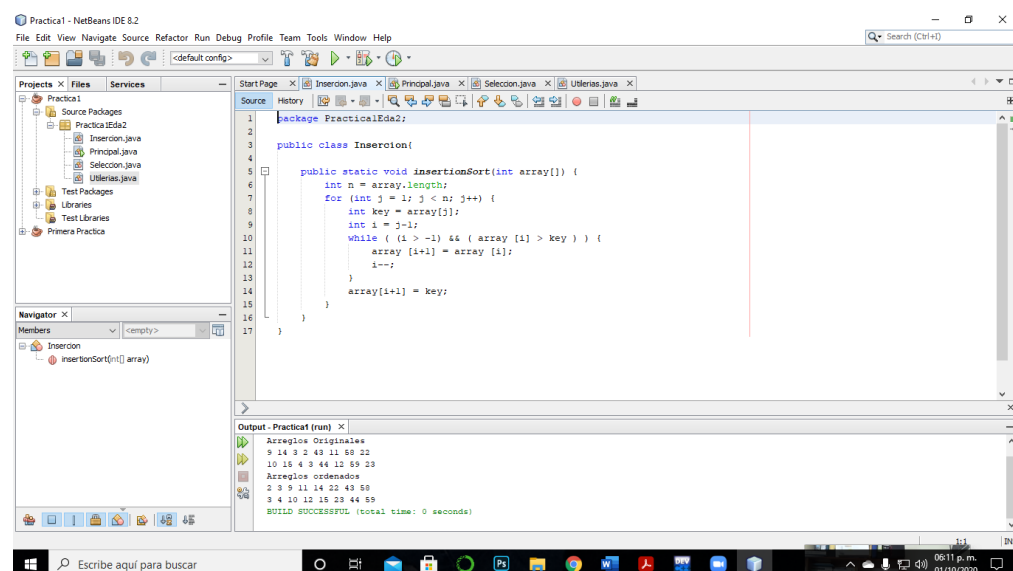
Compiler (5) | Recursos | Registro de Compilación | Depuración | Resultados | Cerrar

Linea	Col.	Archivo	Mensaje
		C:\Users\Brain\AppData\Local\Temp\ccgchtht.o	Segundo.c:(text+0x0): undefined reference to 'printArray'
		C:\Users\Brain\AppData\Local\Temp\ccgchtht.o	Segundo.c:(text+0x0f): undefined reference to 'selectionSort'
		C:\Users\Brain\AppData\Local\Temp\ccgchtht.o	Segundo.c:(text+0x1f): undefined reference to 'printArray'
		C:\Users\Brain\AppData\Local\Temp\ccgchtht.o	Segundo.c:(text+0x2f): undefined reference to 'insertionSort'
		C:\Users\Brain\Documents\EDA\I\Ejercicio 1\collect2.exe	[Error] ld returned 1 exit status

Line: 23 Col: 40 Sel: 0 Lines: 32 Length: 684 Insertar Done parsing in 0.015 seconds

Me quedé en esa parte porque en realidad ese error se repetía hiciera lo que hiciera, además que llevaba media hora buscando que tenía mal y pues no encontré cual es el error, así que espero encontrarlo con más calma fuera de clase.

Tercer Ejercicio:



```
package PracticalEda2;

public class Insercion {

    public static void insertionSort(int array[]) {
        int n = array.length;
        for (int j = 1; j < n; j++) {
            int key = array[j];
            int i = j-1;
            while ( (i > -1) && ( array[i] > key ) ) {
                array[i+1] = array[i];
                i--;
            }
            array[i+1] = key;
        }
    }
}
```

Output - Practical (run)

```
Arreglos Originales
9 14 3 2 49 11 58 22
10 15 4 3 44 12 69 23
Arreglos ordenados
2 3 9 11 14 22 49 58
9 4 10 12 15 23 44 59
BUILD SUCCESSFUL (total time: 0 seconds)
```

Dentro de los cuatro ejecutables de java, el de "Selection" e "Insertion", muestran el código correspondiente a cada tipo de algoritmo, de hecho son los mismo practicamente a los vistos en la clase.

En la parte "Principal", se encuentran los arreglos y su llenado, ademas de que se imprimen en su estado natural y posteriormente se llaman las funciones de "Selection y de "Insertion", para el acomodo de los elementos y al final se imprimen los arreglos ordenados, de acuerdo un orden ascedente.

En la parte "Utilidad", se encuentran las funciones de impresión de los arreglos, y la función de intercambiar que se utiliza en el algoritmo de "Selection", por otro lado es muy similar al ejercicio uno, pero este si corrió sin problemas.