Logotipo

Descripción generada automáticamente

# Método selección

# 

**Fecha:** 14/03/2023

**Materia:** Estructura de datos

**Profesor/Profesora:** Ismael López Buenrostro

**Nombre:** Luis Rigoberto Ramírez Llamas

**Registro:** 21100298

**Carrera:** Drllo. Software

**Grupo:** 4E1

**Descripción del programa**

Este programa es un código en C++ a través de visual studio, en donde se introduce un conjunto 1 a 50 de números enteros ya sean ordenados o desordenados, positivos o negativos con la finalidad de ordenarlos con el método de selección

En este código se usa una clase Métodos y una función Selección a través de métodos para ambas opciones tanto en mostrar al final como cada interacción

**Lista de requerimientos**

Fabricar un programa que realice el proceso de ordenación por el método de selección el cual deberá tener las siguientes características.

a) Ordenar un conjunto de números enteros, la cantidad mínima deberá ser 1 y el máximo 50.

b) Deberá ingresar los valores correspondientes.

c) Deberá tener dos opciones de ordenación (Elija como preguntar esta opción).

a. La primera opción después de ingresar los datos estos se muestran ordenados.

b. La segunda opción mostrara todas cada una de las iteraciones del método

d) El o los módulos de ordenación deberán ser métodos.

e) El sistema está diseñado en C++.

f) La ordenación será de menor a mayor

g) Mostrar el mensaje de error correspondiente cada posibilidad.

**Contenido del PSP**

|  |  |  |  |
| --- | --- | --- | --- |
| Código fuente: | Estimado  (min) | Real  (min) | Error |
| Tiempo de Diseño | 20 | 25 | 25% |
| Tiempo de Codificación. | 120 | 190 | 58% |
| Tiempo estimado de fabricación | 140 | 215 | 54% |
| Total, de líneas de código nuevas. | 60 | 47 | 22% |
| Total, de líneas de código reusadas. | 39 | 39 | 0% |
| Total, de líneas de código del programa. | 90 | 76 | 15% |
| Cantidad de errores de compilación. | 20 | 11 | 45% |
| Cantidad de errores de ejecución. | 5 | 1 | 80% |

**Código fuente**

#include <iostream>

using namespace std;

class Metodos

{

public:

void Seleccion(double Numeros[50], int Tam, char Respuesta);

};

void Metodos::Seleccion(double Numeros[50], int Tam, char Respuesta)

{

for (int i = Tam; i >= 0; i--)

{

int Intercambios = 0;

int NumMax = i;

double Aux;

for (int j = i - 1; j >= 0; j--)

{

if (Numeros[j] > Numeros[NumMax])

{

NumMax = j;

Intercambios++;

}

}

if (NumMax != i)

{

Aux = Numeros[i];

Numeros[i] = Numeros[NumMax];

Numeros[NumMax] = Aux;

}

if (Respuesta == 'S')

{

for (int i = 1; i <= Tam; i++)

{

cout << "\t" << Numeros[i];

}

cout << endl;

}

if (Intercambios == 0)

{

break;

}

}

if (Respuesta == 'N')

{

for (int i = 1; i <= Tam; i++)

{

cout << "\t" << Numeros[i];

}

}

};

int main()

{

int Tam;

char R;

cout << "Cuantos numeros vas a ingresar" << endl;

cin >> Tam;

if (Tam < 0 || Tam > 51)

{

cout << "Tamaño no compatible" << endl;

}

else

{

double Numeros[50];

for (int i = 1; i <= Tam; i++)

{

cout << "Introduce tus numeros: " << endl;

cin >> Numeros[i];

}

cout << "Desea ver el proceso de ordenamiento? \n[S] Si\n[N] No" << endl;

cin >> R;

Metodos Sel;

cout << "Estos son tus numeros ordenados : \n" << endl;

Sel.Seleccion(Numeros, Tam, R);

}

return 0;

};