

Universidad Mariano Gálvez de Guatemala

Carrera de Ingeniería en Sistemas – 5090

Curso: Programación I

Docente: Ing. Carlos Alejandro Arias

Laboratorio 4

Funciones y Bibliotecas

Pablo Javier Roldán Vásquez

Carné: 5090-23-13164

Fecha: 11/02/2023

Introducción

Este programa en C++ permite realizar operaciones matemáticas simples como suma, resta, multiplicación, división y módulo, de acuerdo con la elección del usuario. Utiliza funciones para realizar las operaciones y mostrar los resultados de manera clara en la pantalla. Además, permite al usuario realizar múltiples operaciones en una sola ejecución

Se declaran las bibliotecas

```
#include <iostream>
#include <cmath>
```

Función para las operaciones matemáticas

```
float operaciones(float a, float b, char operador) {
    switch (operador) { //Utiliza un switch para seleccionar la operación basada en el operador dado.
        case '+':
            return a + b;
        case '-':
            return a - b;
        case '*':
            return a * b;
        case '/':
            if (b != 0) {
                return a / b;
            } else {
                //cerr: es el flujo de error estándar que se utiliza para generar los errores.
                cerr << "Error: División por cero." << endl; //Muestra un mensaje de error si se intenta dividir por cero.
                return 0.0;
            }
        case '%':
            if (b != 0) {
                /* Toma dos valores flotantes (a y b), los convierte a enteros eliminando decimales,
                 realiza la operación de módulo entre ellos y devuelve el resultado de esa operación como un entero*/
                return static_cast<int>(a) % static_cast<int>(b);
            } else {
                cerr << "Error: Módulo por cero." << endl; //Muestra un mensaje de error si se intenta calcular el módulo por cero.
                return 0.0;
            }
        default: //Si el operador no es ninguno de los anteriores, muestra un mensaje de error.
            cerr << "Error: Operador no válido." << endl;
            return 0.0;
    }
}
```

Se muestran los resultados

```
//Función para imprimir resultados
void Resultados(float resultado) {
    system("cls"); //Limpia la pantalla
    cout << "El resultado es: " << resultado << endl; //Imprime el resultado de la operación.
    system("pause");
}
```

Se le piden los datos al usuario

```
int main() {
    //Declaración de variables
    int cantidadOperaciones;
    float num1, num2;
    char operador, continuar;
    system("cls");
    system("color 8e");

    do {
        //Solicitar al usuario la cantidad de operaciones a realizar
        cout << "Ingrese la cantidad de operaciones a realizar: ";
        cin >> cantidadOperaciones;

        //Bucle para realizar operaciones según la cantidad ingresada
        for (int i = 0; i < cantidadOperaciones; ++i) {
            system("cls"); //Limpia la pantalla
            //Solicita al usuario los números y operador para la operación actual
            cout << "Ingrese el primer numero " << i + 1 << ": ";
            cin >> num1;

            cout << "Ingrese el segundo numero " << i + 1 << ": ";
            cin >> num2;

            cout << "Ingrese el operador (+, -, *, /, %) para la operación " << i + 1 << ": ";
            cin >> operador;
        }
    } while (continuar == 's' || continuar == 'S');
```

Pregunta al usuario si desea realizar más operaciones

```
system("cls");  
cout << "Desea continuar (s/n)?" << endl;  
cin >> continuar;  
  
} while (continuar == 's' || continuar == 'S');  
  
system("cls"); //Limpia la pantalla  
//Cierre del programa  
cout << "Programa finalizado." << endl;
```

Programa funcionando

```
Ingrese el primer numero 2: 3  
Ingrese el segundo numero 2: 5  
Ingrese el operador (+, -, *, /, %) para la operaci|n 2: *
```

```
El resultado es: 15  
Presione una tecla para continuar . . .
```

```
Desea continuar (s/n)?  
n
```

```
Programa finalizado.  
-----
```

Conclusión

Este programa proporciona una herramienta eficiente y fácil de usar para realizar operaciones matemáticas básicas. Su estructura modular y su capacidad para manejar múltiples operaciones hacen que sea útil en diversas situaciones. Además, la incorporación de controles para evitar errores, como la división por cero, mejora la experiencia del usuario.

Link de github: <https://github.com/PabloRoldan2/Laboratorio-4.git>