



UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA

Curso: Programación, Jornada Matutina

Sección: "A"

Catedrática: Ing. Carlos Alejandro Arias.



Carnet: __5090-23-1762

Nombre completo: Roberto Carlos de León Gramajo

Fecha entrega: 21/02/2024

Tema tarea: Hoja de Trabajo#5

Introducción

Realizar 5 ciclos, y funciones diferentes, en donde ingresemos números diferentes por cada una, y se realice los trabajos de contar dígitos, sumar dígitos, ordenar números, encontrar la factorial de un número, y contar sumar los números pares e impares de un número.

```
#include <iostream>
```

```
#include <cmath>
```

```
#include <stdlib.h>
```

```
using namespace std;
```

```
// realizamos un función con retorno, para que ayude con la suma de los numeros pares o impares de los cuales se compone un número, como 5, que tiene 1,2,3,4,5, de los cuales 2 y 4, son pares, que dan 6 en su suma, o 1, 3, y 5 los cuales son impares, y su suma es 9
```

```
int sumaPI(int n, bool pares) {
```

```
    int suma = 0;
```

```
    //desarrollamos un for, para que realice un ciclo contado un inicio que sera de i valor 1, y finalizara con n, que es el número ingresado por el usuario, cada conda ira sumando a i un valor, para que llege a las condiciones para que este ciclo funcione
```

```
    for (int i = 1; i <= n; ++i) {
```

```
        //utilizamos && para comparar compara las respuestas de pares que es verdadero, con i, que se cumplira si es par, o impar, que el "o" es "||", pero este deberá dividirse y si el residuo es igual a 0 o a 1, seguira con el procedimiento,
```

```
        if ((pares && i % 2 == 0) || (!pares && i % 2 == 1)) {
```

```
            // en cada vuelta nosotros sumamos a suma i, que tendra un valor diferente según la vuelta o ciclo en el que se encuentre
```

```
            suma += i;
```

```
        }
```

```
    }
```

```
    return suma;
```

```
}
```

```
// Funcion para factorial, no pedimos ningun numero de ingreso, más que solo los que el usuario ya ha ingresado
```

```
int factorial(int numero) {
```

```
    //establecemos que la variable factorialN es igual a 1, lo que nos ayudara a no empezar desde 0, sino desde una base 1.
```

```

int factorialN = 1;

while (numero < 0) {

    //no se aceptara números negativos

    cout << "El numero debe ser positivo. Introduzca un numero: "<< endl;

    cin >> numero;

}

while (numero > 1) {

    //estamos diciendo que factorialN se multiplicara con numero, que es la variable
    que el usuario ingreso, y por cada vuelta, se plantera que numero sea mayor a 1, y como podemos
    ver, cada vuelta del ciclo esta restando un numero a el valor de numero, hasta que ya no sea
    mayor que 1, y while, no siga trabajando.

    factorialN *= numero;

    numero--;

}

return factorialN;

}

```

// Los digitos de un número, ya pueden ser solo uno, o varios, depende de la cantidad exacta de numeros que se ingresen

```

int CDigitos(int digito) {

    int contador = 0;

    //establecemos que contador sea 0, y que nuestro do while, tenga como condición que
    funcionara solamente si digito no es igual a 0, si es igual a 0 este contador seguira con el ciclo

    do {

        //dividimos el valor que el usuario ingredo, que en este caso es la variable digito, y que
        cada vuelta se sume un numero a contador, lo que nos dira que si un numero es unitario, solo
        tendra un ciclo, pero si es más grande, ya decimal, o más grande, este seguira dando ciclos, y
        sumando uno, a contador,

        digito /= 10;

        contador++;

    } while (digito != 0);

    return contador;
}

```

```
}
```

// En este caso no devolvemos un resultado, sino establecemos una funcion sin retorno, sino, que entregue un procedimiento, en el cual evalua los digitos ingresados, y según estos sean mayores o menores, este hará el trabajo de evaluar, y así se podrá ordenar los numeros según su evaluación.

```
void NumerosOrdenados(int &a, int &b, int &c) {
```

//establemos tres if, los cuales nos ayudaran a dar una respuesta segun los valores que el usuario ingreso, donde utilizamos swap, para el intercambio de valores, ya sea a con b, o b con c.

```
if (a > b) {
```

```
    swap(a, b);
```

```
}
```

```
if (b > c) {
```

```
    swap(b, c);
```

```
}
```

```
if (a > b) {
```

```
    swap(a, b);
```

```
}
```

```
}
```

// realizamos una funcion para sumar los digitos de una variable, segun sus difitos, estos se sumaran, como si tiene 4 digitos, con un valor independiente a cada uno, o el mismo para todos, estos se sumaran, si es 0 no tiene nungun valor, pero si es un numero x, todos sus digitos serán sumados.

```
int masDigitos(int numero) {
```

```
    int suma = 0;
```

//while tiene como condición que será realizara el ciclo si numero no es igual a 0.

```
    while (numero != 0) {
```

//establecemos la suma como, el valor de numero será dividido por 10, como también se establecera que su valor para el siguiente ciclo será de la división con 10, hasta llegar a 0, y el ciclo while, ya no siga.

```
        suma += numero % 10;
```

```
        numero /= 10;
```

```

    }

    return suma;
}

//establecemos un menú independiente, para que se vea más ordenado, y no saturar con tanto
contenido a el int central.

int guia() {
    int eleccion;

    cout << "\n\n\t" << "//Hoja de trabajo#5\\" << endl;

    cout << "\n\t--- Elige el ejercicio ---" << endl;

    cout << "1. Suma de numeros pares/impares" << endl;

    cout << "2. Factorial de un numero" << endl;

    cout << "3. Contador de digitos" << endl;

    cout << "4. Ordenamiento de numeros" << endl;

    cout << "5. Suma de digitos de un numero" << endl;

    cout << "0. Salir" << endl;

    cin >> eleccion;

    return eleccion;
}

int main() {
    int opcion;

    bool salir = false;

    //nuestro do-while nos dice que funcionara solo si while no es igual a 6, sino imprimira en pantalla
    el ultimo mensaje, que dira, salir del programa.

    do{

        system("cls");

        //establecemos que opcion sea igual a la respuesta que nos dará guia, que es nuestro
        menú en otra funcion con retorno.

        opcion = guia();

        switch (opcion) {

```

```

case 1: {
    //establecemos los valores necesarios para que funcione nuestro proceso, con valores
    enteros, y booleanos, para que sea envaluado en el procedimiento.

    system("cls");

    cout << "\n\t#Suma de numeros pares/impares" << endl;

    int suma, n;

    bool pares;

    cout << "Introduzca un numero: ";

    cin >> n;

    cout << "¿Sumar numeros pares (1) o impares (0)? ";

    cin >> pares;

    //suma será lo mismo que la funcion llamada "sumaPI", que dignifica sumaParealmpares.

    suma = sumaPI(n, pares);

    //lo que significa que "?", es decir que la primera opción será tomada como verdadera, y la
    respuesta será "pares", y si la respuesta es falsa, imprimira la respuesta "impares".

    cout << "La suma de los numeros " << (pares ? "pares" : "impares") << " hasta " << n << " es: "
    << suma << endl;

    system("pause");

    break;
}

case 2: {
    system("cls");

    cout << "\n\t#Factorial de un numero" << endl;

    int n;

    cout << "Introduzca un numero: ";

    cin >> n;

    //establecemos que nuestra variable fact, sera igual a nuestra funcion "factorial".

    int fact = factorial(n);

    cout << "El factorial de " << n << " es: " << fact << endl;

    system("pause");

```

```

        break;
    }
    case 3: {
        system("cls");
        cout << "\n\t#Contador de digitos" << endl;
        int num;
        cout << "Introduzca un numero: ";
        cin >> num;

        //establecemos que nuestra variable Digitos, sera igual a nuestra funcion "CDigitos", que
        //significac Contador de digitos.
        int Digitos = CDigitos(num);
        cout << "El numero " << num << " tiene " << Digitos << " digitos" << endl;
        system("pause");
        break;
    }
    case 4: {
        system("cls");
        cout << "\n\t#Ordenamiento de numeros" << endl;

        //como son tres valores los que necesitamos, establecemos 3 variables para el programa,
        //que cada una sea de un valor entero.
        int a, b, c;
        cout << "Introduzca el primer numeros: ";
        cin >> a;
        cout << "Introduzca el segundo numeros: ";
        cin >> b;
        cout << "Introduzca el tercer numeros: ";
        cin >> c;

        //establecemos la respuesta como el usuario lleno cada variable.
        cout << "Numeros antes de ordenar: " << a << ", " << b << ", " << c << endl;
        NumerosOrdenados(a, b, c);
    }
}

```


//al momento que regreso nuestro función sin retorno, y realizo todo su procedimiento, imprimimos una nueva respuesta, que será ordenada nuevamente.

```
    cout << "Numeros después de ordenar: " << a << ", " << b << ", " << c << endl;

    system("pause");

    break;

}

case 5: {

    system("cls");

    cout << "\n\t#Suma de digitos de un numero" << endl;

    int n;

    cout << "Introduzca un numero: ";

    cin >> n;

    //suma será igual a la función "masDigitos", en donde se realiza todo el procedimiento de la
    suma entre digitos de un numero.

    int suma = masDigitos(n);

    cout << "La suma de los digitos de " << n << " es: " << suma << endl;

    system("pause");

    break;

}

case 6: {

    system("cls");

    int n;

    cout << "Saliendo del programa..." << endl;

    break;}

    default:

    cout << "Opcion no encontrada. Intenta de nuevo." << endl;

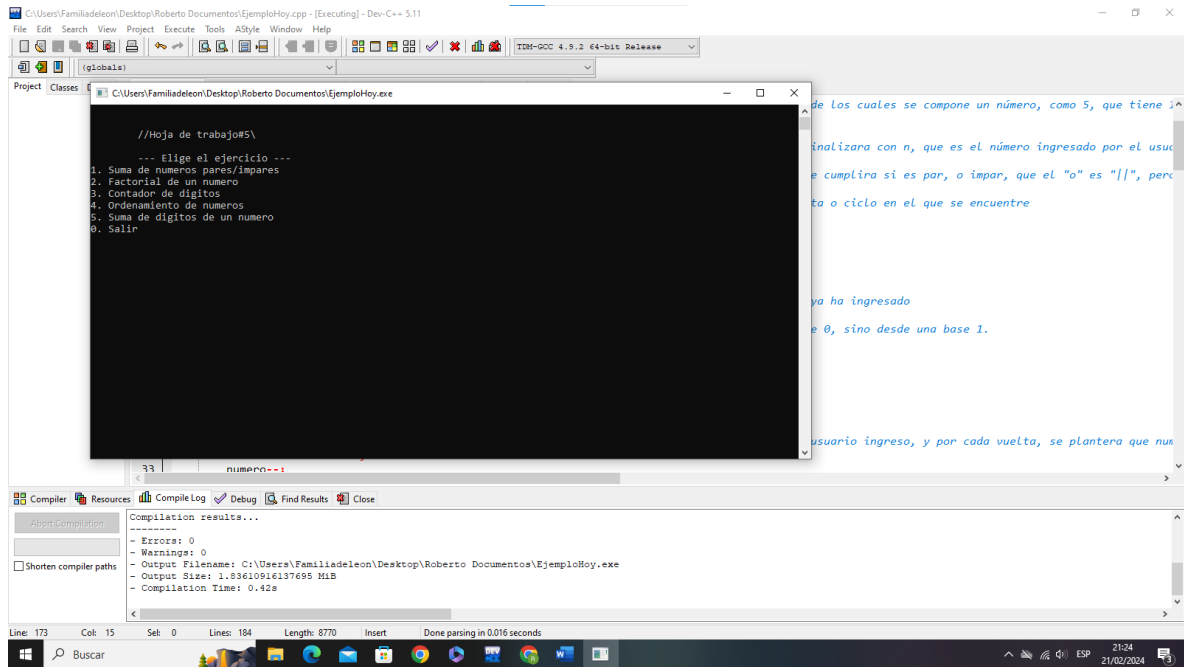
}

}while (opcion !=6);

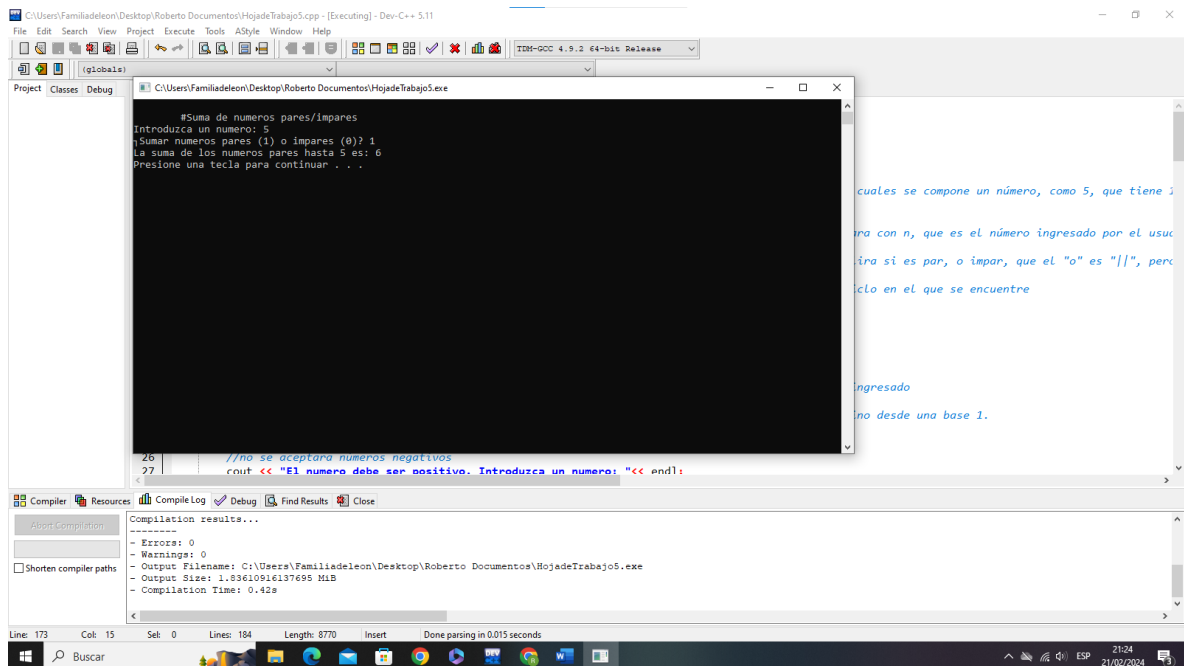
system("pause");

return 0;}
```

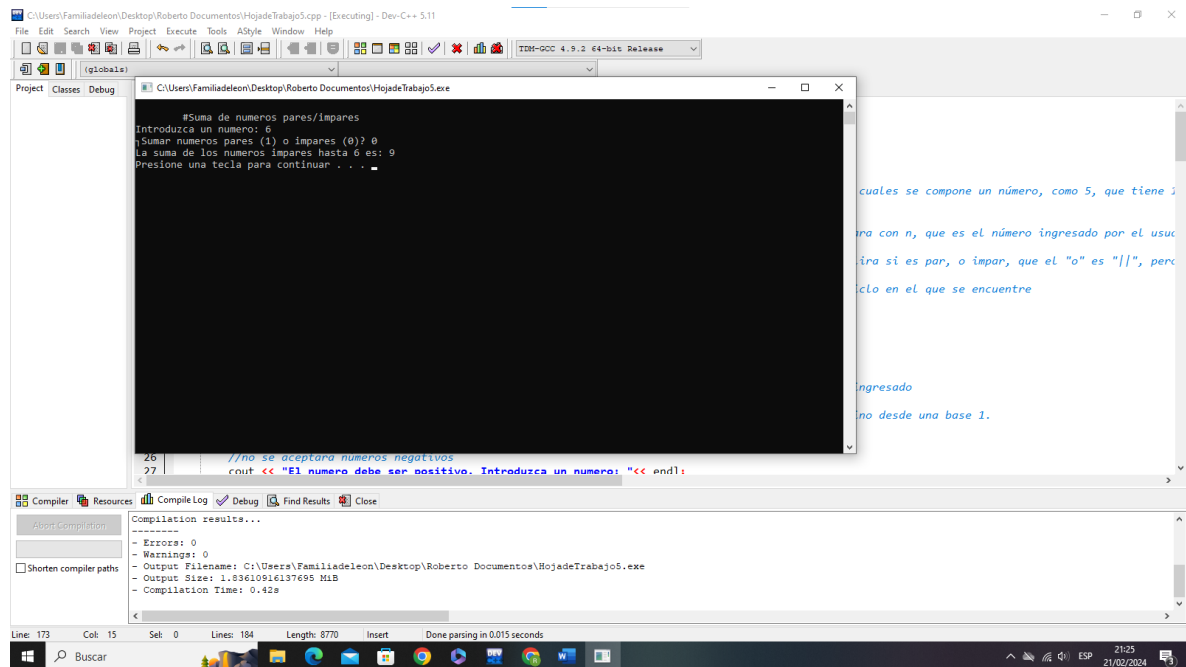
Imágenes de algunos procedimientos:



Realizamos la suma de los números pares de 5



Realizamos la suma de los números impares de 6



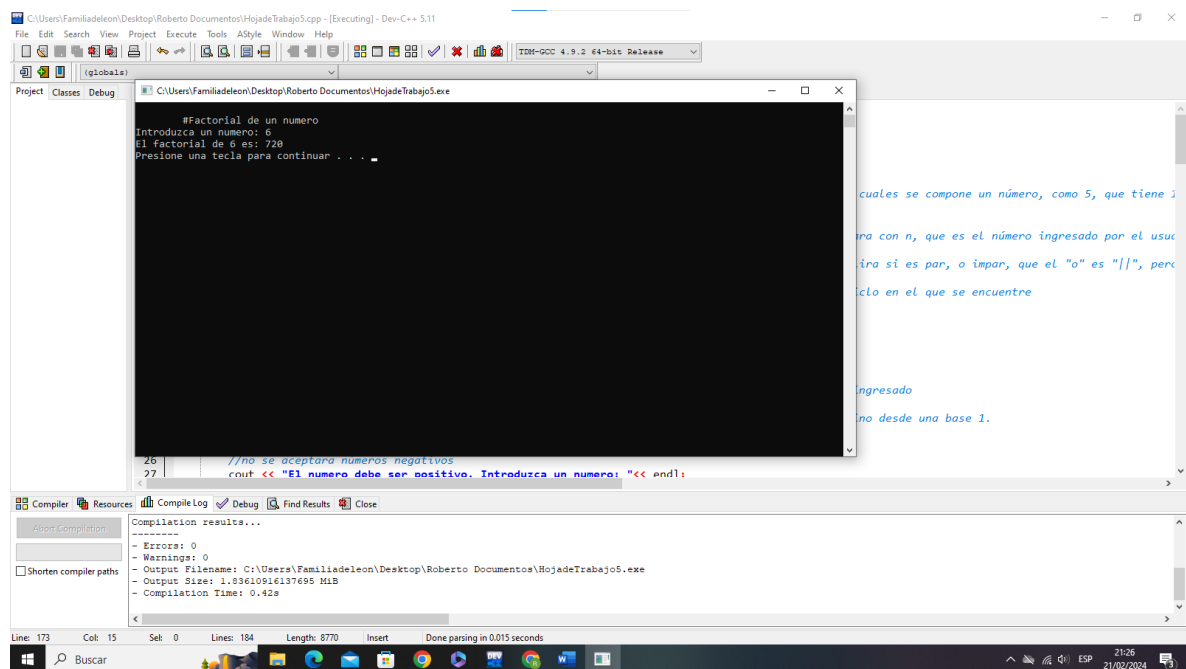
```
#Suma de numeros pares/impares
Introduzca un numero: 6
Sumar numeros pares (1) o impares (0)? 0
La suma de los numeros impares hasta 6 es: 9
Presione una tecla para continuar . . .
```

```
//no se aceptara numeros negativos
cout << "El numero debe ser positivo. Introduzca un numero: "<< endl;
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Familiadeleon\Desktop\Roberto Documentos\HojadeTrabajo5.exe
- Output Size: 1.83610916137695 MIB
- Compilation Time: 0.42s

Queremos saber cuál es la factorial de 6



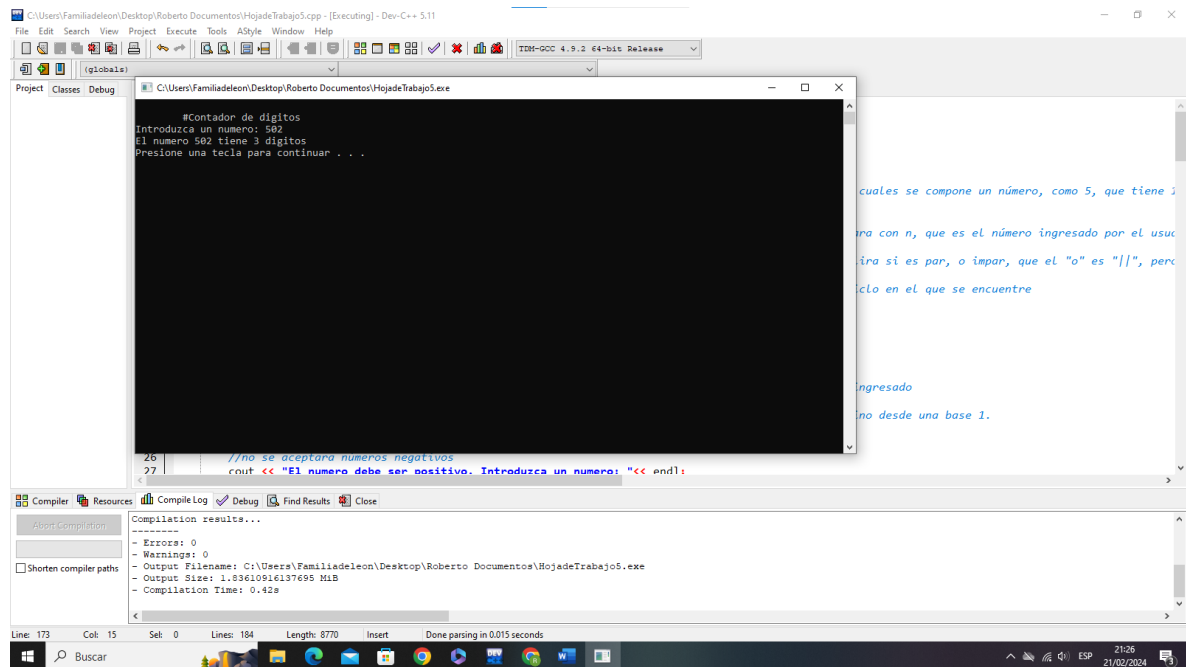
```
#Factorial de un numero
Introduzca un numero: 6
El factorial de 6 es: 720
Presione una tecla para continuar . . .
```

```
//no se aceptara numeros negativos
cout << "El numero debe ser positivo. Introduzca un numero: "<< endl;
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Familiadeleon\Desktop\Roberto Documentos\HojadeTrabajo5.exe
- Output Size: 1.83610916137695 MIB
- Compilation Time: 0.42s

Los dígitos de 502 son:



```
#Contador de digitos
Introduzca un numero: 502
El numero 502 tiene 3 digitos
Presione una tecla para continuar . . .

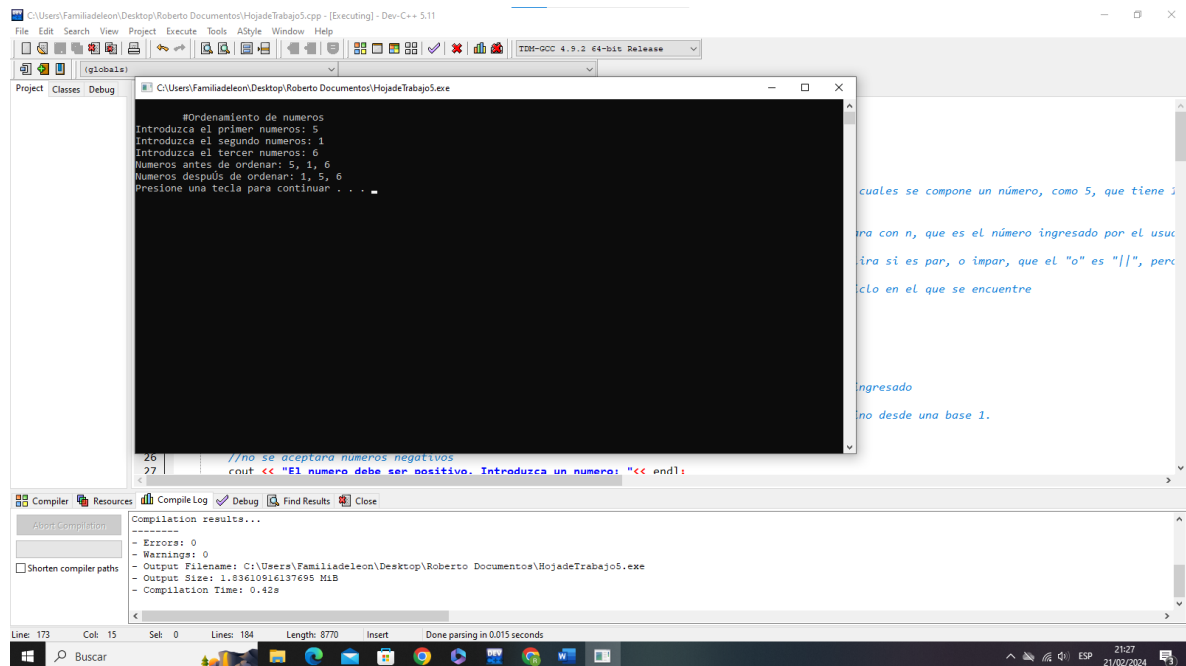
//no se aceptara numeros negativos
cout << "El numero debe ser positivo. Introduzca un numero: "<< endl;
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Familiadeleon\Desktop\Roberto Documentos\HojadeTrabajo5.exe
- Output Size: 1.83610916137695 MIB
- Compilation Time: 0.42s

Line: 173 Col: 15 Set: 0 Lines: 184 Length: 8770 Insert Done parsing in 0.015 seconds

Vamos a ordenar los números 5, 6, y 1:



```
#Ordenamiento de numeros
Introduzca el primer numero: 5
Introduzca el segundo numero: 1
Introduzca el tercer numero: 6
Numeros antes de ordenar: 5, 1, 6
Numeros despues de ordenar: 1, 5, 6
Presione una tecla para continuar . . .

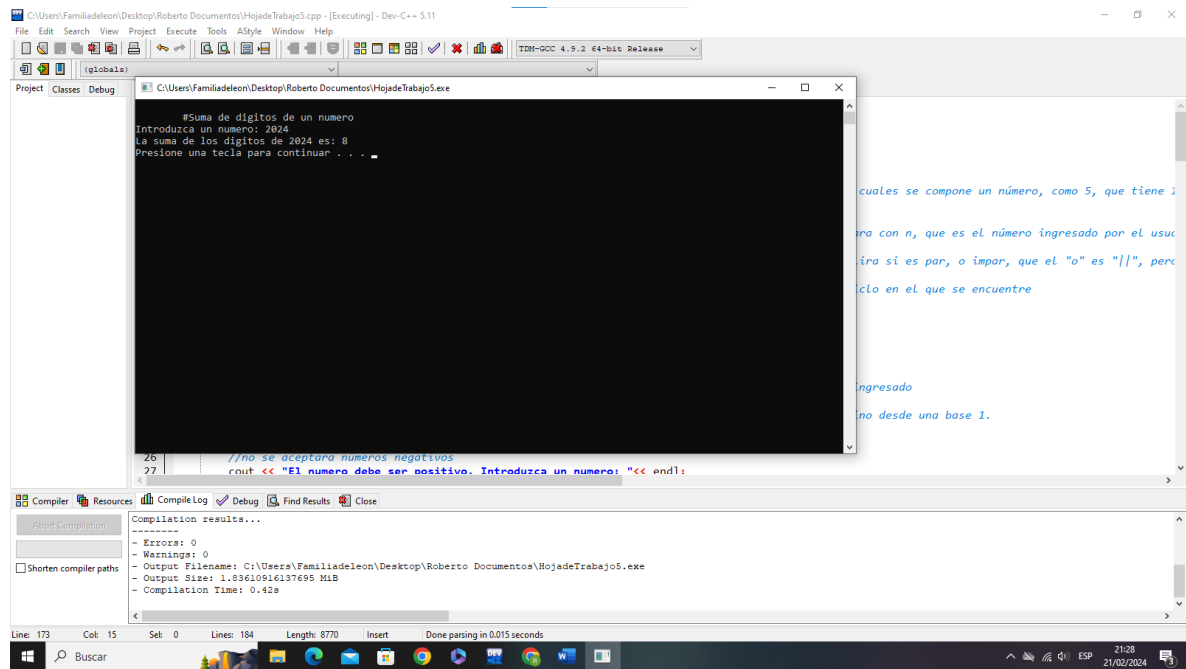
//no se aceptara numeros negativos
cout << "El numero debe ser positivo. Introduzca un numero: "<< endl;
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Familiadeleon\Desktop\Roberto Documentos\HojadeTrabajo5.exe
- Output Size: 1.83610916137695 MIB
- Compilation Time: 0.42s

Line: 173 Col: 15 Set: 0 Lines: 184 Length: 8770 Insert Done parsing in 0.015 seconds

La suma de dígitos del número 2024:



The screenshot shows a C++ IDE with a project named "HojadeTrabajo5.exe". The code in the editor is as follows:

```
#Suma de digitos de un numero
Introduzca un numero: 2024
La suma de los digitos de 2024 es: 8
Presione una tecla para continuar . . .
```

The output window shows the same text, indicating the program has executed successfully. The compiler window shows the following compilation results:

```
Compilation results...
-----
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Familiadeleon\Desktop\Roberto Documentos\HojadeTrabajo5.exe
- Output Size: 1.83610916137695 MIB
- Compilation Time: 0.42s
```

The status bar at the bottom indicates the current line is 173, column 15, and the file is "HojadeTrabajo5.cpp".

GitHub: [RobertodeLeonUmg/HojadeTrabajo-5: ciclos y funciones \(github.com\)](https://github.com/RobertodeLeonUmg/HojadeTrabajo-5: ciclos y funciones).

Conclusiones

Realizar un trabajo con ciclos, y funciones es muy bonito cuando te sale, pero un pequeño error, o que no se haya visto el código correcto, no hará que avances, y esto fue lo que me paso, hasta que lo logre solucionar, al analizar línea por línea de todo mi código.