|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE DE LA PRÁCTICA** | **Practicas del Lenguaje en C** | | | **No.** | **1** |
| **ASIGNATURA:** | **Métodos Numéricos** | **CARRERA:** | **INGENIERIA EN SISTEMAS COMPUTACIONALES** | **PLAN:** | **ISIC-2010-224** |

**I. COMPETENCIA(S) ESPECÍFICA(S):**

**II. MATERIAL EMPLEADO:**

* Word
* Dev-Cpp 5.11 TDM-GCC 4.9.2 Setup

**III. DESARROLLO DE LA PRÁCTICA:**

En este reporte se va a mostrar cómo es que se programa en Lenguaje C desde cero esto me ayudo a poder entender cómo es que este lenguaje nos permite realizar varias acciones de manera que pueda entender su comportamiento o bien su manejo.

Lo primero que debo de preguntarme es:

¿Qué es el lenguaje C?

C es un lenguaje de programación estructurada desarrollado a principios de los años 70. Es un lenguaje de nivel medio, pero con muchas características de bajo nivel.

Otra cosa que debemos de tener en cuenta son sus principales características las cuales son:

* Lenguaje de programación de propósito general, muy adecuado para la programación de sistemas (UNIX ) fue escrito en C.
* Lenguaje relativamente pequeño: solo ofrece sentencias de control sencillas y funciones.
* \*La E/S no forma parte del lenguaje, sino que se proporciona a través de una biblioteca de funciones.

Ya una vez entendido eso empezamos a programar en c lo cual utilizamos el programa Dev-C++ para poder programar en lenguaje C comenzamos.

**2.Lenguaje C**

Los principales componentes para poder programar en C son las importar de las librerías que son;

#include<stdio.h>

#include<stdlib.h>

Que nos ayudaran a poder agregar componentes de estas. También utilizaremos el método principal que es

Int main(){

sintaxis

}

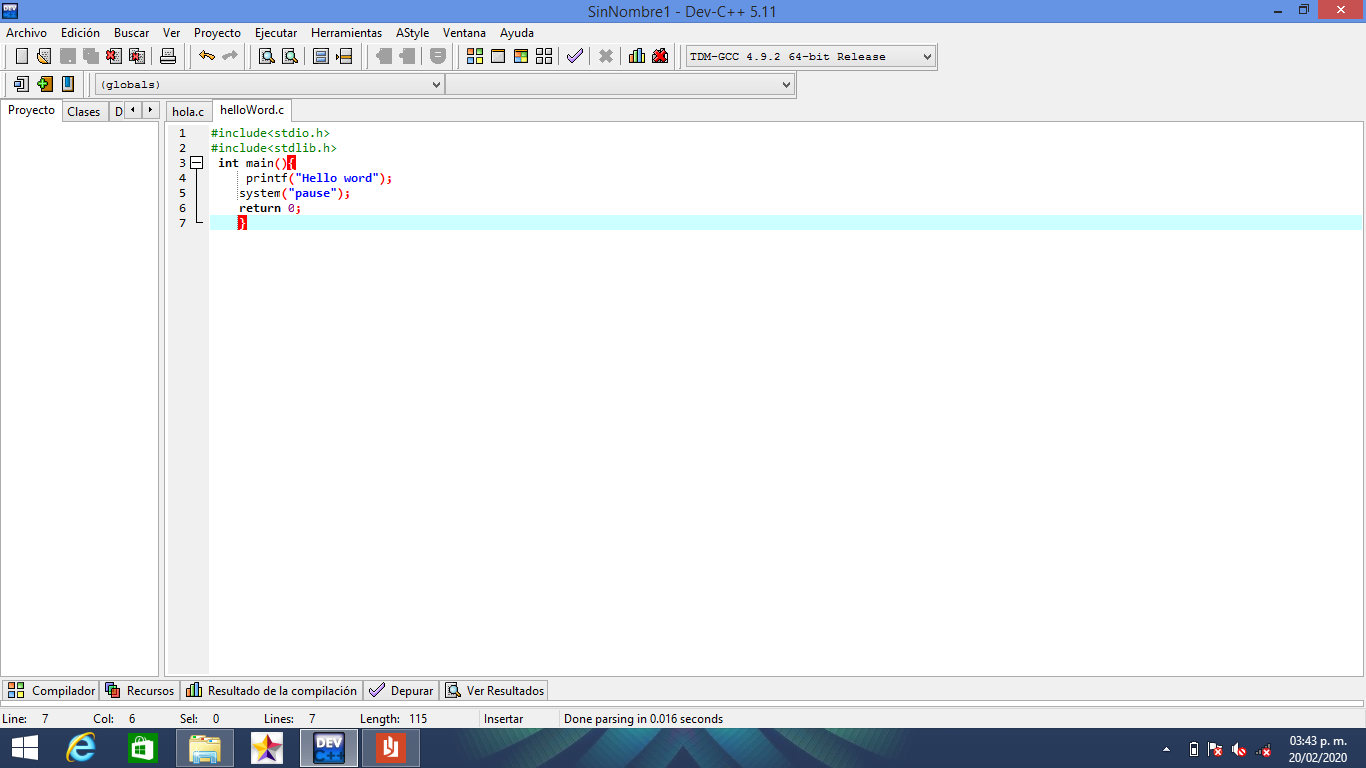
Dentro de las llaves va a llevar las líneas que queremos que se ejecuten asi como es mostrar un texto, realizar una operación o bien realizar una condición.

Lenguaje C

Ejercicio 1

Vamos a programar un ejercicio muy básico que se dice que un programador lo primero que programa es su Hola Mundo y no vamos a perder la costumbre. Para esto en la pantalla de abajo se muestra un ejercicio y se muestra los datos que lo componen.

Importamos las librerías

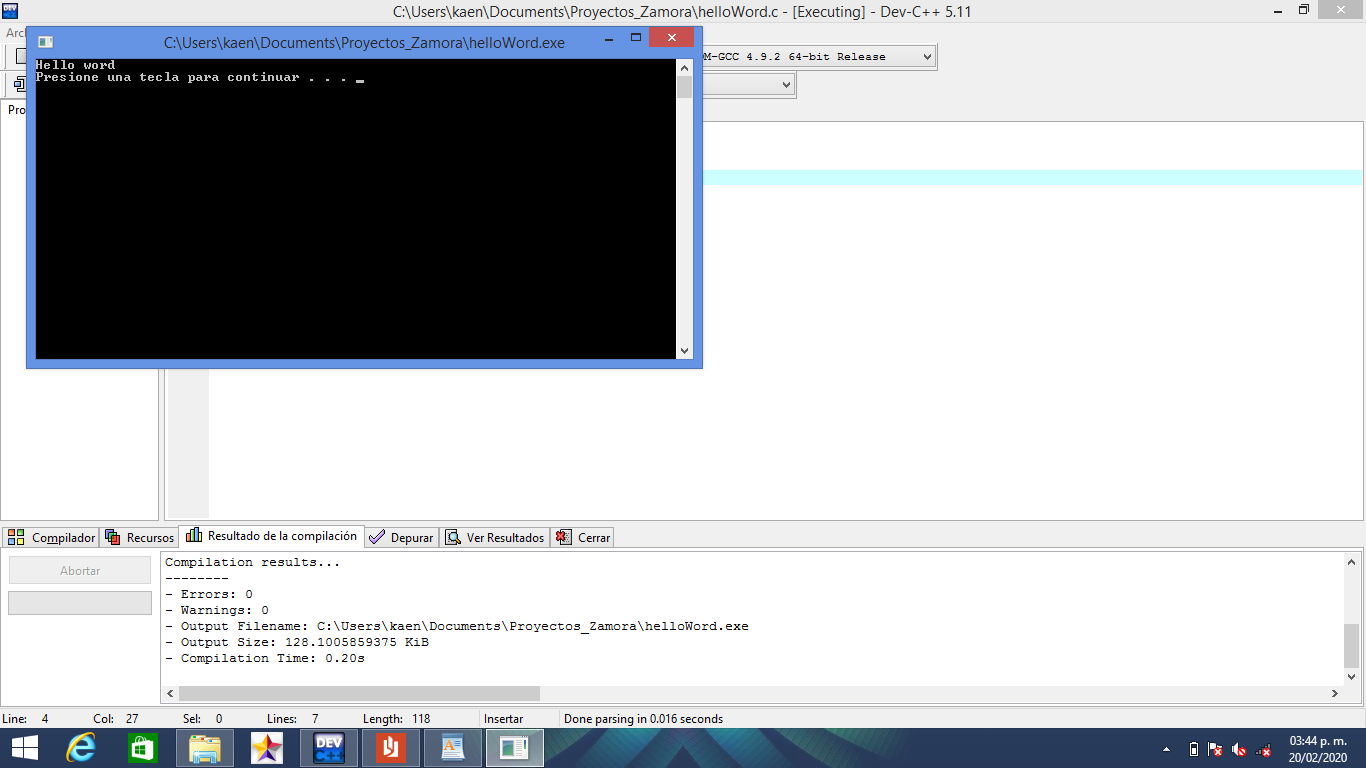


El system(“pause ”) nos sirve para que cuando ejecutemos nuestro programa la consola se quede abierta y no se cierre.

Con ayuda del printf vamos a mandar un mensaje que contendrá el Hola Mundo

Hacemos uso del método principal

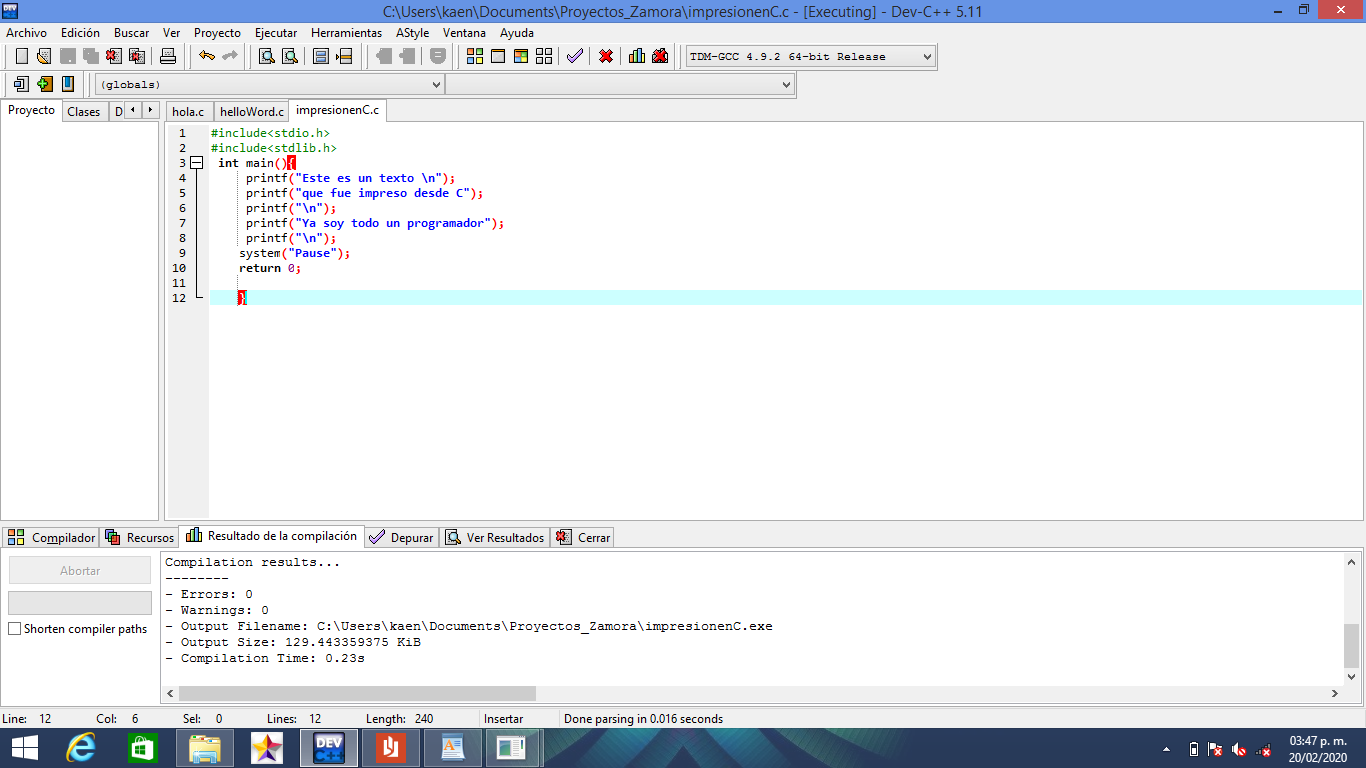
Y al final ejecutamos nuestra aplicación y notamos que se muestra un mensaje



Ejercicio 2

En este ejercicio se muestra el uso también de printf que nos ayuda como se mencionó anteriormente a mostrar mensajes para imprimirlos en consola, después mas adelante se mencionara que más se puede hacer con esta función.

El ejemplo que está en la pantalla solo se muestran mensajes que se van a mostrar.

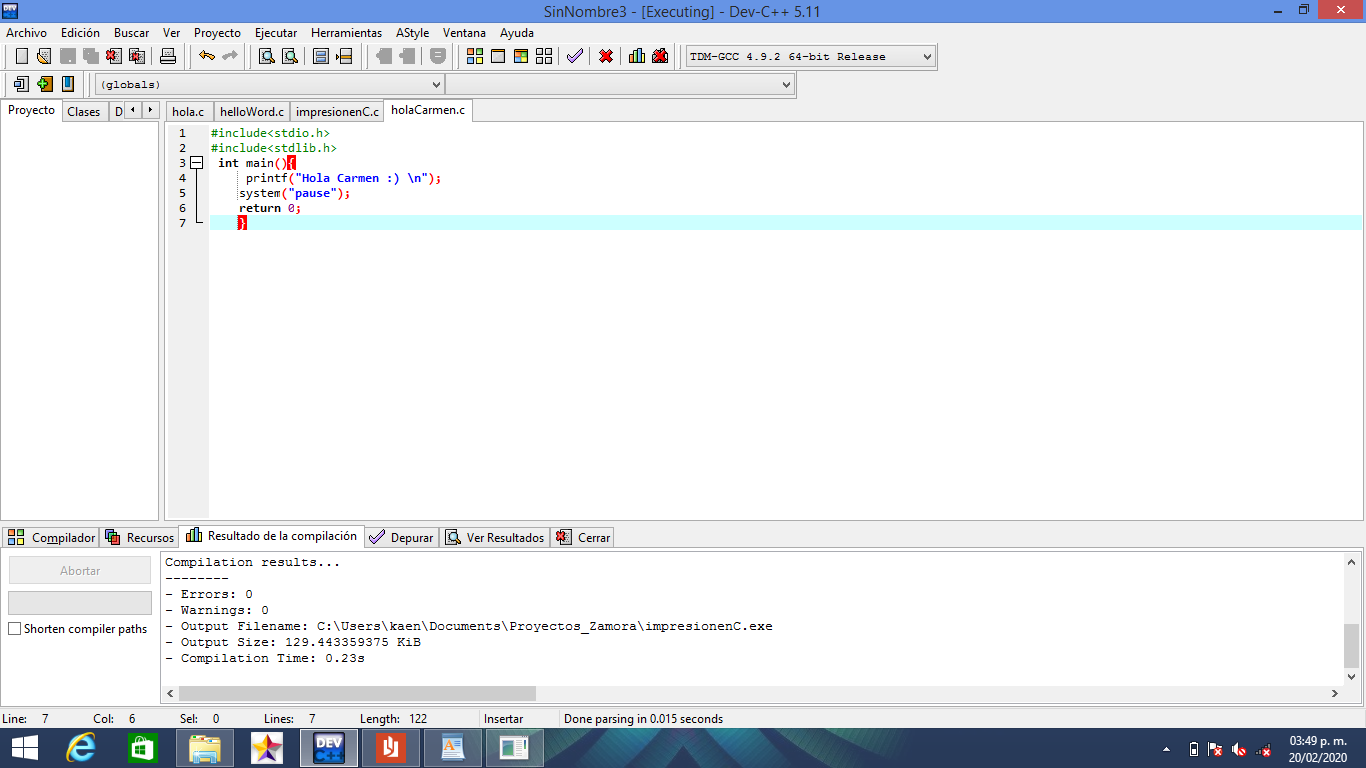


En la pantalla de abajo se muestra los mensajes que necesitábamos que se mostraran la \n nos ayuda a poder hacer saltos de linea

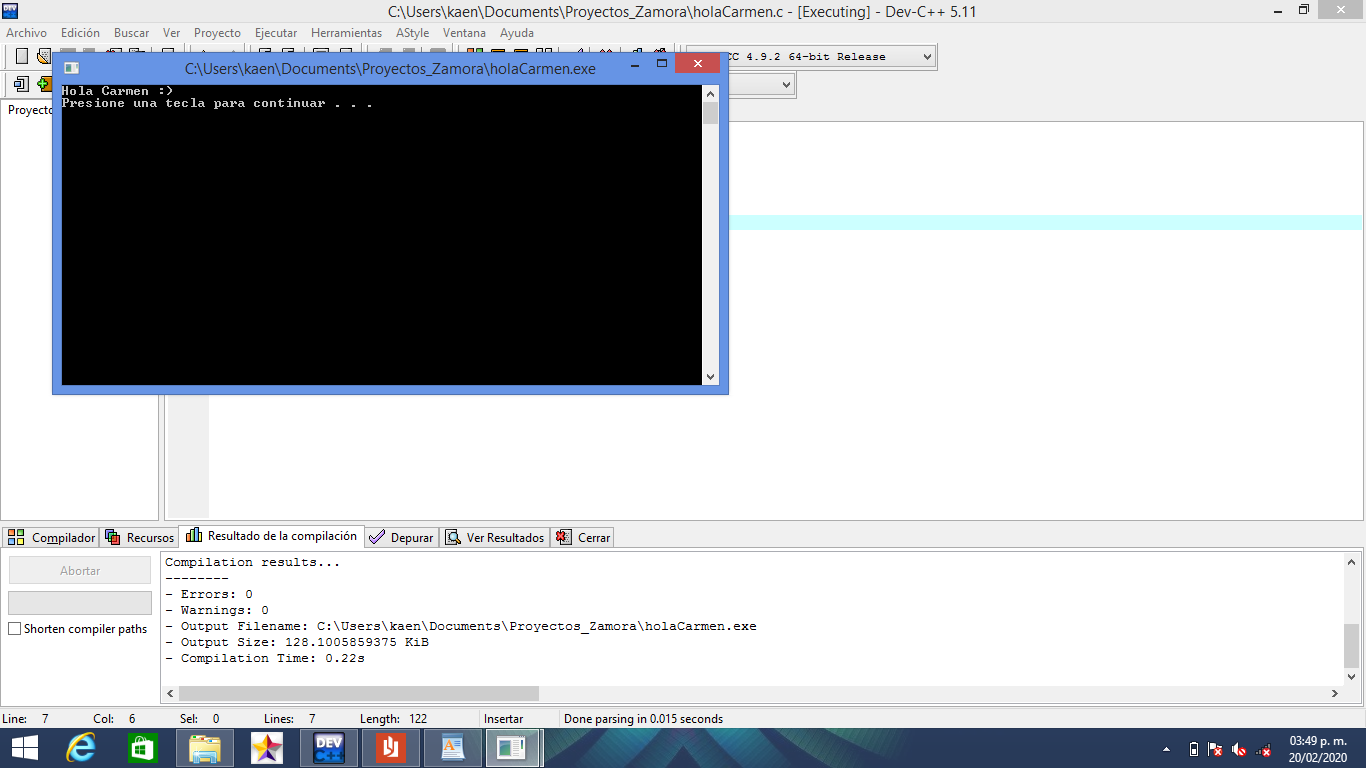


Ejercicio 3

En el siguiente ejercicio solo era mostrar un solo mensaje que digiera hola y mi nombre y eso fue lo que realice

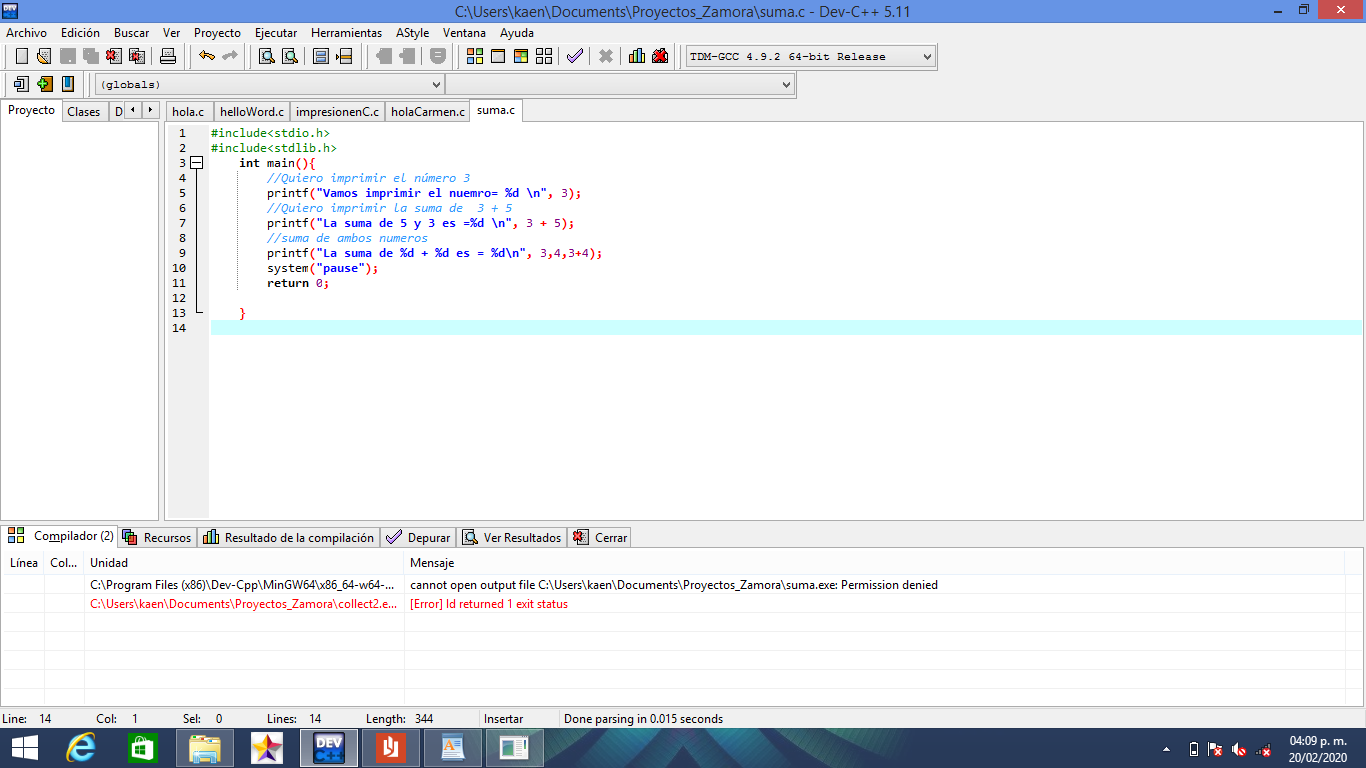


Cuando se ejecuta se muestra de la siguiente manera y eso quiere decir que estuvo muy bien programado lo que deseábamos hacer.



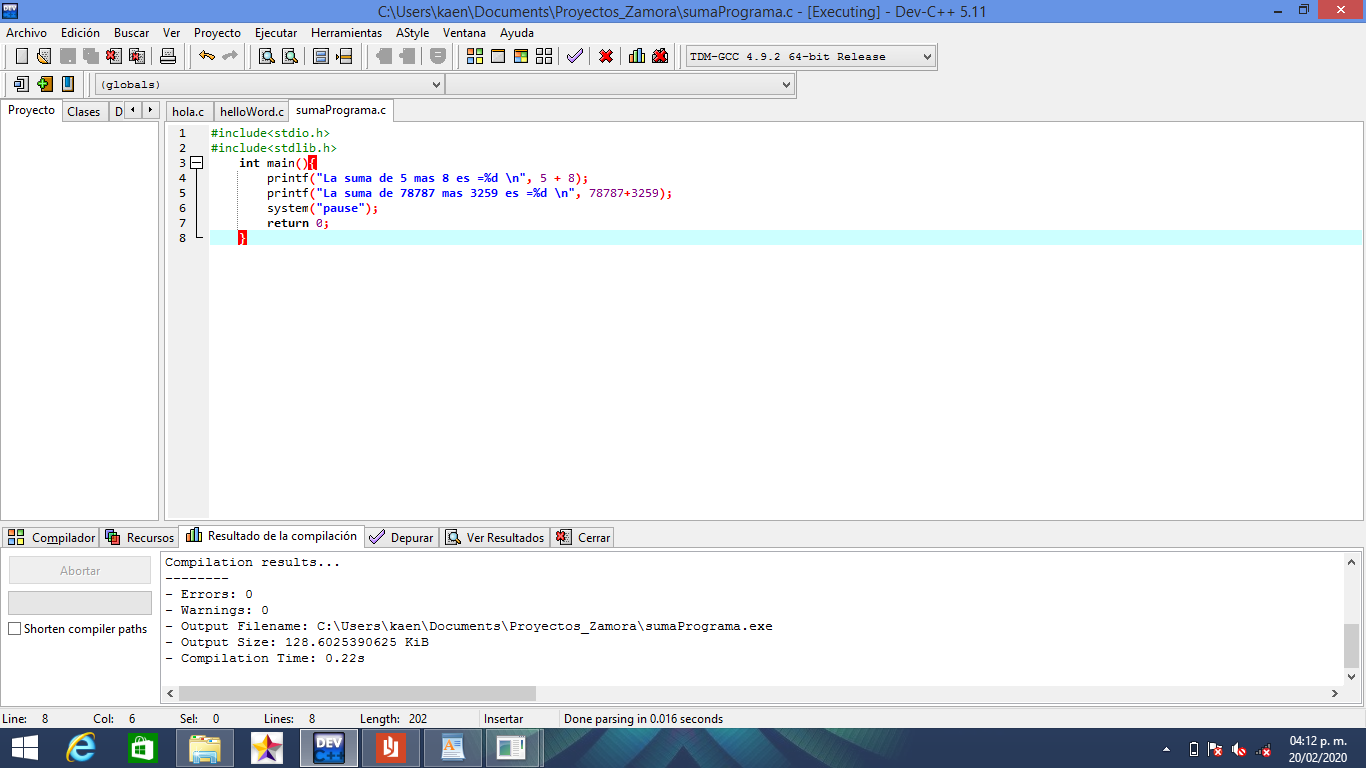
Ejercicio 4

En este ejercicio teníamos que hacer una suma de dos números, así como también mostrar un mensaje, la función printf nos ayuda a poder imprimir un mensaje, así como también a poder imprimir un valor hay. En el ejemplo se muestra el mensaje y está un 3 pero antes tenemos este símbolo %d que nos sirve para indicarle al programa que hay se está esperando un valor que es 3 en este caso.

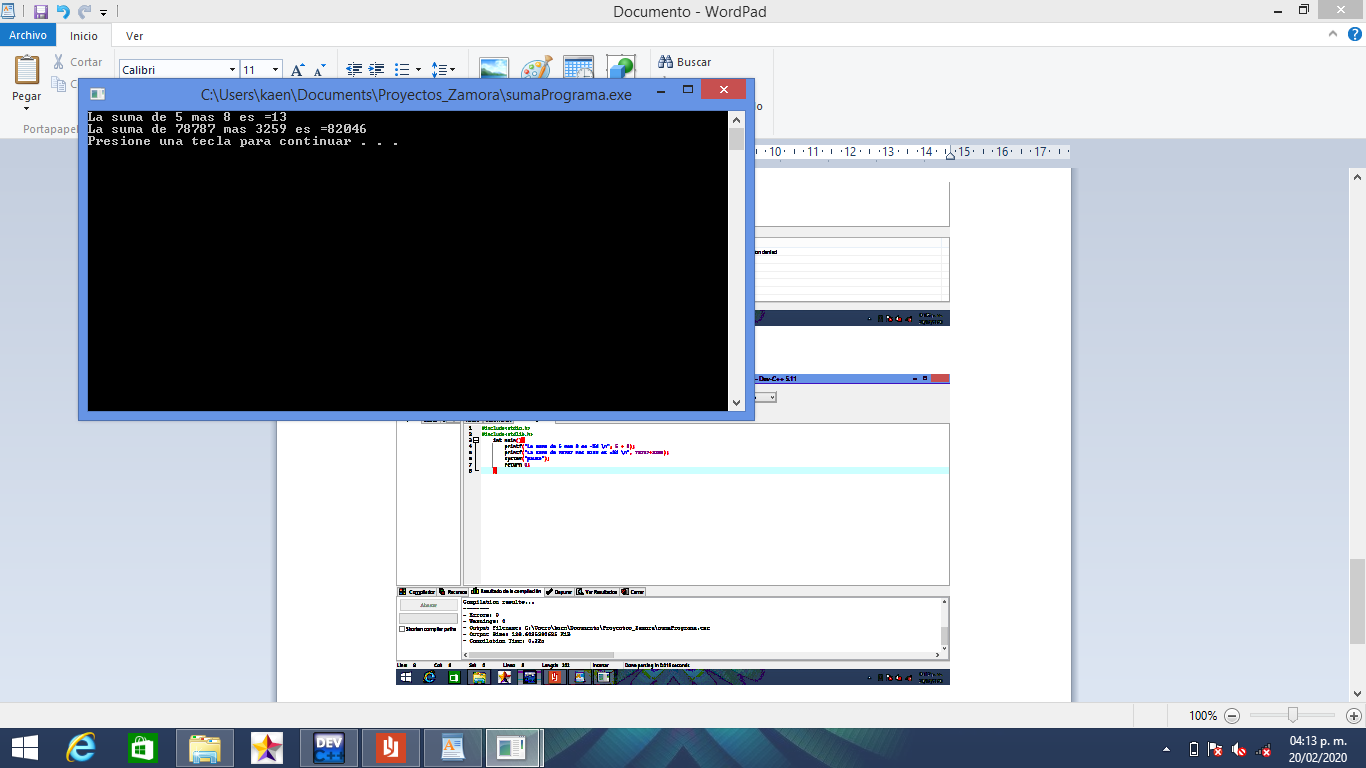


Ejercicio 5

Es este otro ejercicio era realizar dos sumas como se muestra en la pantalla está el mensaje y el símbolo %d que se refiere que se está esperando un valor y es el resultado de 5 más 8 el símbolo de la \n nos indica que va hacer un salto de línea.



En la pantalla de abajo se muestra la suma de las dos cantidades.



**3.Variable\_C**

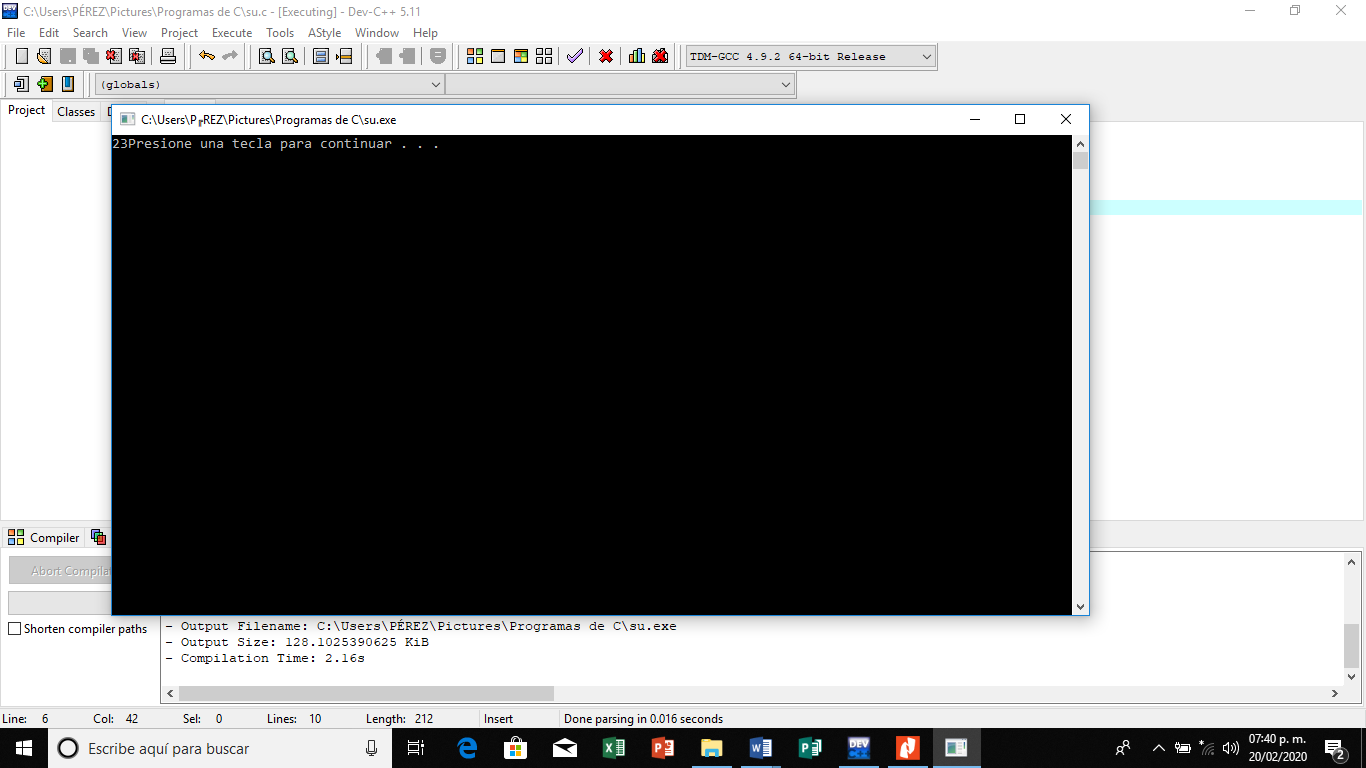
El lenguaje C es fuertemente tipado, lo que significa que todas las variables deben ser declaradas antes de utilizarse.

Ejercicio 1

En el ejercicio se muestra cómo es que declaramos 2 variable y a estas les asignamos un valor ya dado que se guardara en memoria que es 8 y 15 veremos el comportamiento de estas ya que cuando se ejecuta el programa lo único que hace es sumarlas y mostrar el resultado de estas.

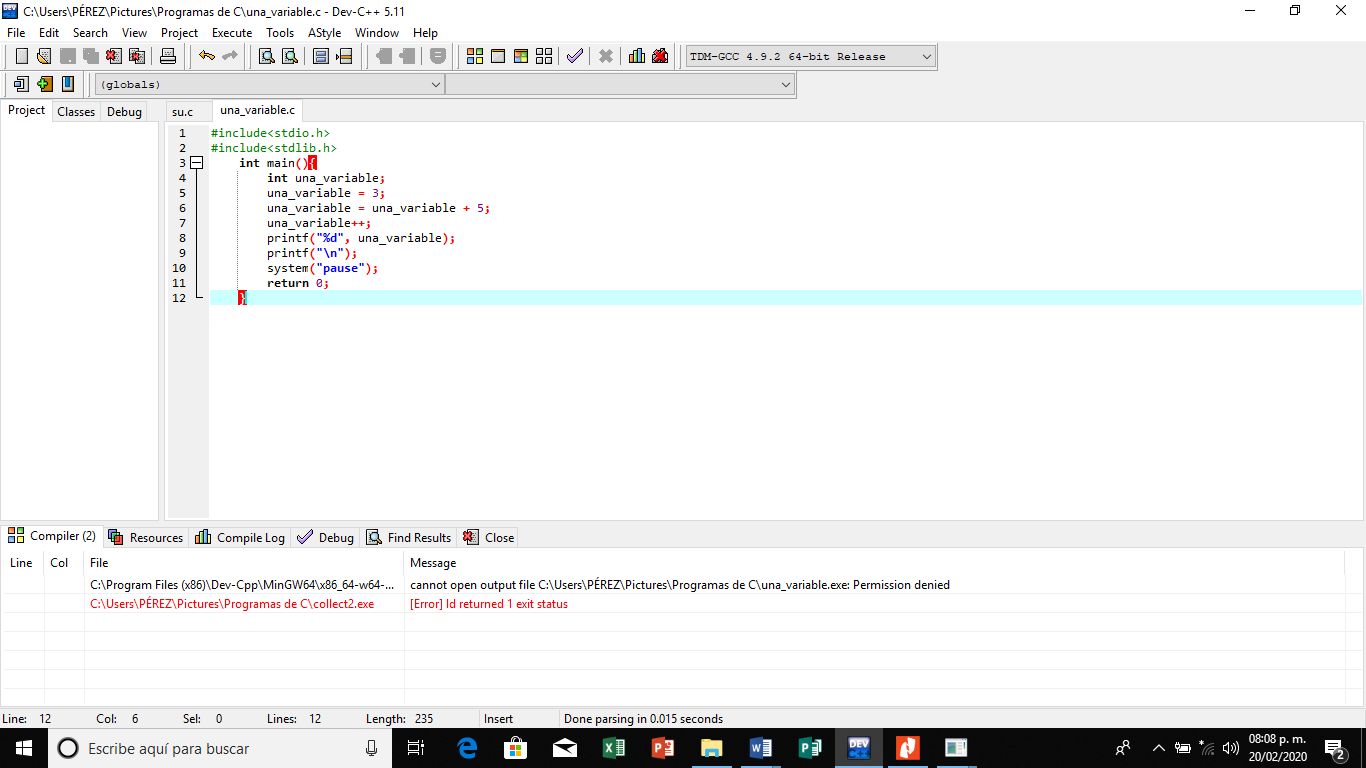


En la pantalla de abajo se muestra el resultado de la suma de estos valores que 5 + 8 es igual a 23 y efectivamente ese valor nos los da.

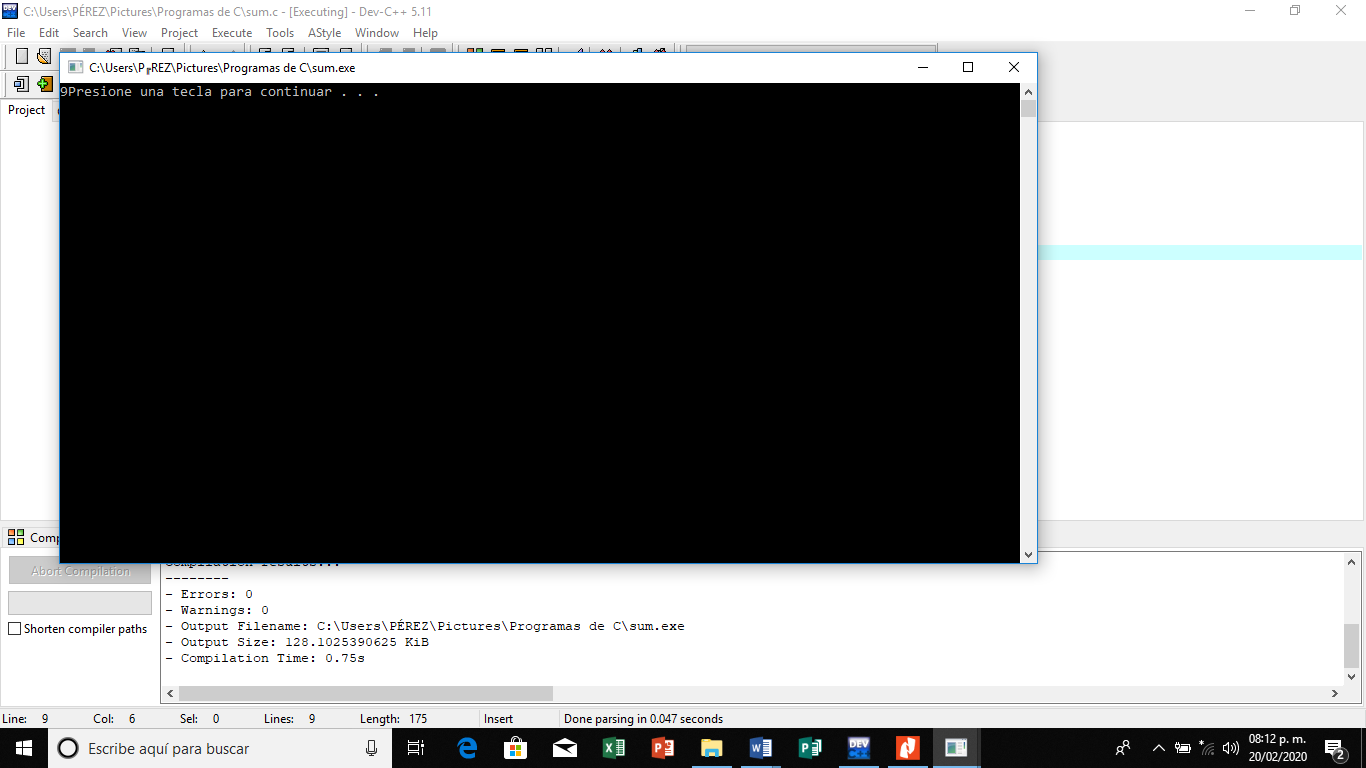


Ejercicio 2

El ejercicio consistía en declarar una variable de tipo entero llamada una\_variable después a esa variable asignarle un valor que en este caso es 3 posteriormente en esa variable vamos a guardar el valor de esa misma más 5 por ultimo esta se incrementa y finalmente imprimimos.

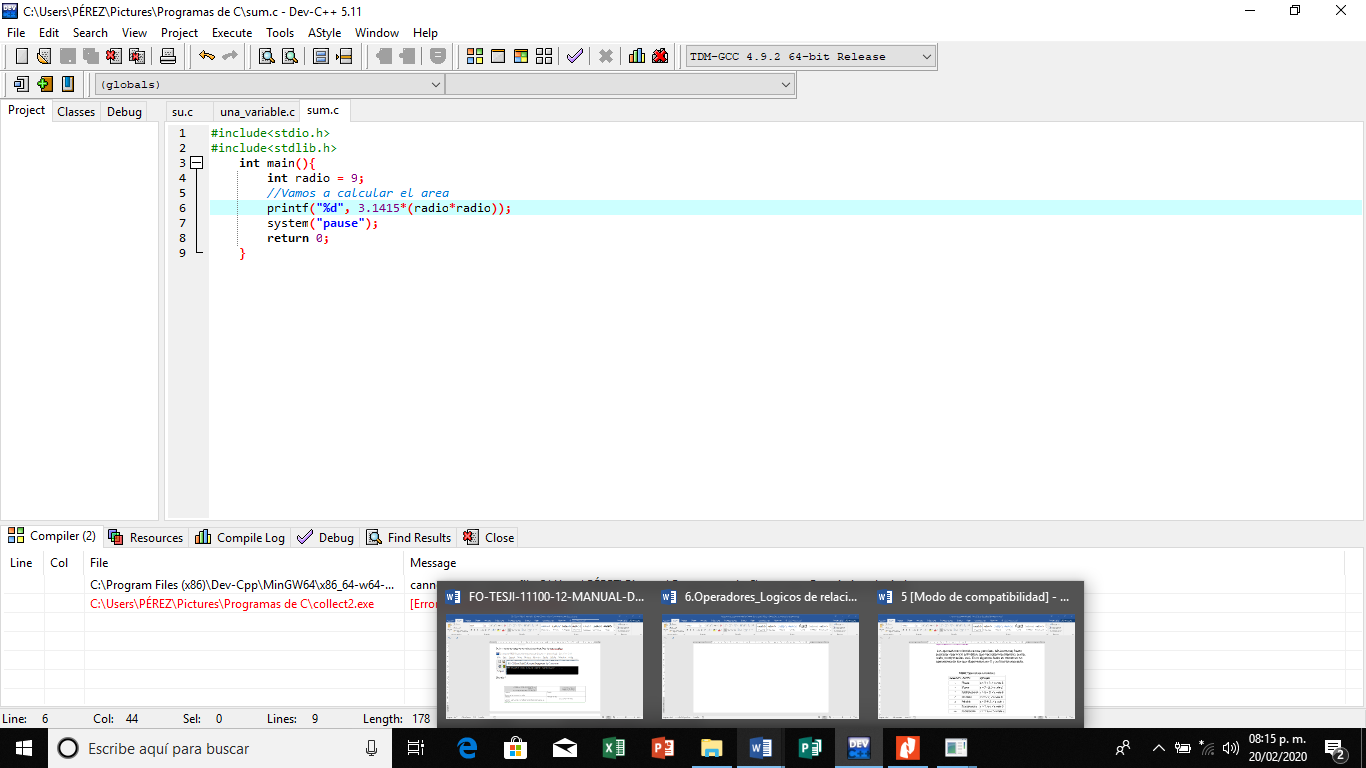


En la ventana de abajo se muestra el resultado final de una\_variable.

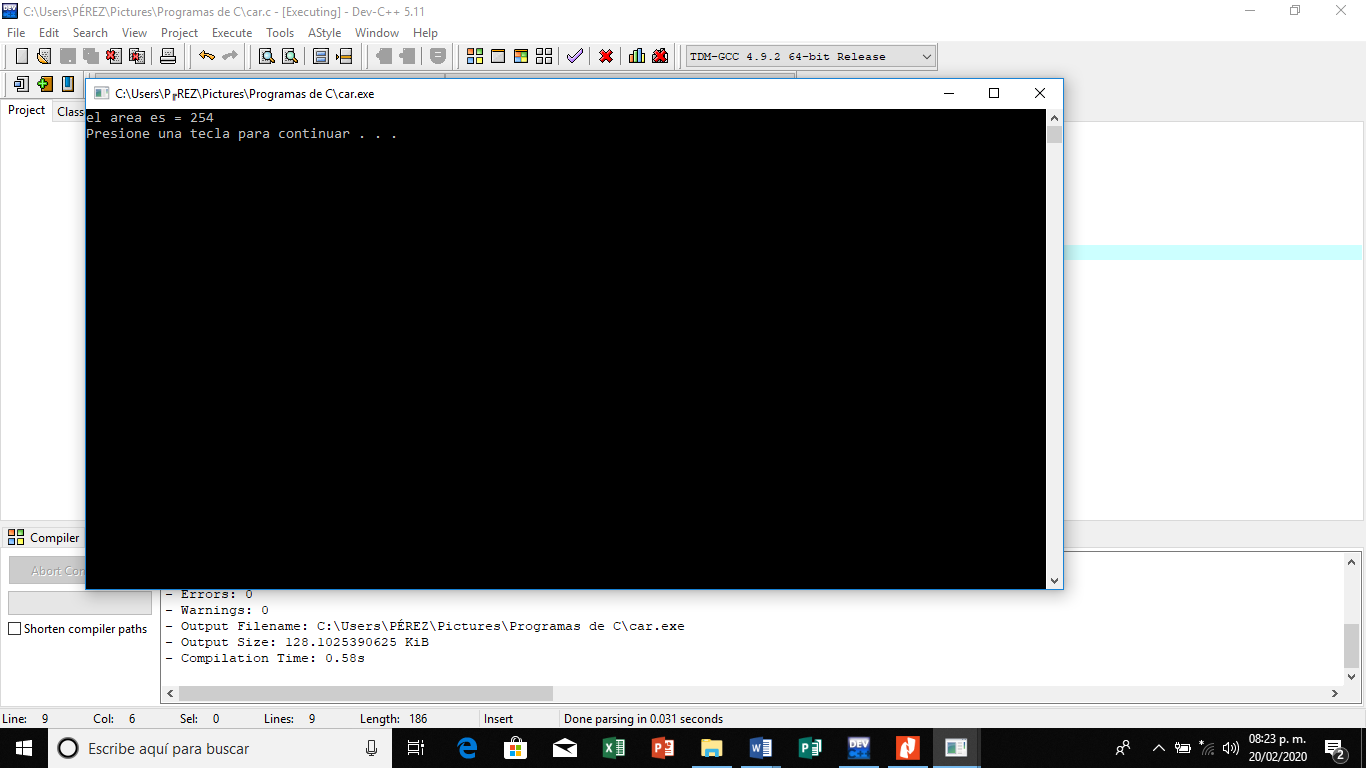


Ejercicio 3

En este ejercicio vamos a calcular el área de un circulo con ayuda de la formula área=pi\*r^2. En la parte de abajo se muestra cómo es que programe el ejercicio para que me diera este resultado.

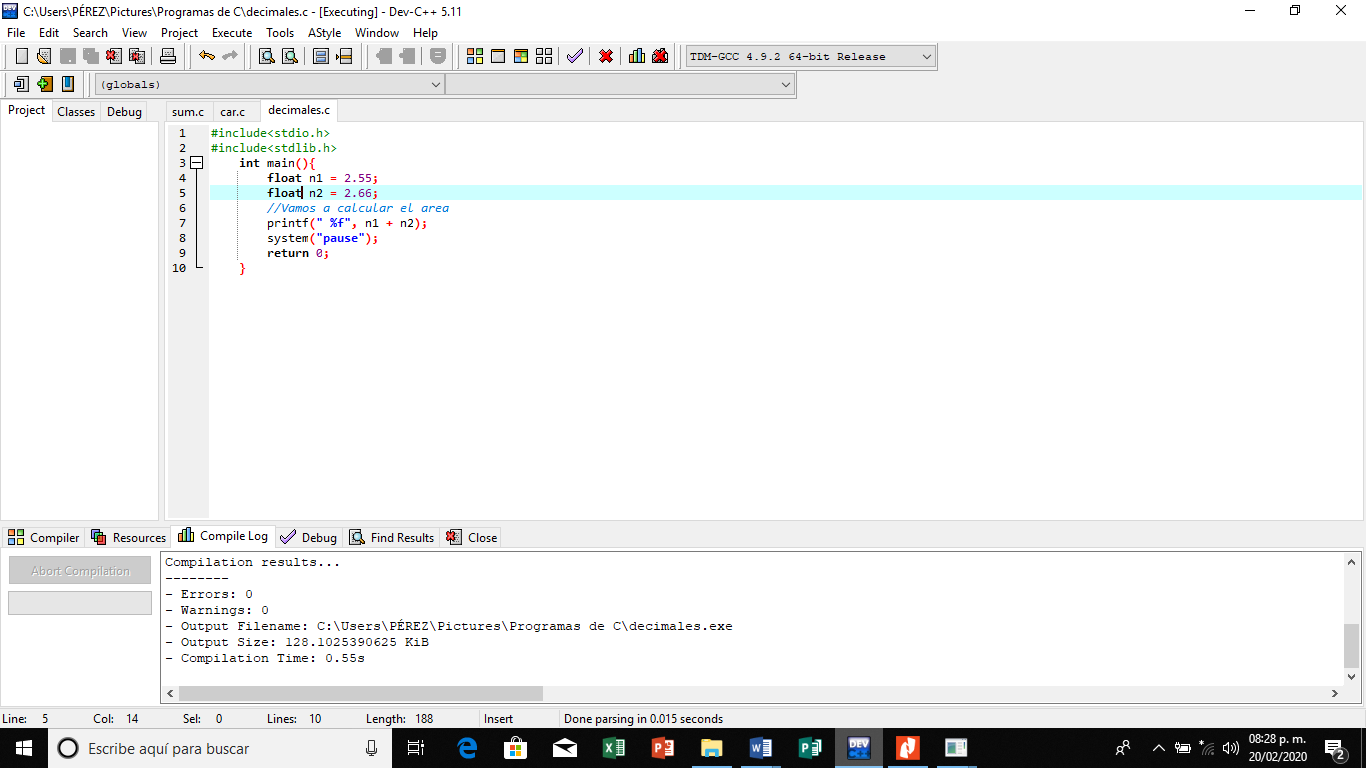


En la pantalla de abajo se muestra el resultado del área del círculo.

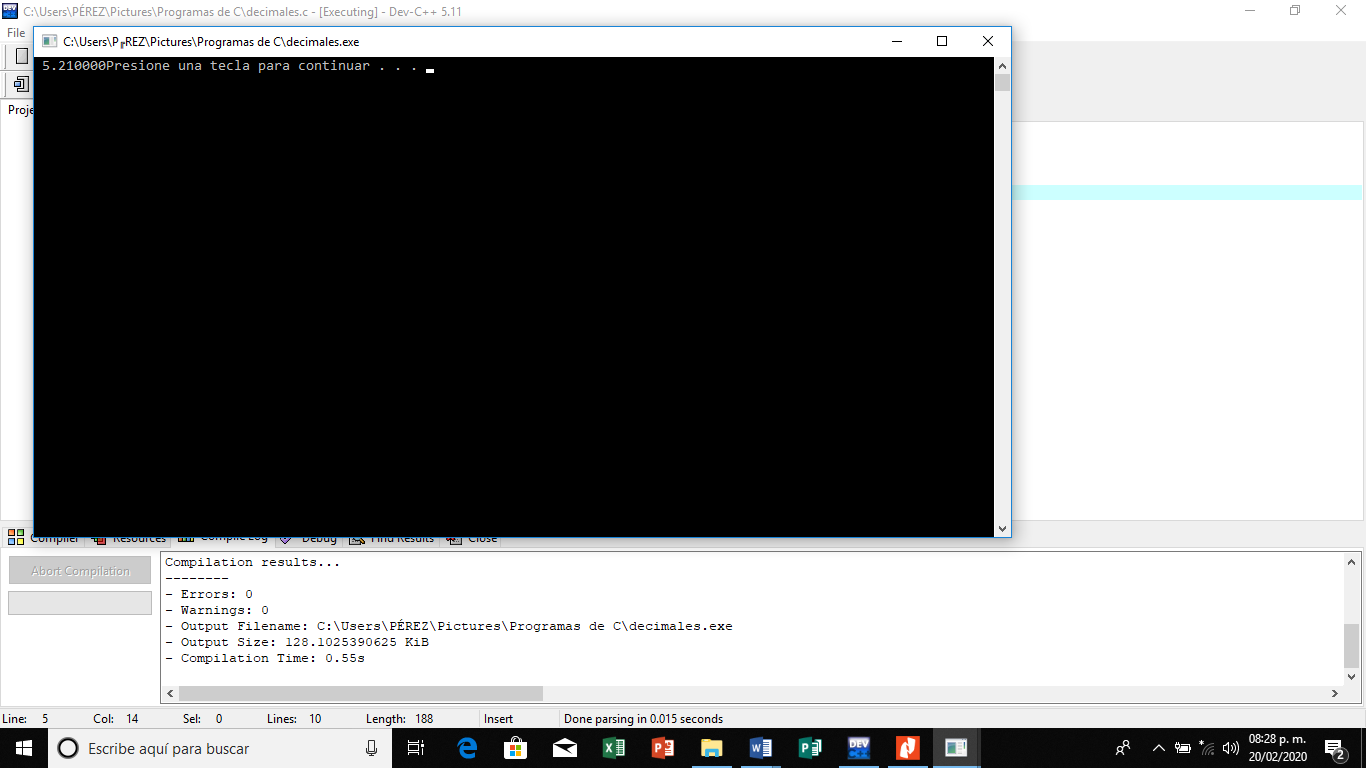


Ejercicio 4

En este otro ejercicio declaramos dos variables y les asignamos un número con punto decimal, vamos a realizar una suma, ocuparemos el signo %f que nos indica que se va a revisar un numero con punto decimal a continuación se muestra:

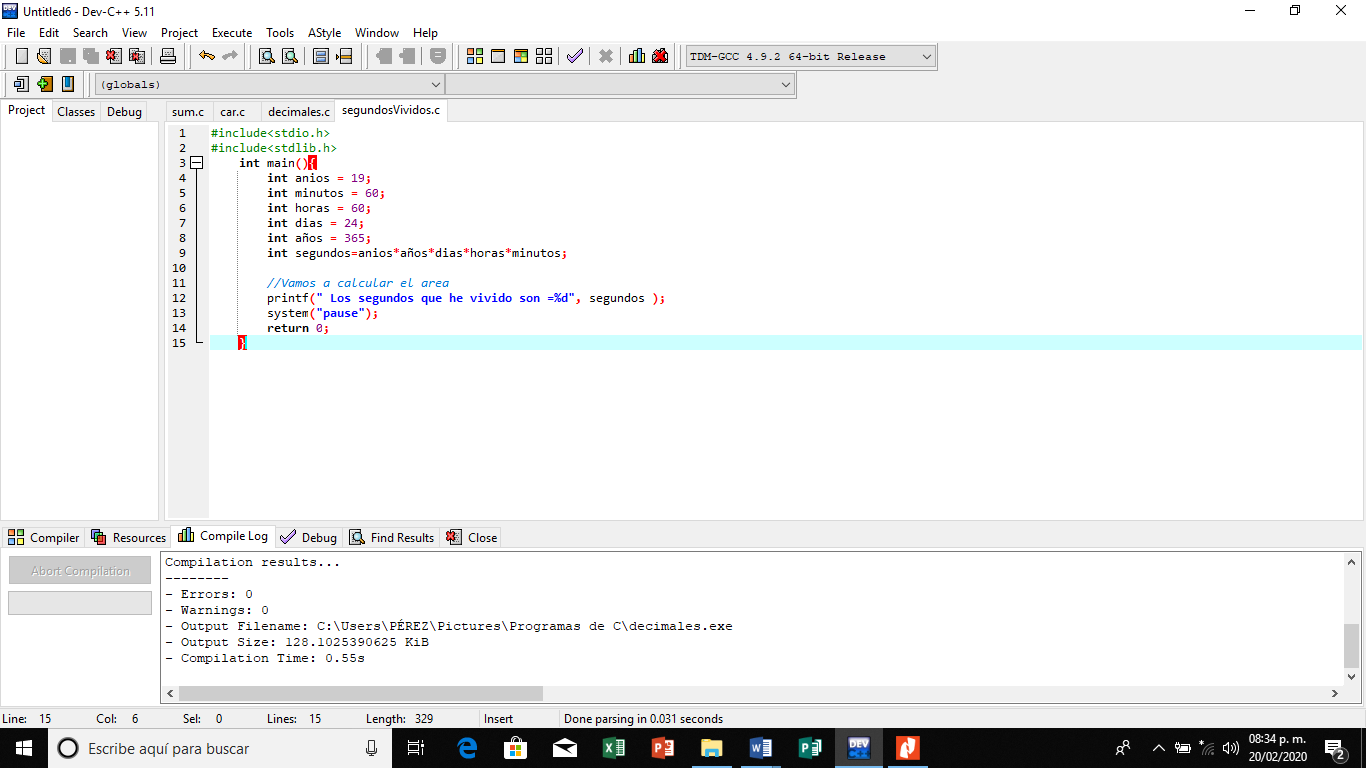


En la pantalla de abajo se muestra el resultado que es 5.21 nos da el resultado con punto decimal.

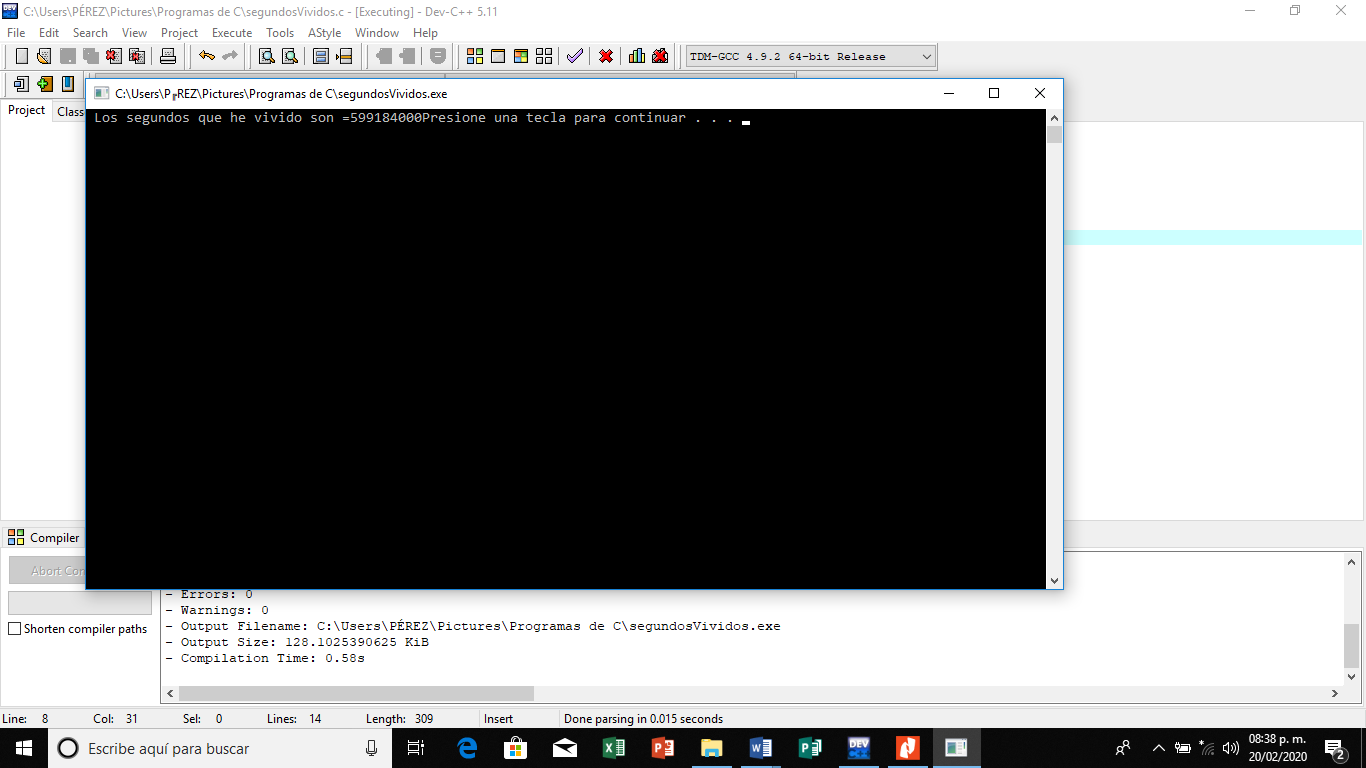


Ejercicio 5

Consistía en calcular mis segundos vividos tenía que declarar mis años, los minutos, horas, días y año para poder calcularlo para ser mas claro se muestra en la parte de abajo como es que se calculó.

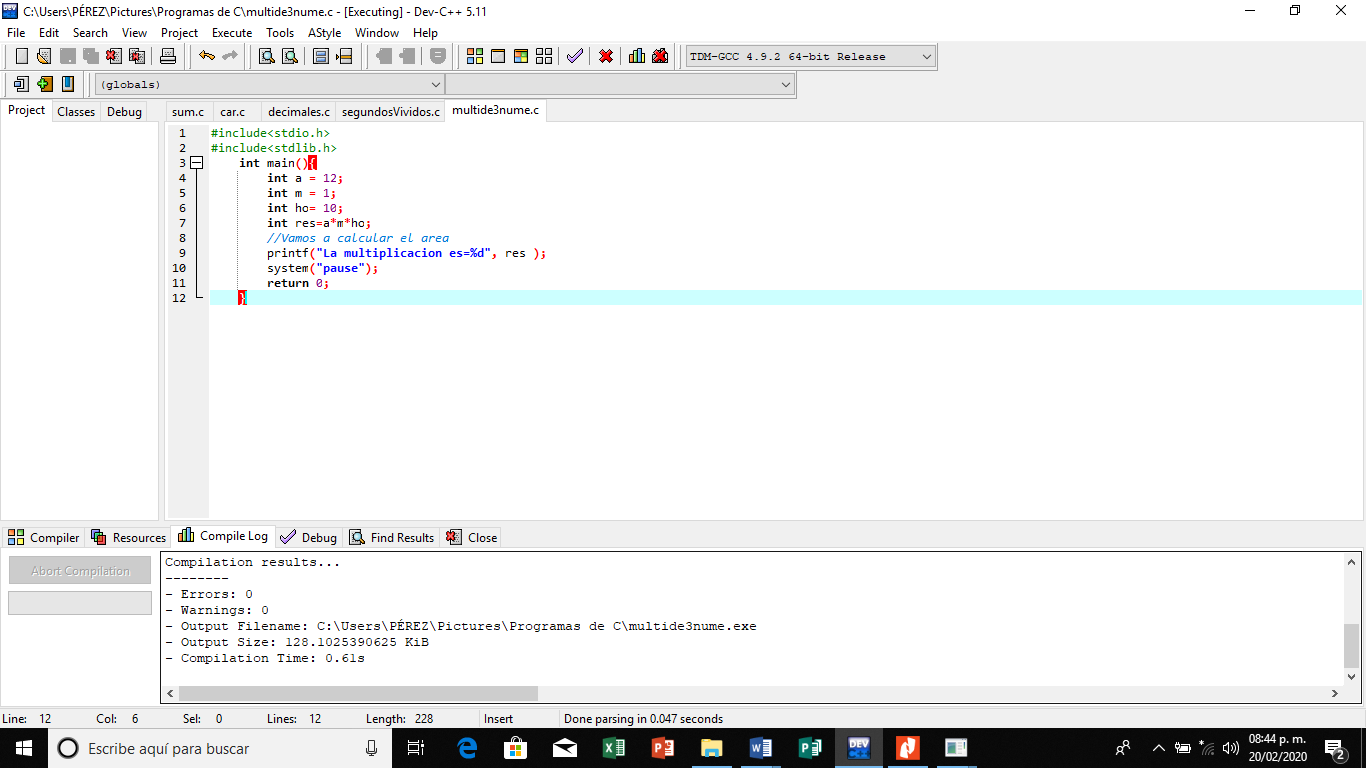


En la parte de abajo se muestra cuantos años he vivido según la operación realizada.

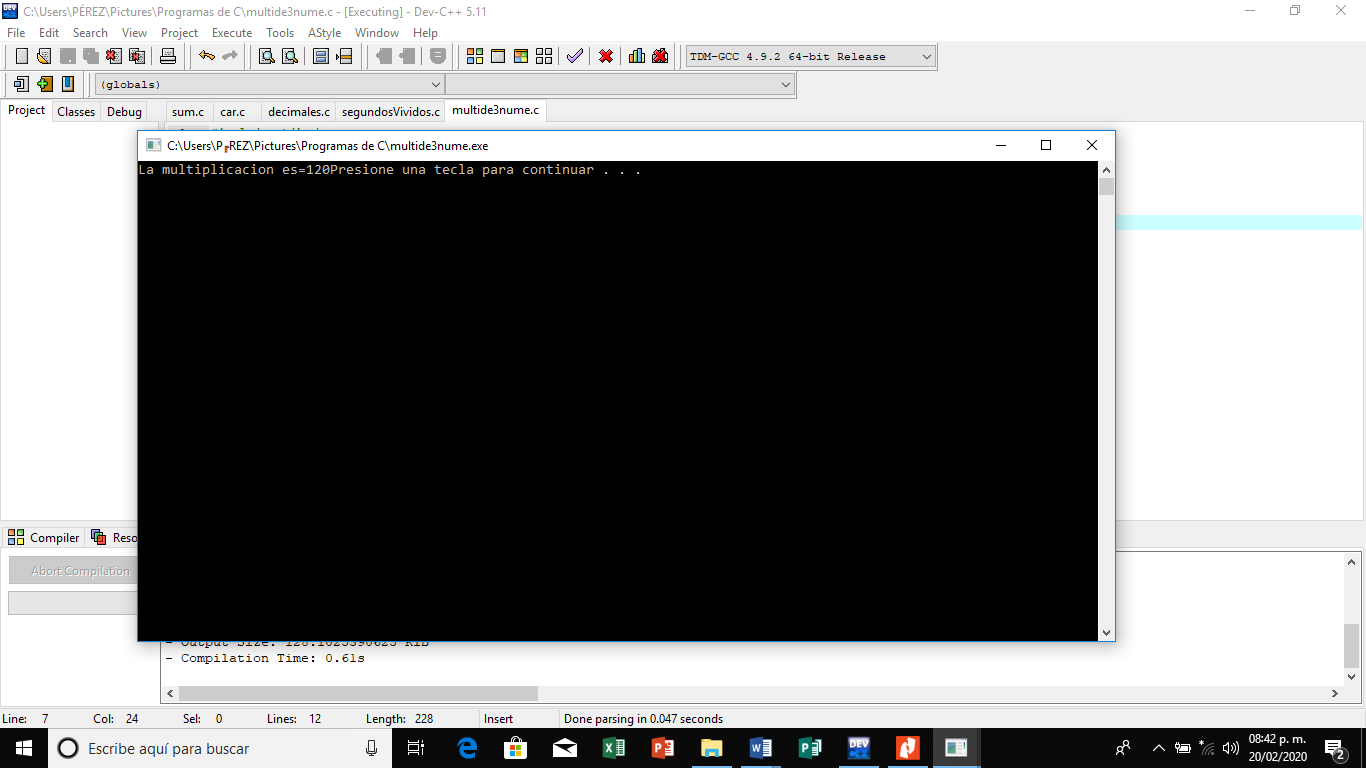


Ejercicio 6

Este ejercicio consistía en declarar 3 variables con un tipo de dato entero y multiplicarlos en la pantalla de abajo se muestra la multiplicación de estas.

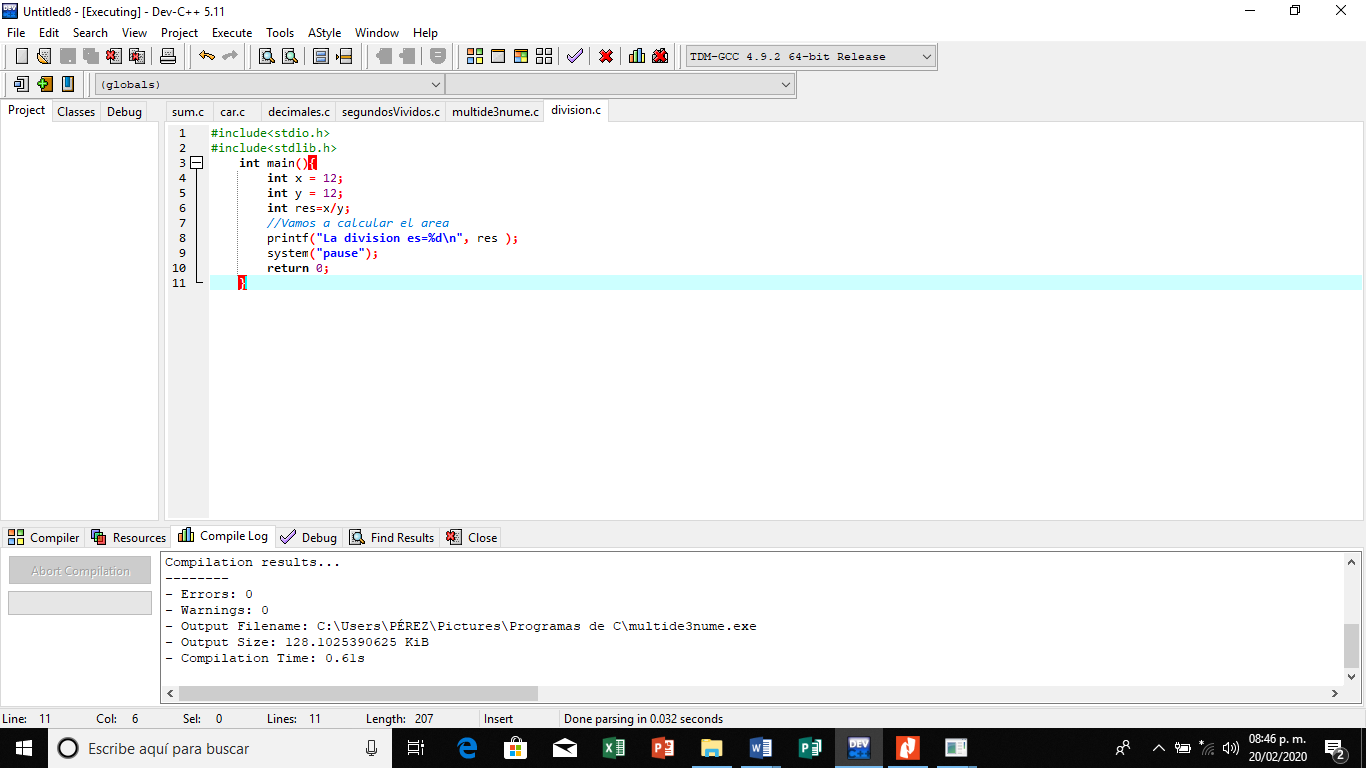


En la pantalla de abajo se muestra el resultado de la multiplicación.

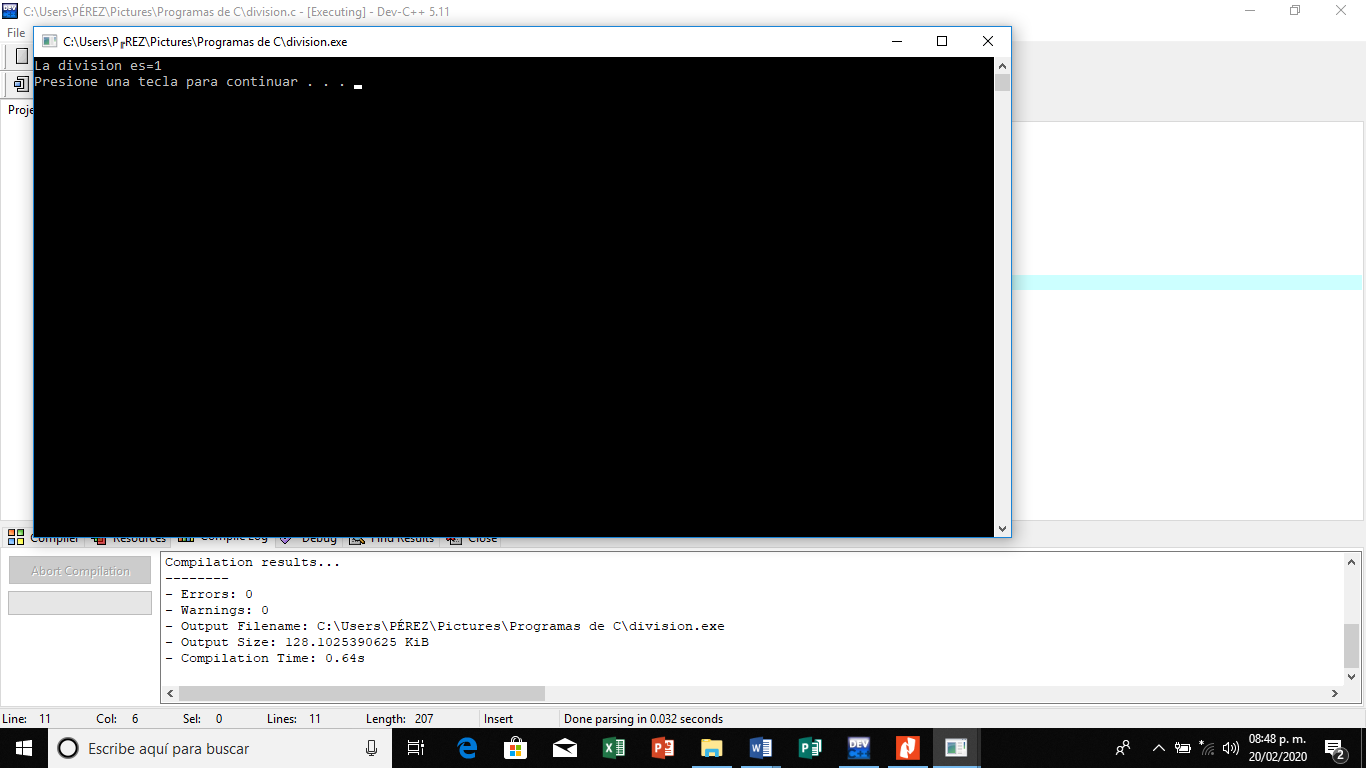


Ejercicio 6

Este ejercicio consistía en calcular la división de un número y otro pero estos ya tenían que estar declarados por medio de una variable.



En la pantalla de abajo se muestra el resultado de la división

.

**4.Operador\_Incremento**

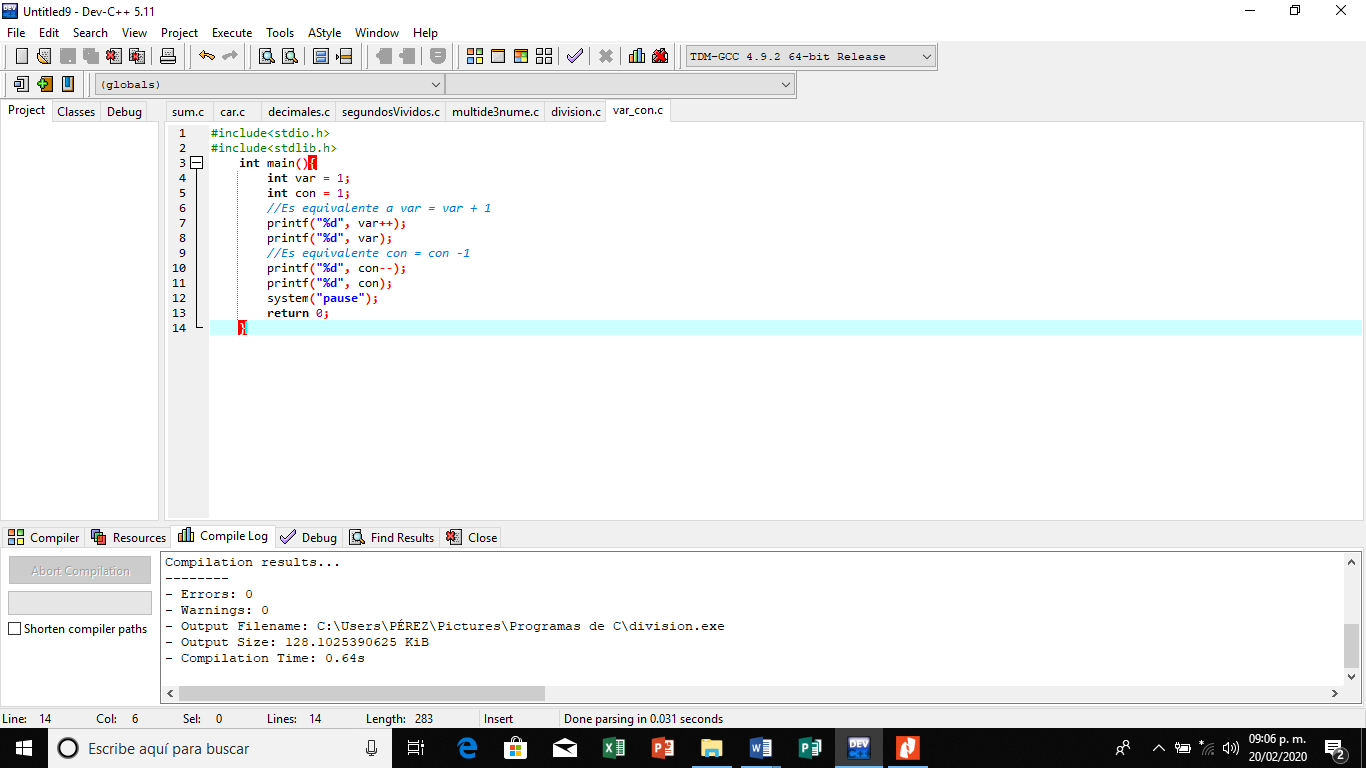
En este ejercicio vamos a mostrar como utilizamos el valor actual de la variable y después le hace un

incremento/decremento

Ejercicio 1

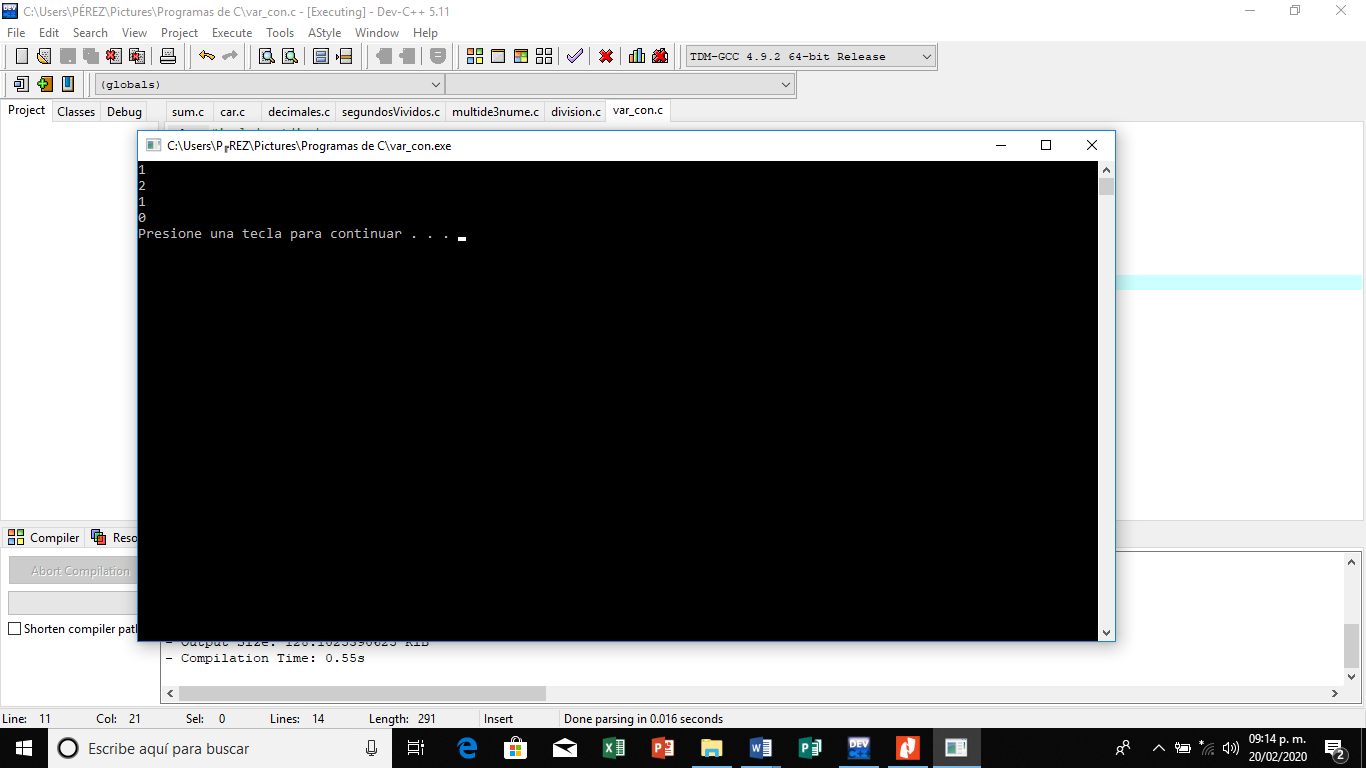
En este ejercicio se muestra cómo es que incrementamos la variable var y disminuimos la variable con.

La variable var vale 1 cuando se ejecuta la línea donde esta incrementa la variable ahora vale 2 y abajo se imprime la variable incrementada.



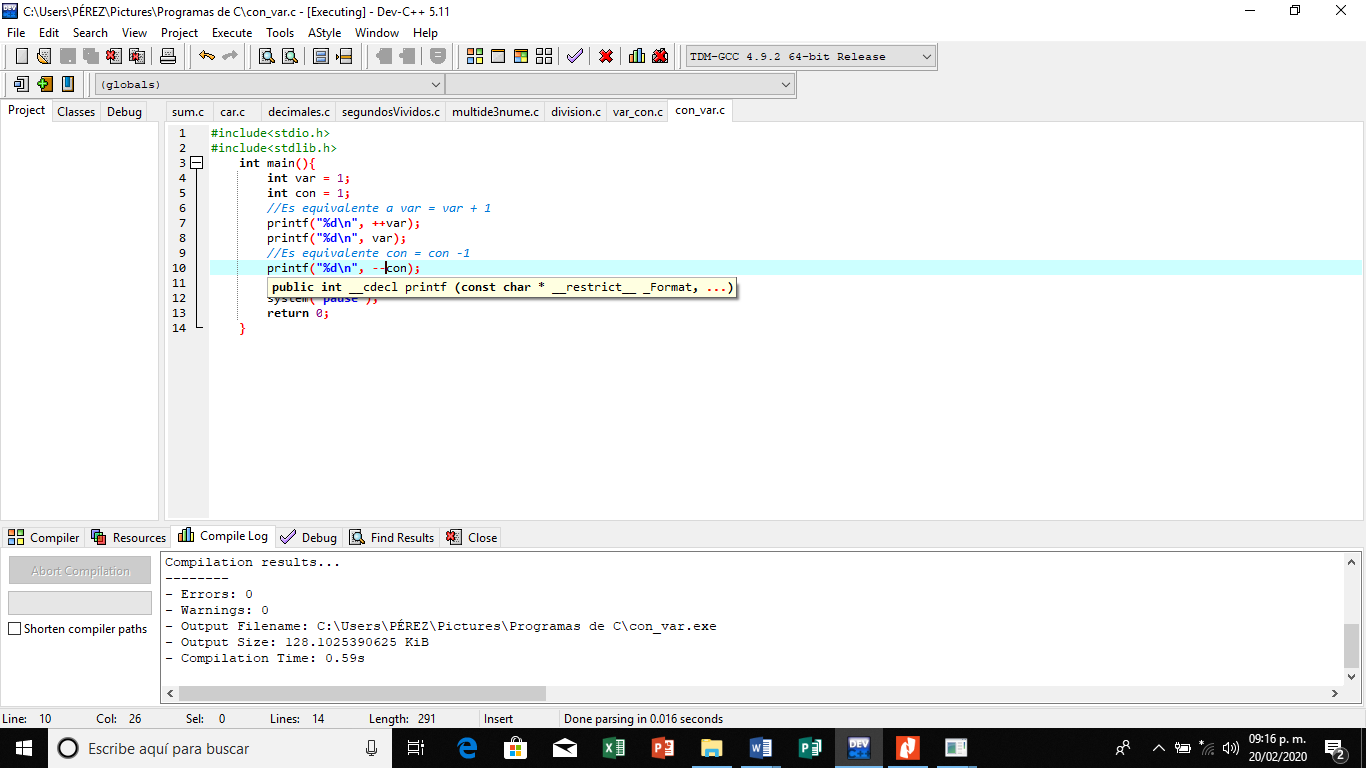
La variable con vale 1 cuando se ejecuta la línea donde esta disminuye este vale ahora 0 y se imprime después 0

En la pantalla de abajo se muestra el resultado de este ejercicio.

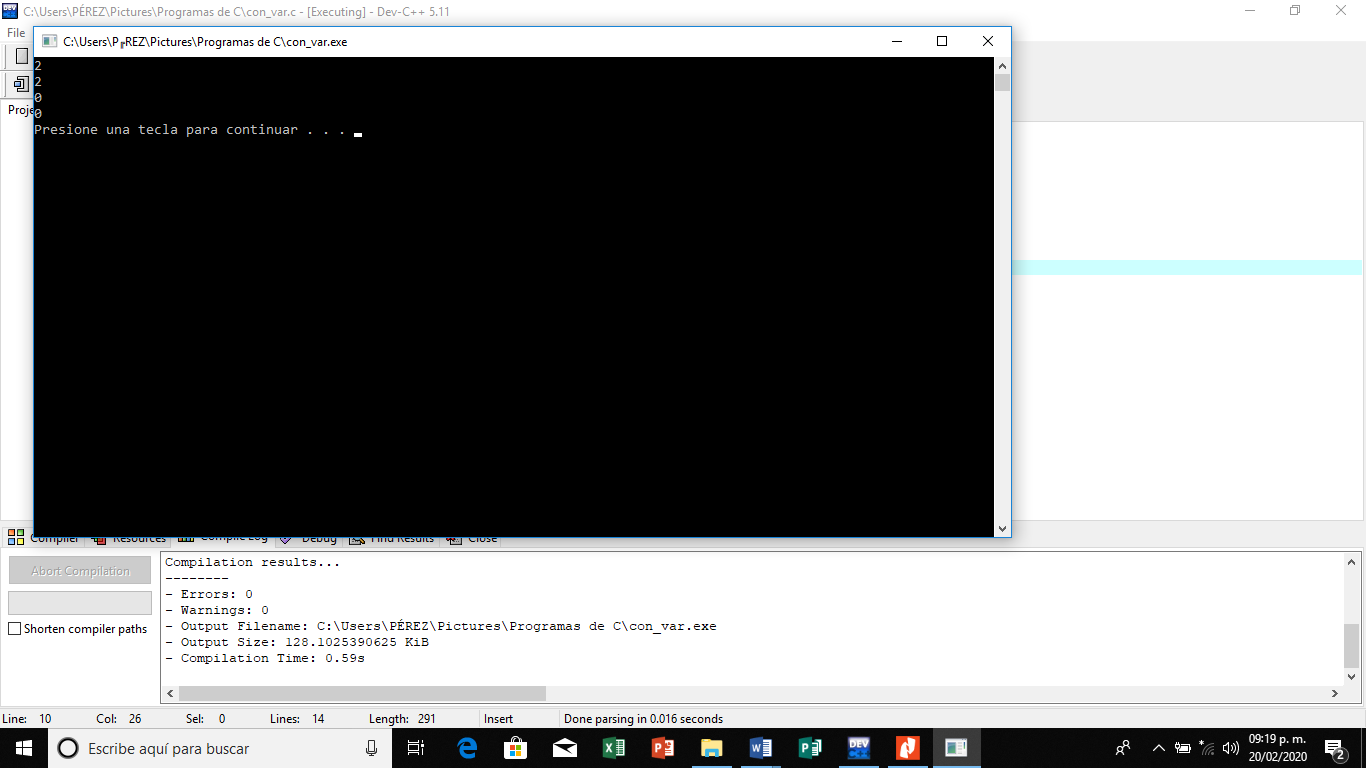


Ejercicio 2

En este ejercicio lo que realizamos es cambiar los signos de incremento y decremento para observar que sucede lo que sucede es que primero se incrementa var y luego se imprime dos veces el incremento eso mismo pasa con el disminuye.

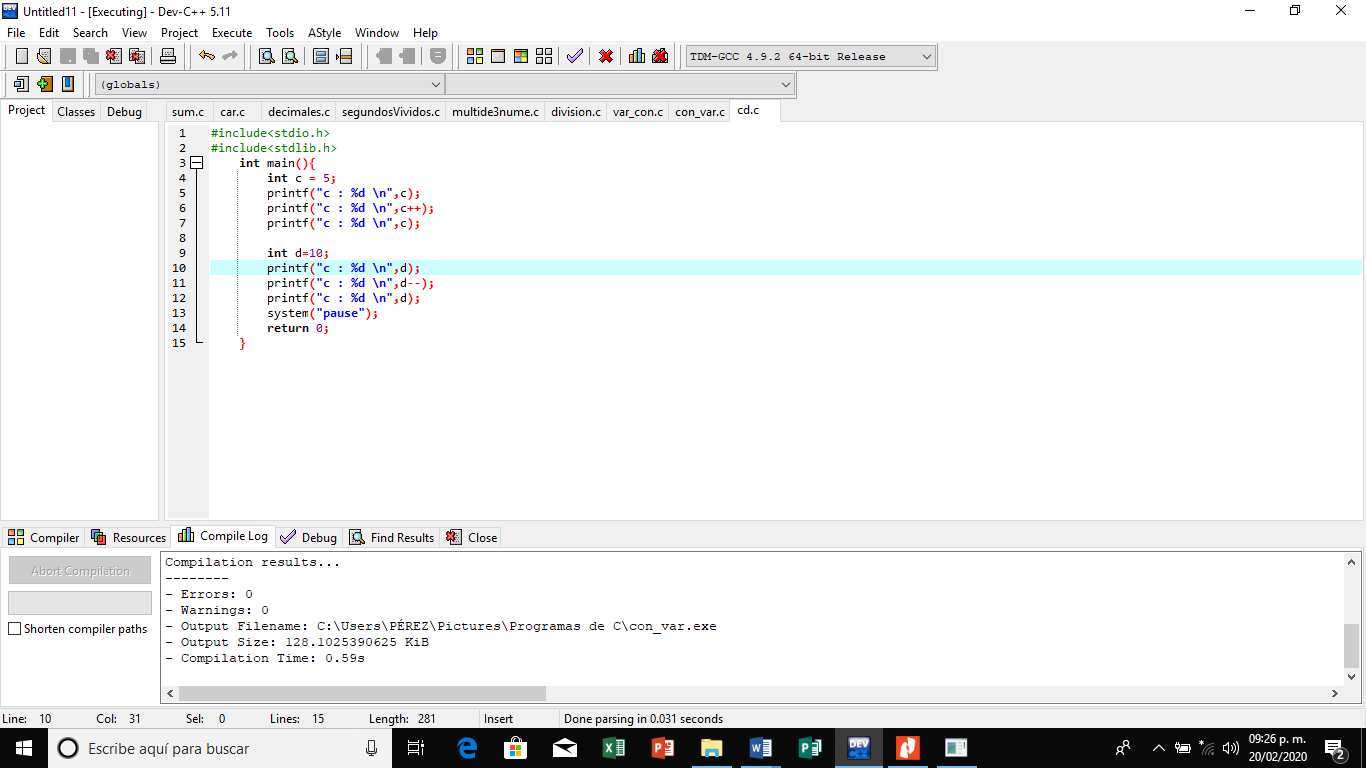


En la pantalla de abajo se muestra el resultado de lo que está arriba.



Ejercicio 3

En este ejercicio declaramos una variable con un valor de 5 para esto voy a mostrar cómo es que primero la imprimo como que valor da cuando realizo un incremento y mandar el valor incrementado. Posteriormente hacemos lo mismo con 10 pero ahora lo decrementos de tal manera que podamos ver su valor y como es que se comporta.

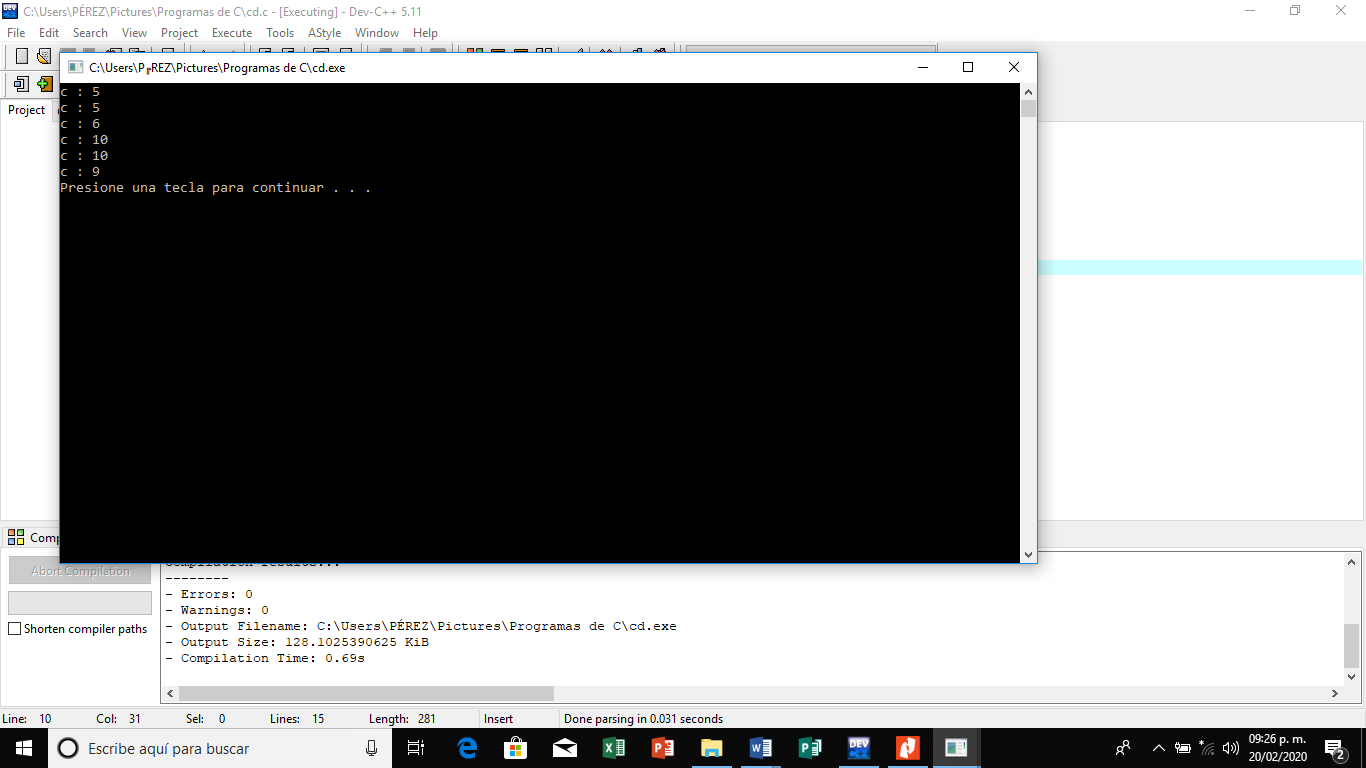


Se imprime c con el incremento que se realizó en este caso es 6.

Se vuelve a imprimir el 5 y después hace el incremento

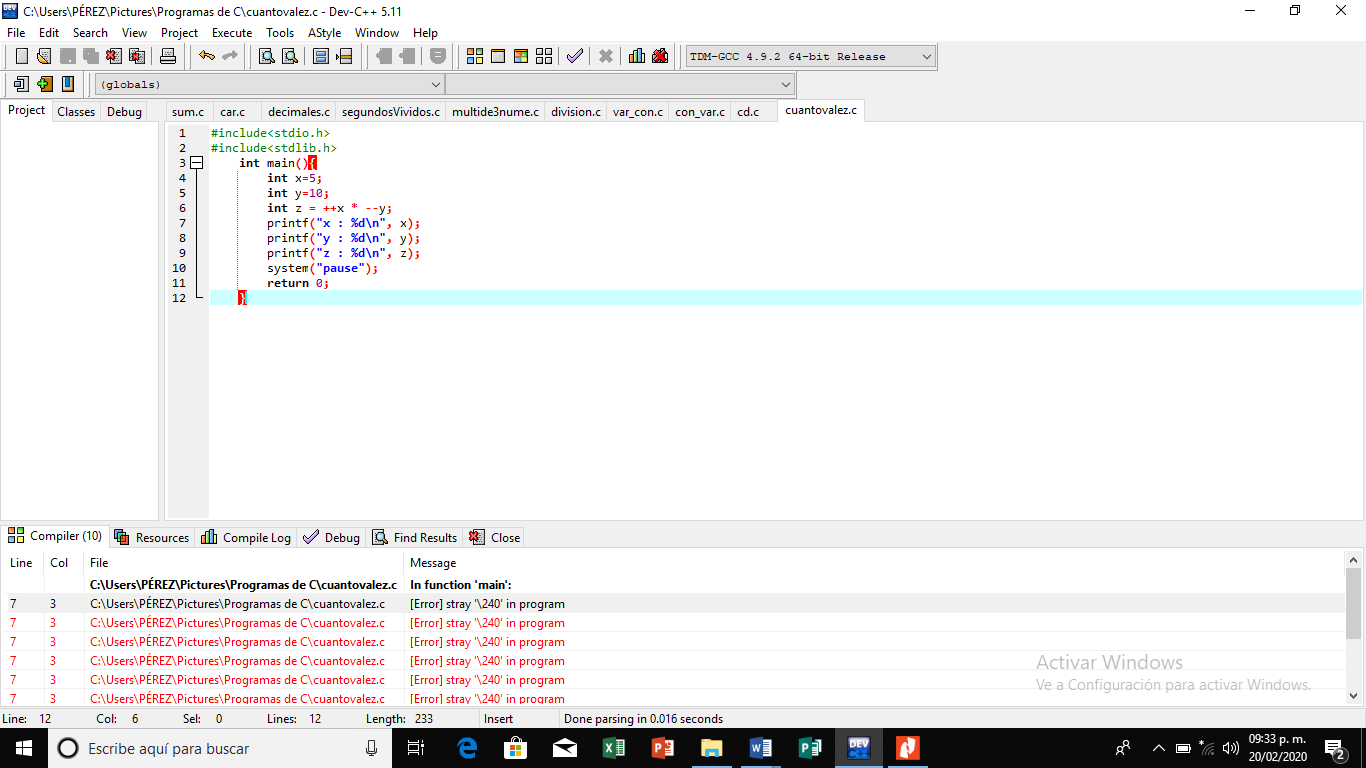
Se imprime el valor de c que es 5

Algo similar pasa con el decremento en la pantalla se abajo se muestra el comportamiento de este resultado.



Ejercicio 4

En el ejercicio de abajo era definir dos variables una con variable x y otra con variable y para poder sacar el valor de z la operación es se incrementa z y se multiplica por y después esta decremento a 9 la respuesta de esta operación es 60.



**5.Precedencia\_de\_operaciones**

PRECEDENCIA DE OPERADORES

¿Qué es un operador?:  
  
 *Un operador es un símbolo (+, -, \*, /, etc) que tiene una función predefinida (suma, resta, multiplicación, etc) y que recibe sus argumentos de manera infija, en el caso de tener 2 argumentos de manera prefija o postfija.*

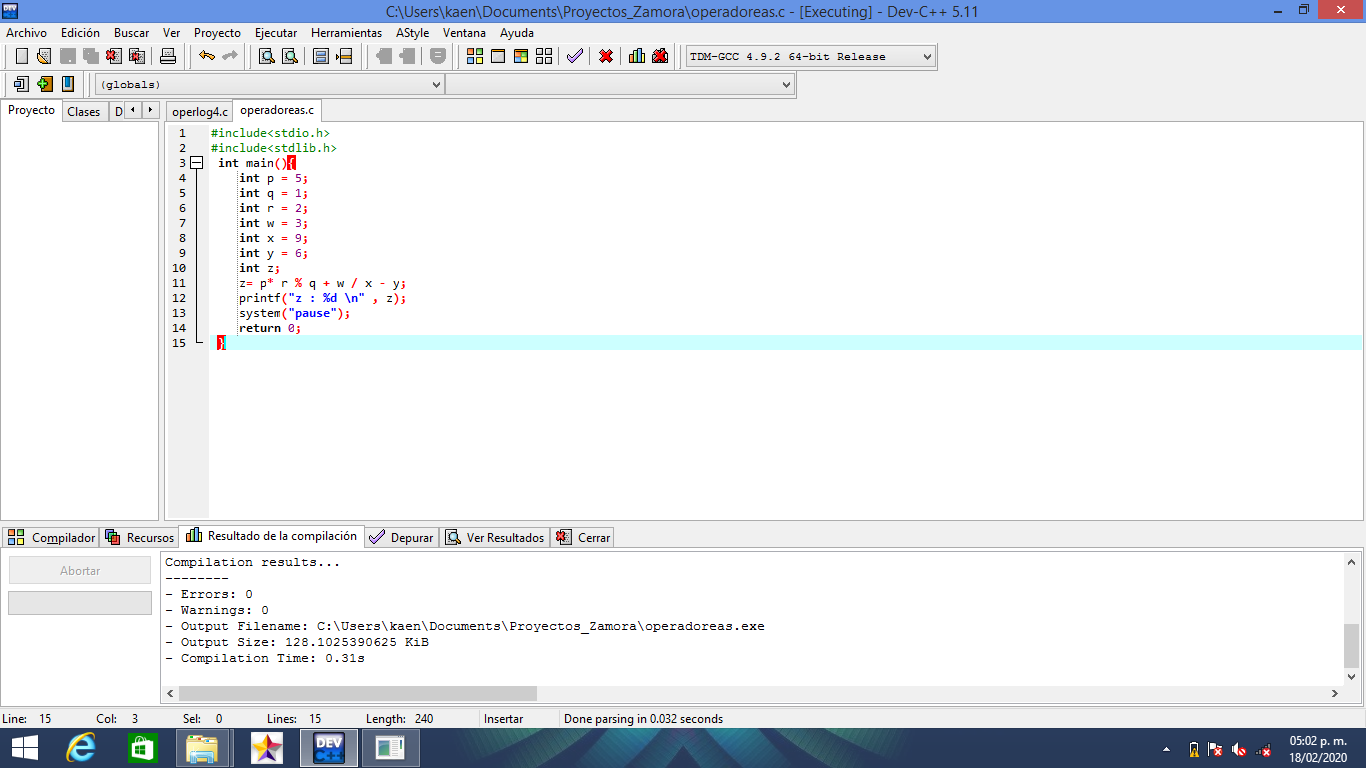
**TIPOS DE OPERADORES**

* Operadores aritméticos
* Operadores relacionales
* Operadores lógicos
* Operadores a nivel de bit (bitwise operators)
* Operadores especiales  
    
    
  Operadores aritméticos  
    
   Los operadores aritméticos nos permiten, básicamente, hacer cualquier operación aritmética, que necesitemos (ejemplo: suma, resta, multiplicación, etc). En la siguiente tabla se muestran los operadores de los que disponemos en C y su función asociada.

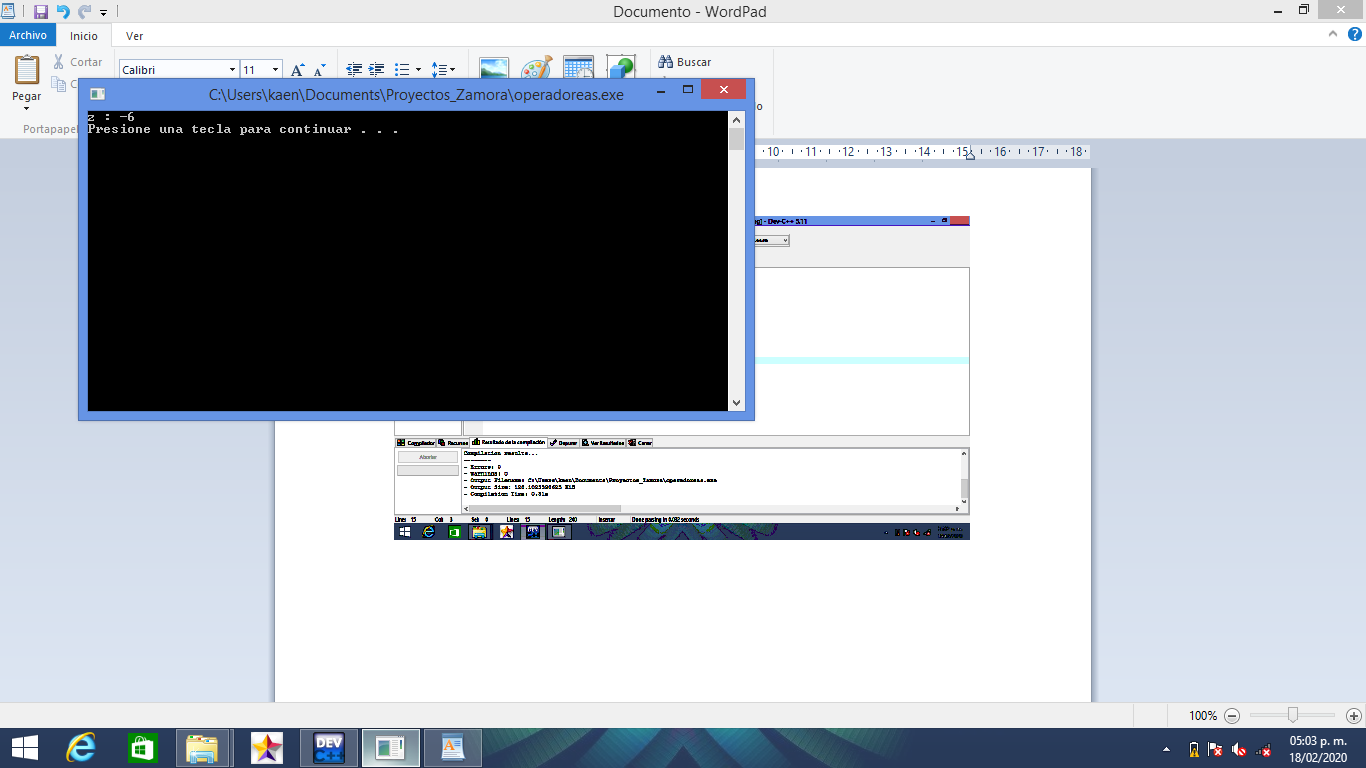
|  |
| --- |
| **Tabla:** Operadores aritméticos |
| |  |  |  | | --- | --- | --- | | *Operador* | *Acción* | *Ejemplo* | | - | Resta | x = 5 + 3; // x vale 2 | | + | Suma | x = 2 - 3; // x vale 5 | | \* | Multiplicación | x = 2 + 3; // x vale 6 | | / | División | x = 6 / 2; // x vale 3 | | ·/. | Módulo | x = 5 % 2; // x vale 1 | | -- | Decremento | x = 1; x--; // x vale 0 | | ++ | Incremento | x = 1; x++; // x vale 2 | |

Ejercicio 1

Bueno ya con ayuda de la explicación de arriba hice un ejercicio que consta de declarar 6 constantes con un valor ya definido y una variable que recordemos que el valor de la constante ya está declarado y el valor de la variable puede cambiar sobre la ejecución del programa. Tenemos que encontrar el valor de z con ayuda de los operadores y ya con respecto a esto vamos a imprimir su resultado.

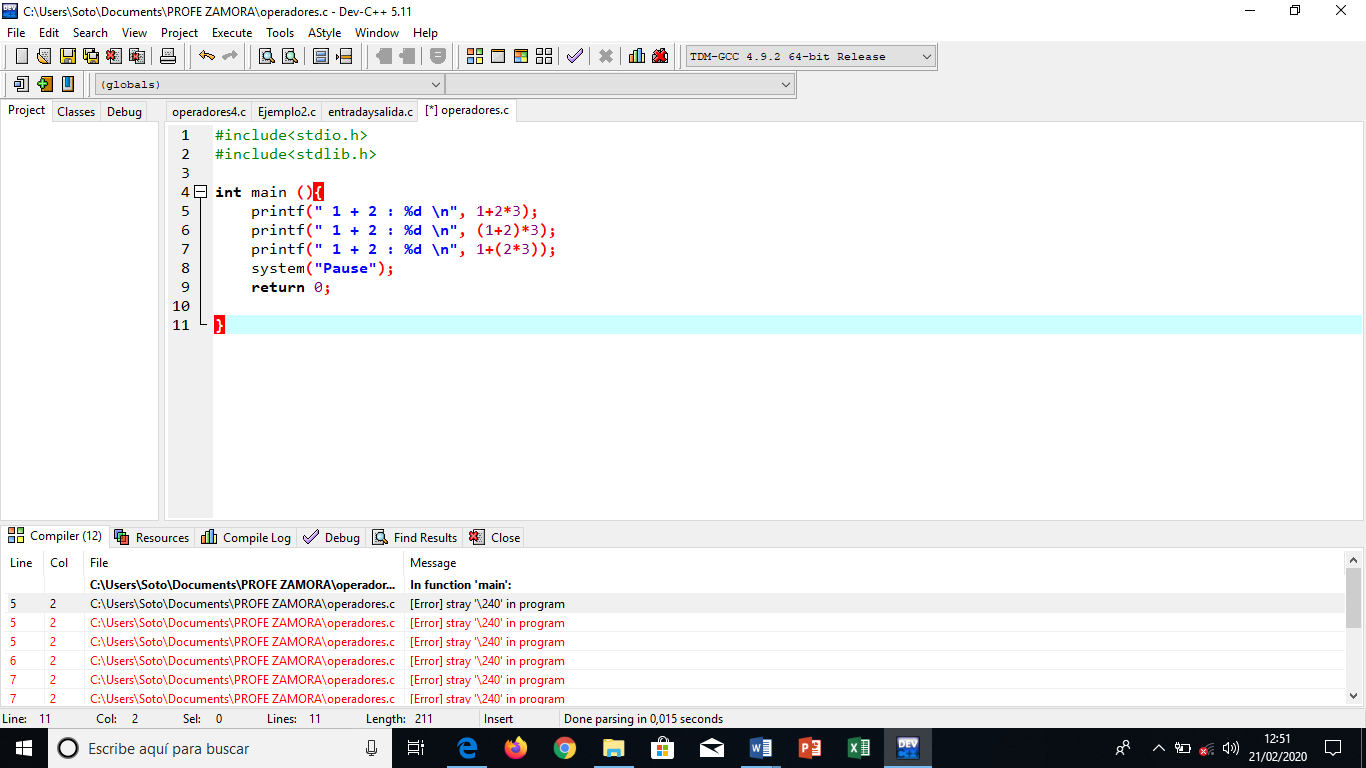


El resultado de esta operación es -6 y se muestra en la pantalla de abajo y así es como hacemos uso de los operadores recordemos que las operaciones se resolverán por prioridad.



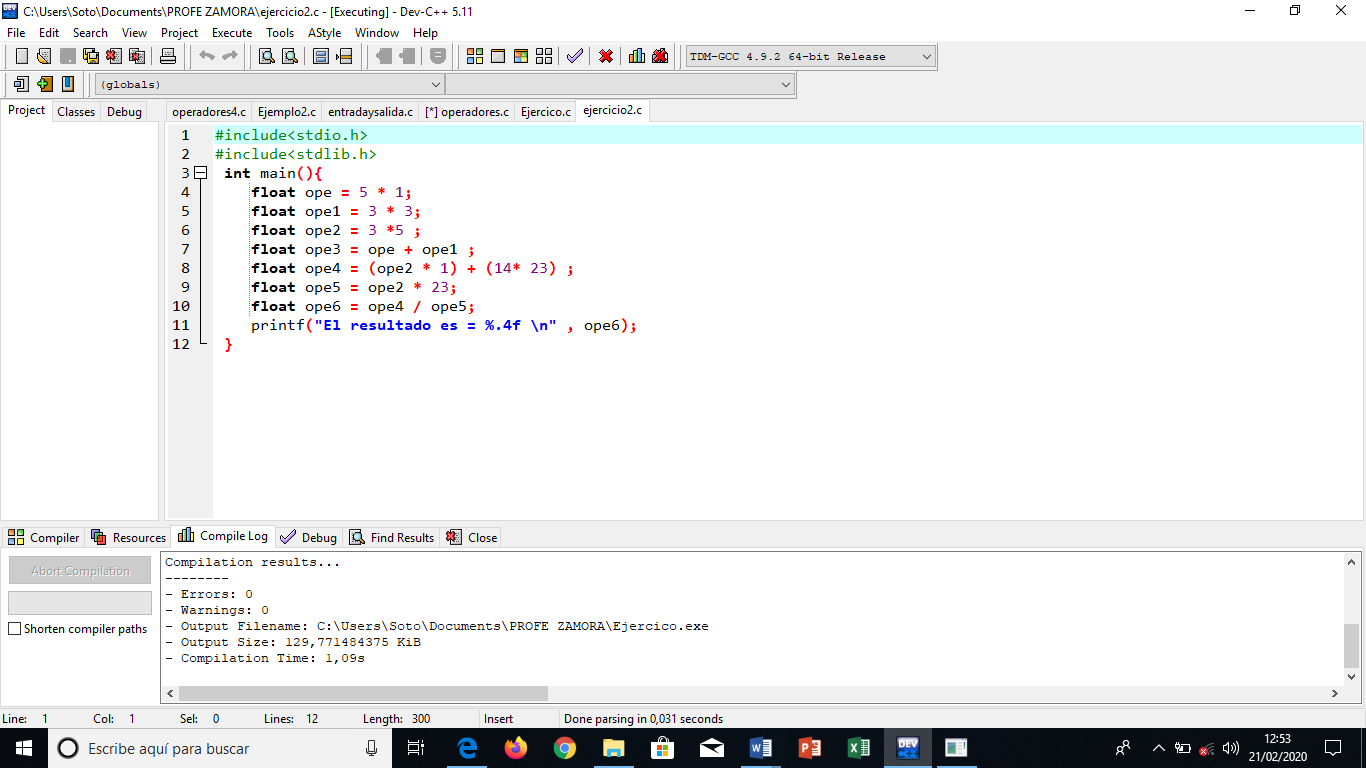
Ejercicio 2

En este ejercicio era realizar la suma y multiplicación de números, en este ejercicio vimos como los operadores se comportan.

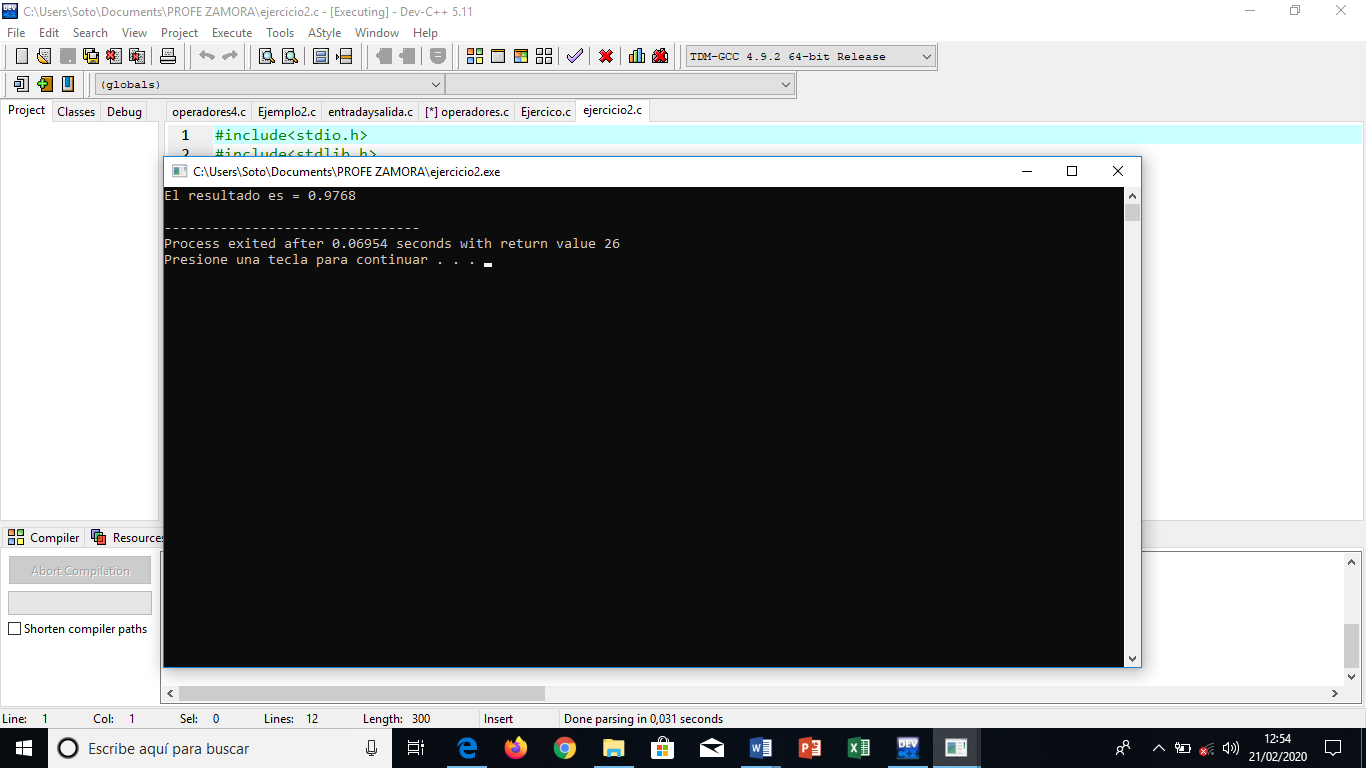


Ejercicio 3

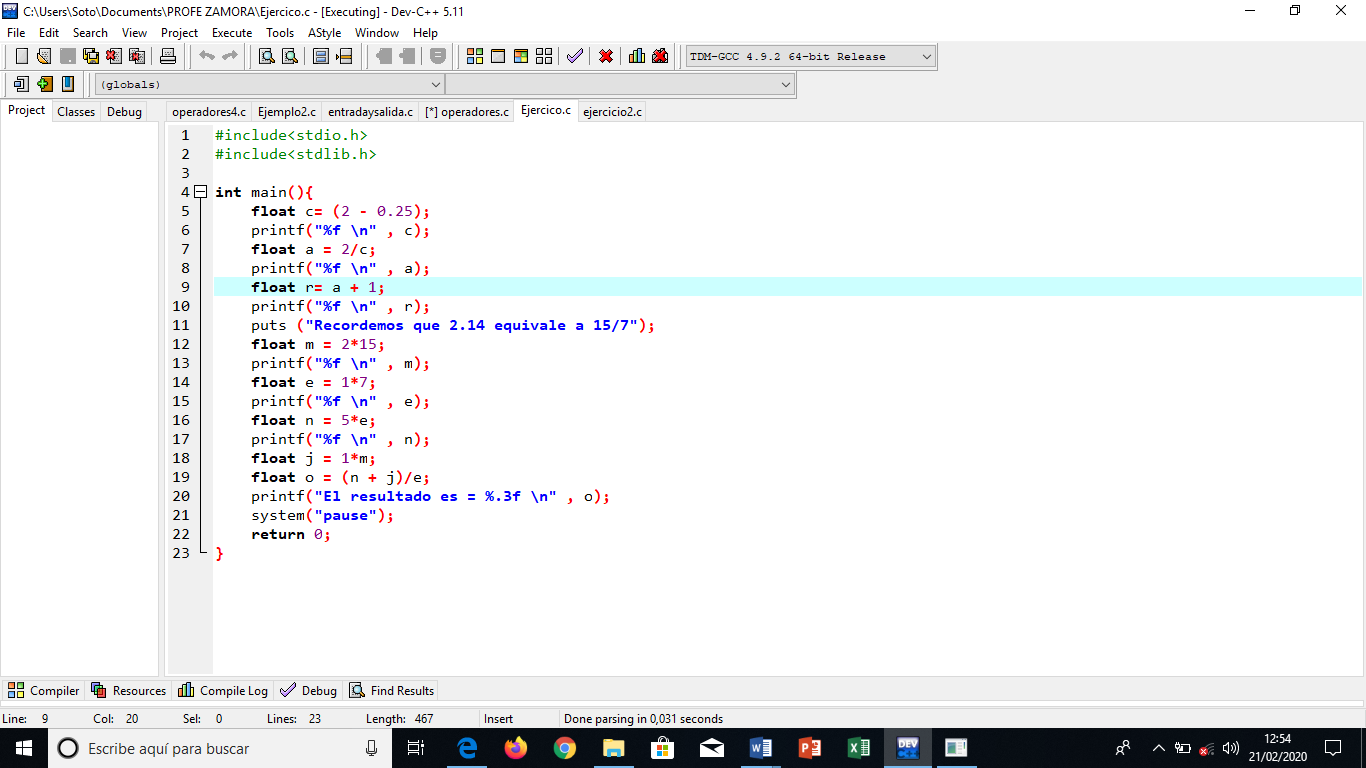
En el ejercicio de abajo se muestra una operación que desviemos de resolver como resultado la operación esta tenía como resultado 0.9768 así que realice la operación por partes para darle solución, cree 7 variable que me ayudaran a guardar los valores de las operaciones y hacerla por partes para no confundirme a la hora de realizarlo.



En la pantalla de abajo se muestra que si da el mismo resultado el cual nos pide el problema.

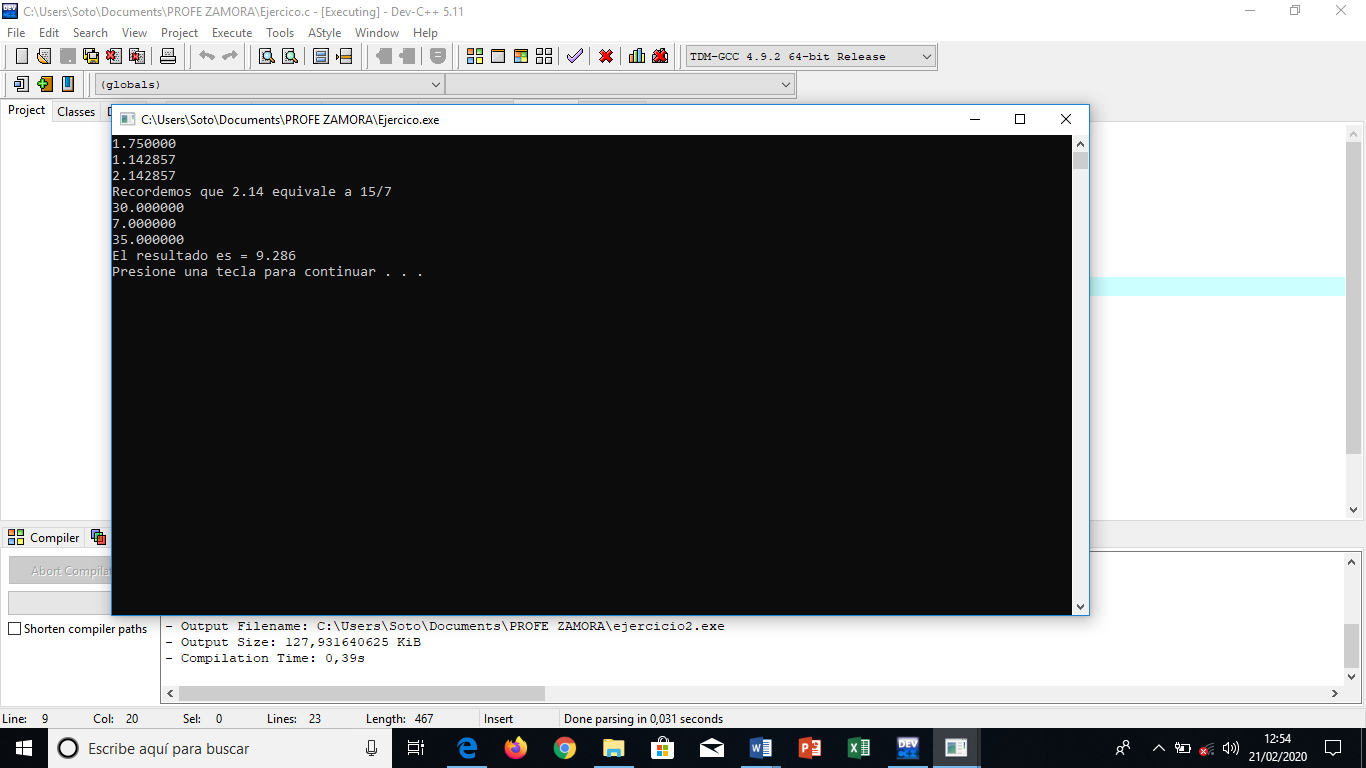


Ejercicio 4



En la pantalla de al lado se muestra la segunda operación que nos ayuda a poder encontrar el resultado de otra operación que debe de dar como resultado 9.28 para esto utilice varias variables para poder guardar el valor y después hacer los cálculos correspondientes para sacar su valor

En la parte de abajo se muestra la pantalla que con el resultado de cada operación y se ve que si llegamos al resultado esperado.



**6.Operadores\_logicos\_y\_de\_relación**

Nos operadores nos proporcionan un resultado a partir de que se cumplan o no una cierta condición, produce un resultado booleano, y sus operadores son también valores lógicos o asimilables a ellos.

En este ejercicio vamos hacer uso de los operadores lógicos que nos permiten realizar condiciones para que una cierta condición se cumpla de una manera que la condición sea falsa o verdadera en la parte de abajo se muestra cómo es que funcionan.

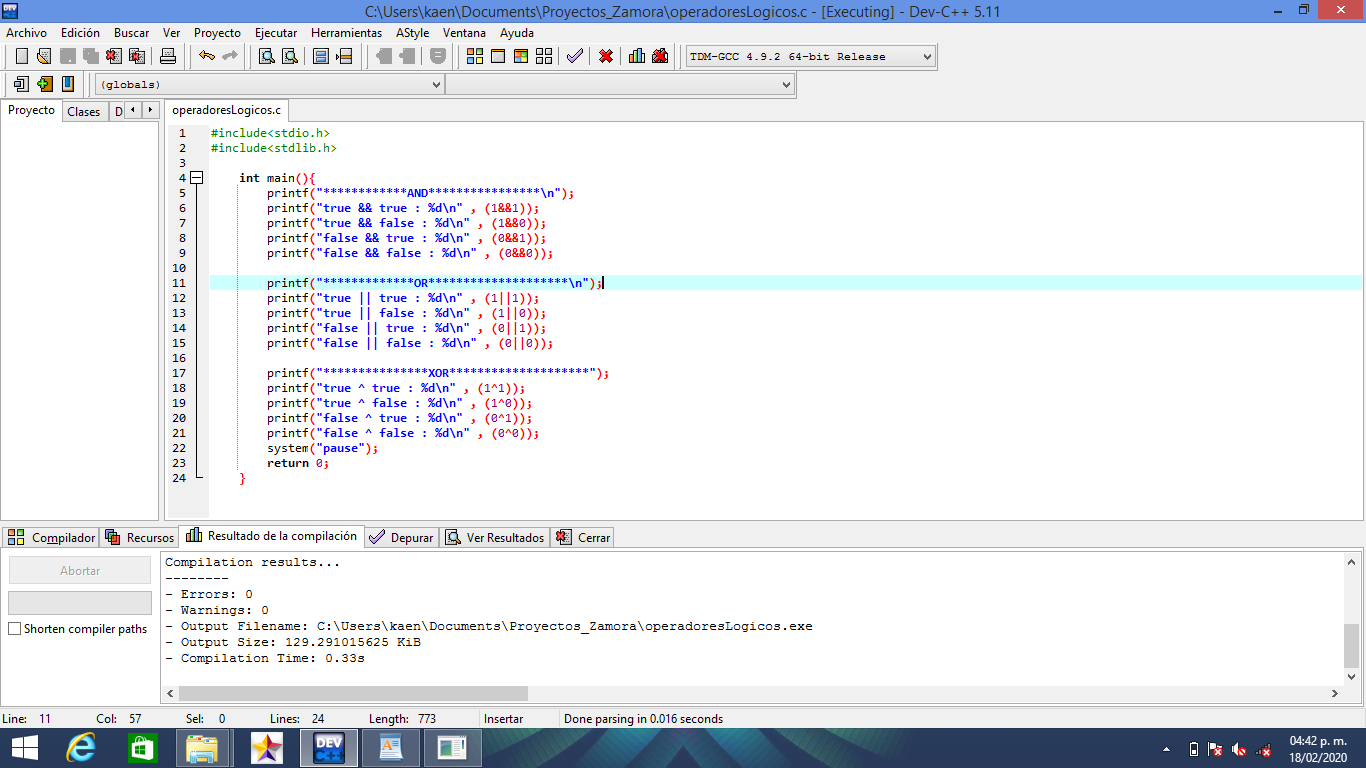
&&=en AND devuelve un valor lógico que nos indica que ambas operaciones devén de ser verdaderas en caso contrario el resultado es falso.

||=es un operador binario devuelve true si alguna de las operaciones es cierta en caso contrario devuelve false.

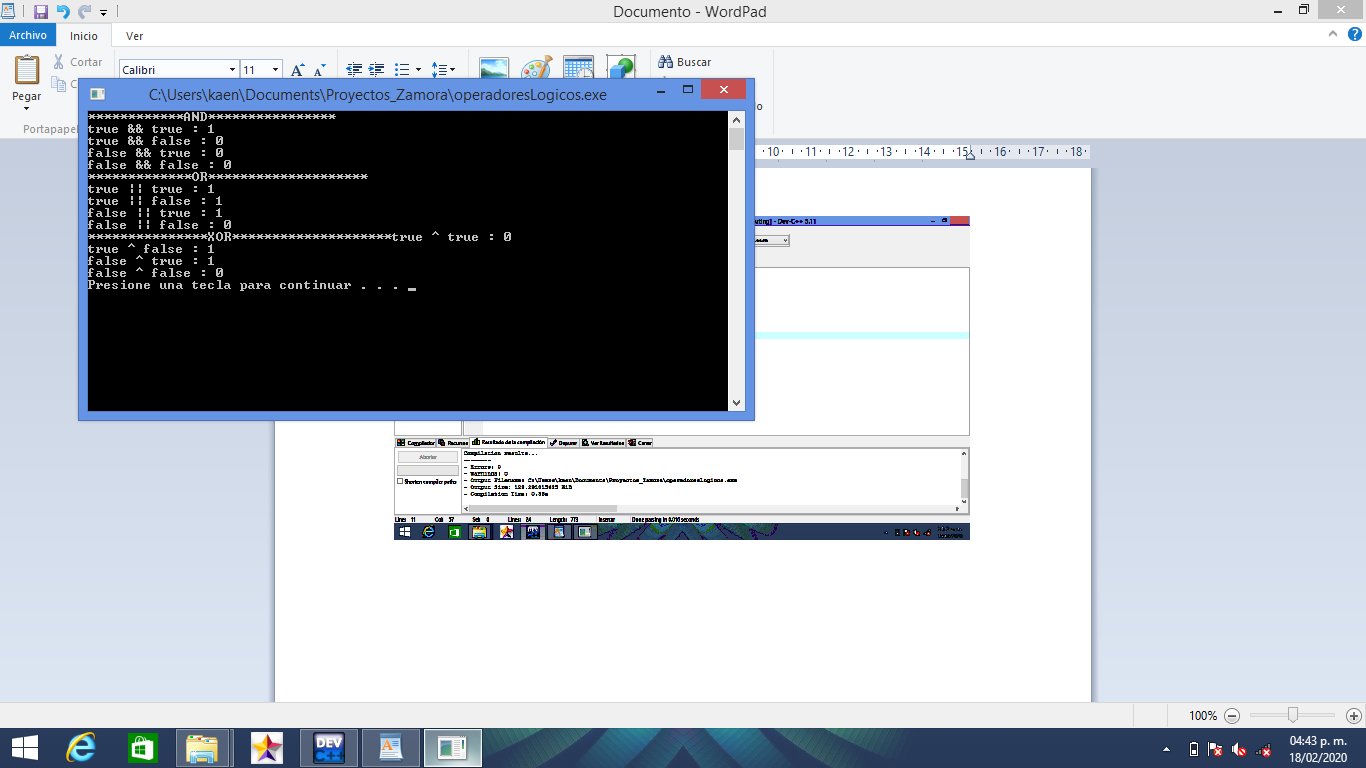
^=operador lógico OR exclusivo entre bits.

Ejercicio 1

En este ejercicio se muestra un ejemplo del uso de los operadores lógicos utilizamos las librerías y el método principal que nos ayudara a poder ejecutar lo que contenga dentro, con ayuda del printf vamos a mandar a imprimir y con ayuda de %d vamos a recibir el valor según corresponda si se cumple o no.

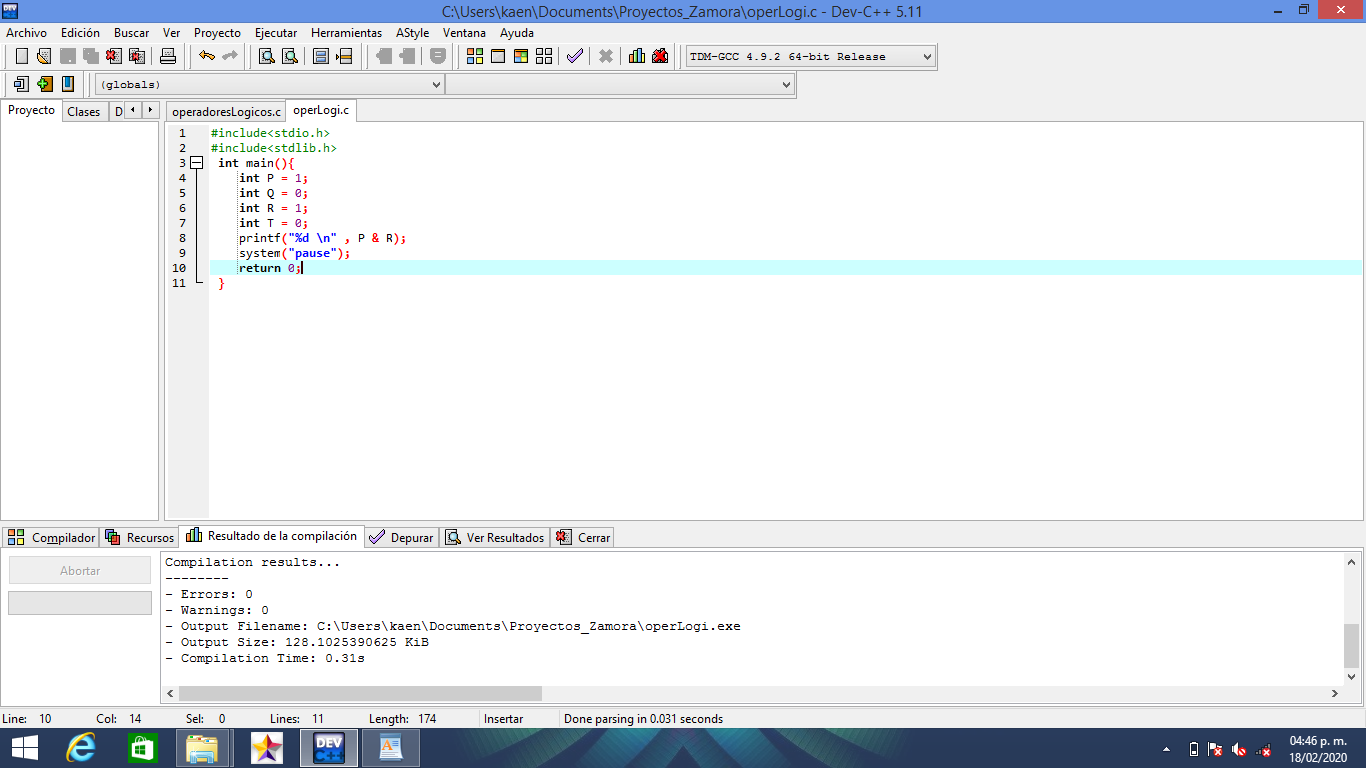


En la pantalla se abajo se muestra lo que programe si sale 1 es que es cierto cuando sale cero es que el falso y no cumple.

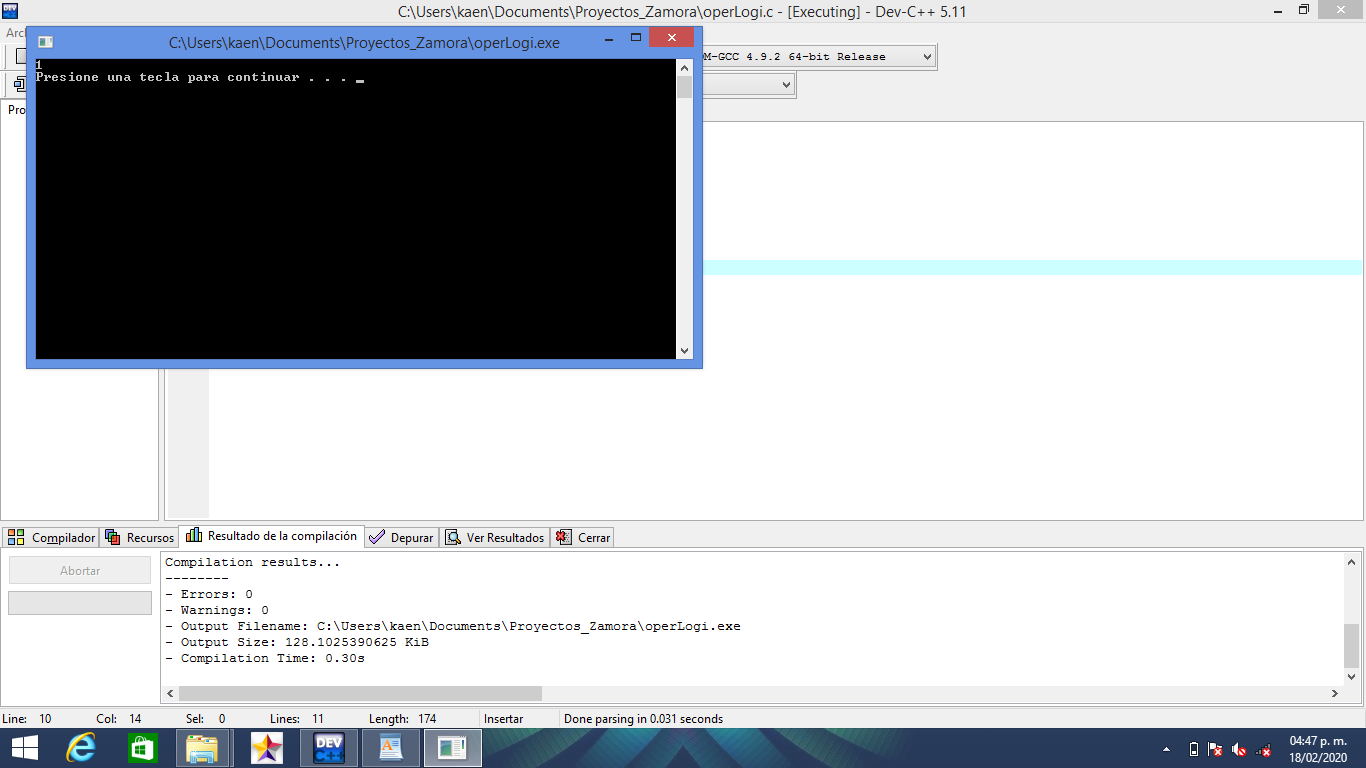


Ejercicio 2

En el ejercicio de abajo se importan las librerías y el método principal dentro de él se colocan 4 variables de tipo entero con un valor ya asignado que se guardara en memoria, vamos a comparar si P y R son iguales con ayuda del operador lógico &.



Tiene que mandar un 1 por que r y p son iguales y si se cumple la condición.



Ejercicio 3

Los operadores de comparación nos ayudan a ver si las operaciones son iguales, son mayor que, menor que, diferente de y eso lo utilizamos cuando vamos a tomar una decisión.

En la parte de abajo se muestran cuatro operaciones que son:

3>5 no se cumple porque es menor 3

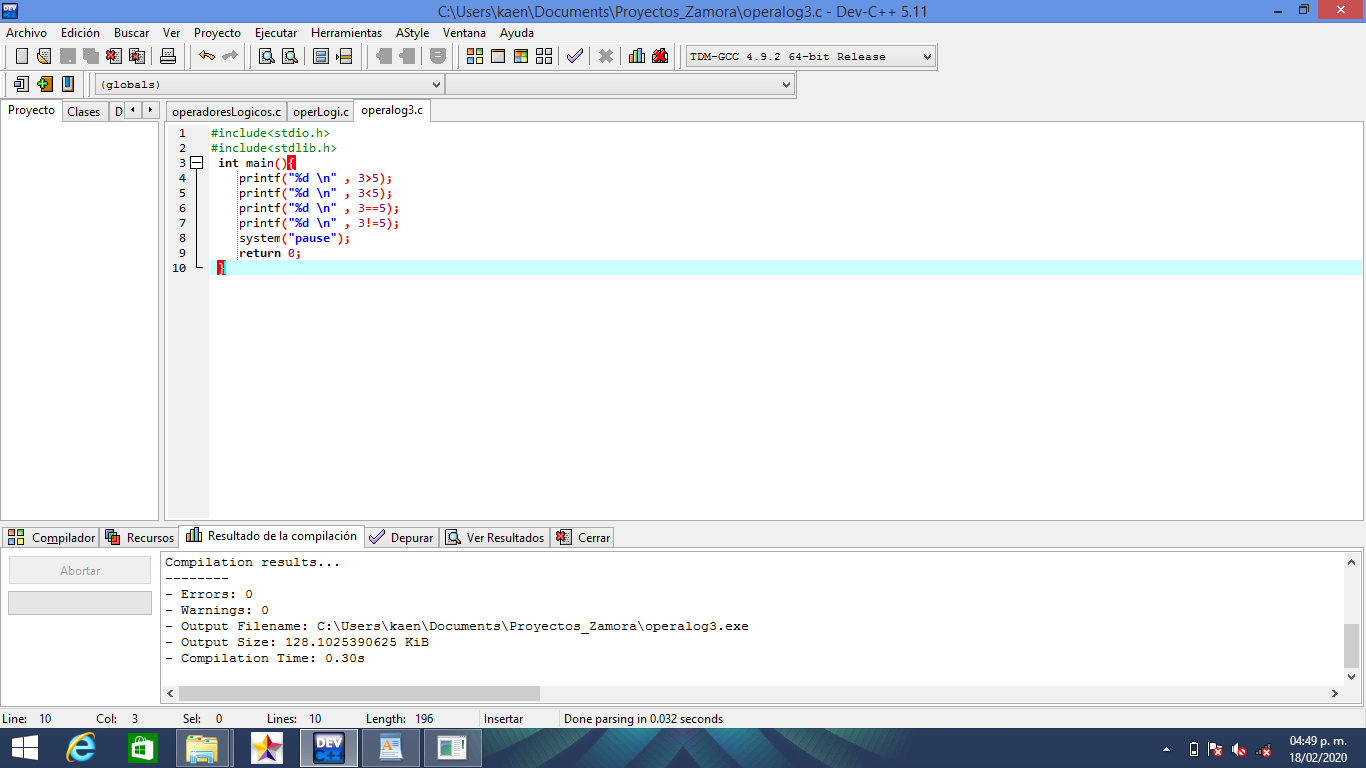
3<5 si se cumple porque 3 es menor

3==5 no se cumple por que no son iguales los números

3!=5 es diferente de 3 y no es igual que 5

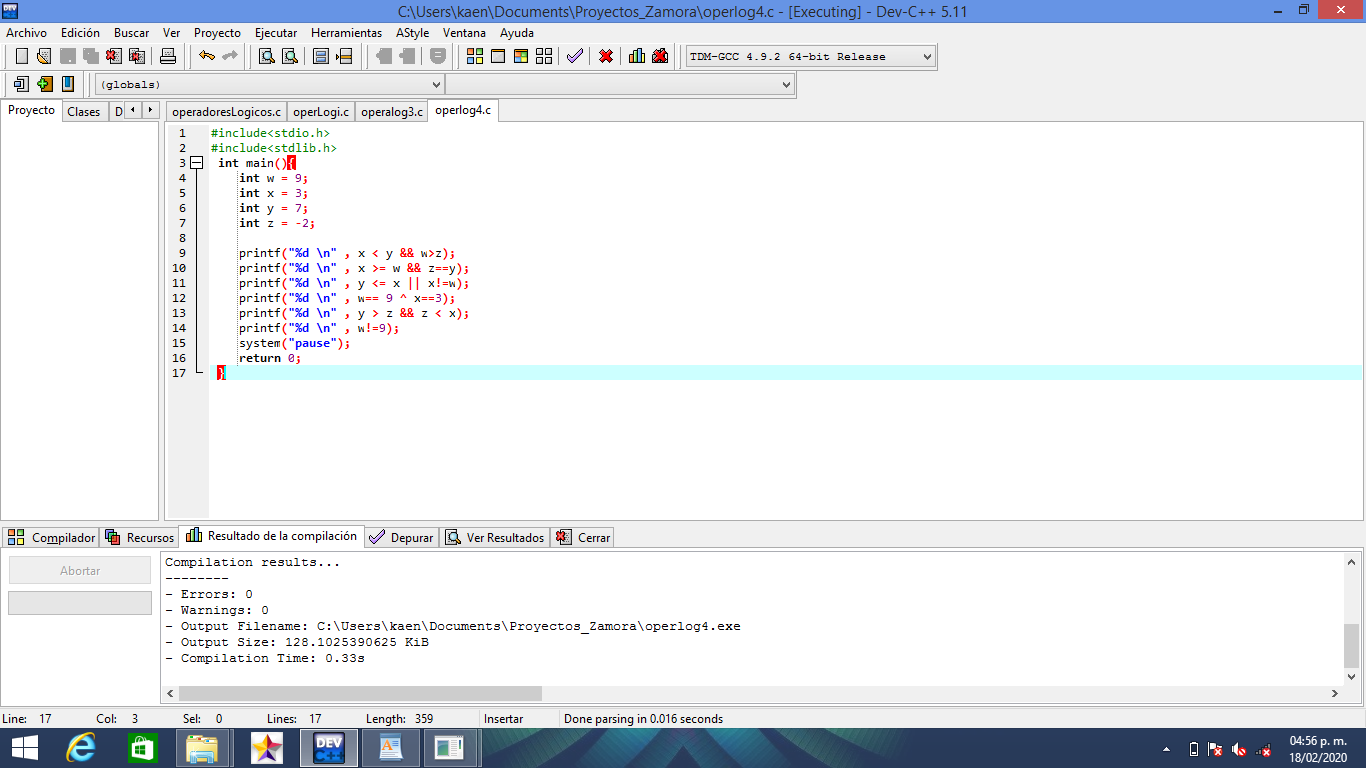
La única que es verdades es 3<5

Y en la pantalla se muestra cómo es que se programa

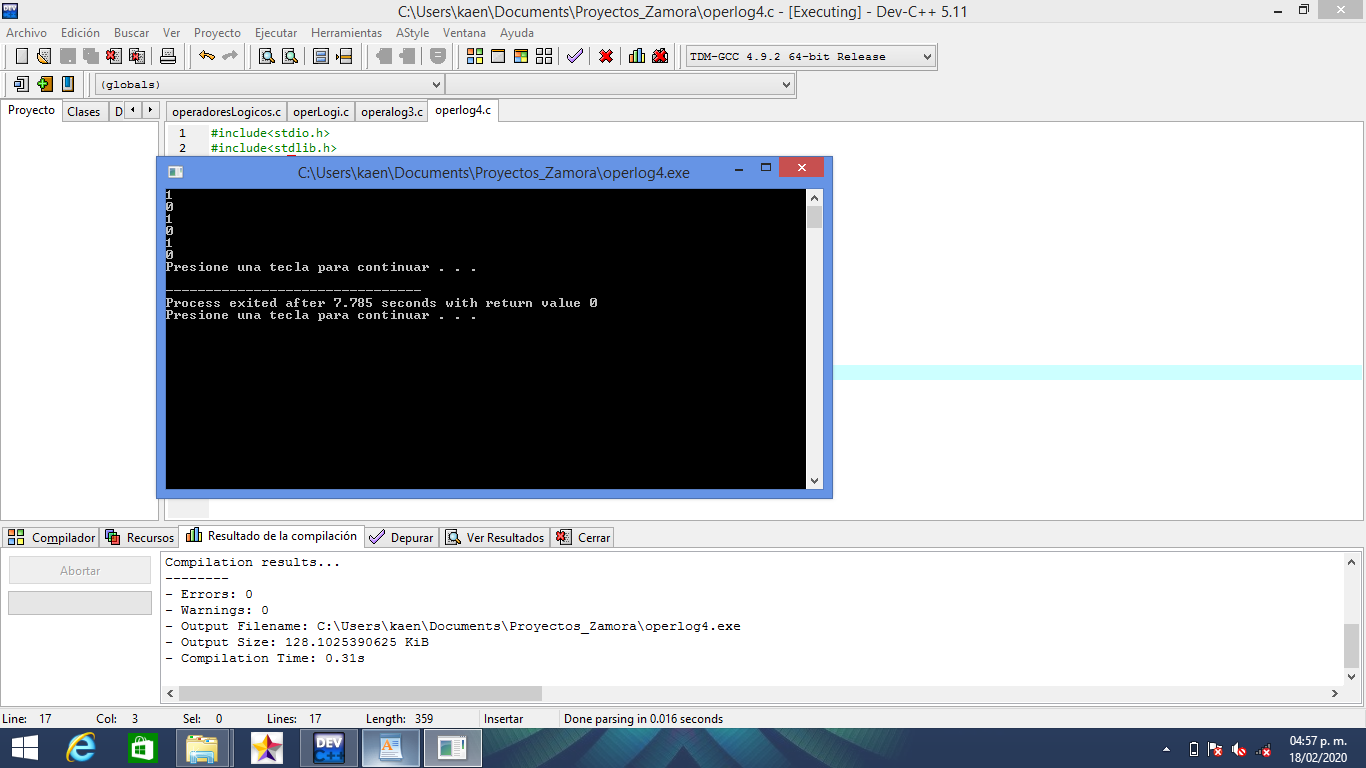


Ejercicio 4

Para poder entender esto hice otro ejemplo que es definir constantes y darles un valor y en la parte de abajo comparar si las condiciones e cumplen o no para esto hice uso del printf para poder imprimir un mensaje y a la vez imprimir si es verdadero o falso.



En la parte de abajo se muestra el resultado de las comparaciones que se realizaron el 1 es verdadero y el 0 es falso.

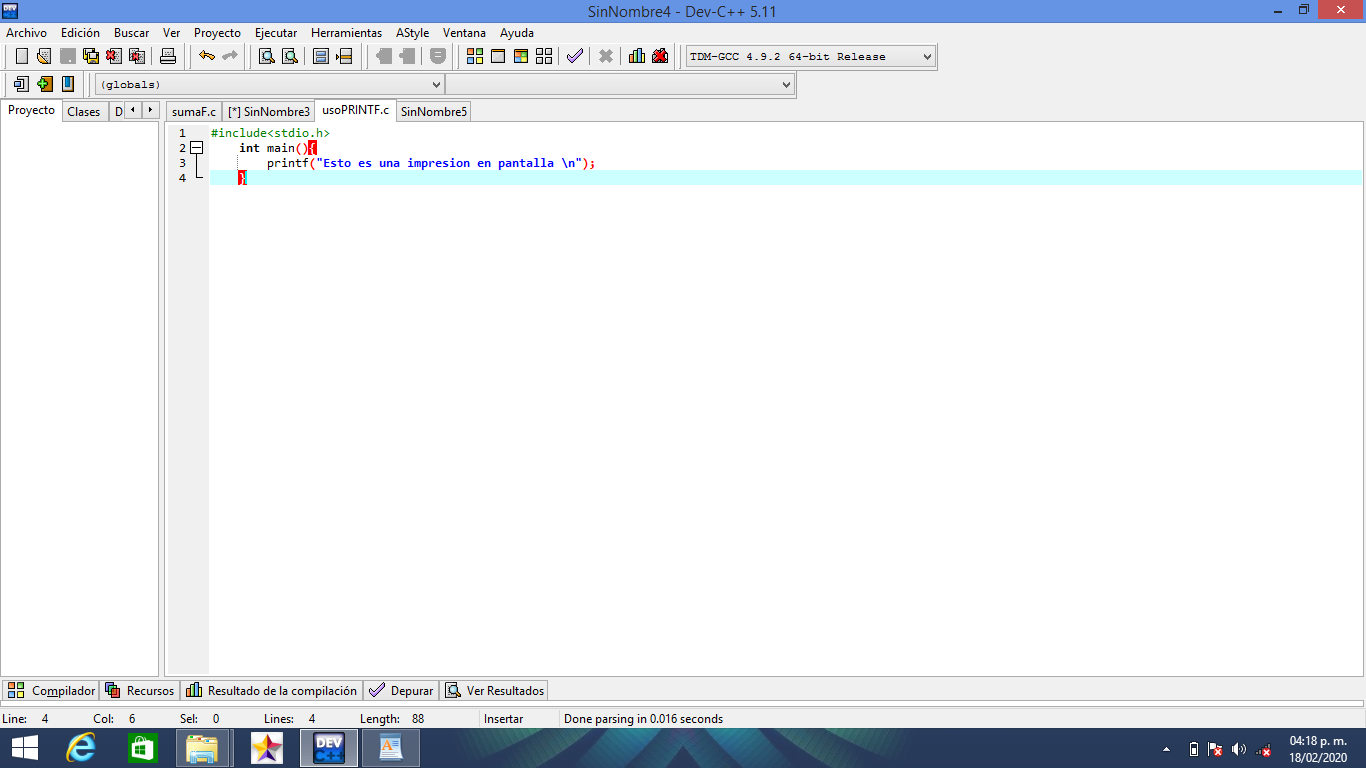


**7. Entrada y Salida de datos**

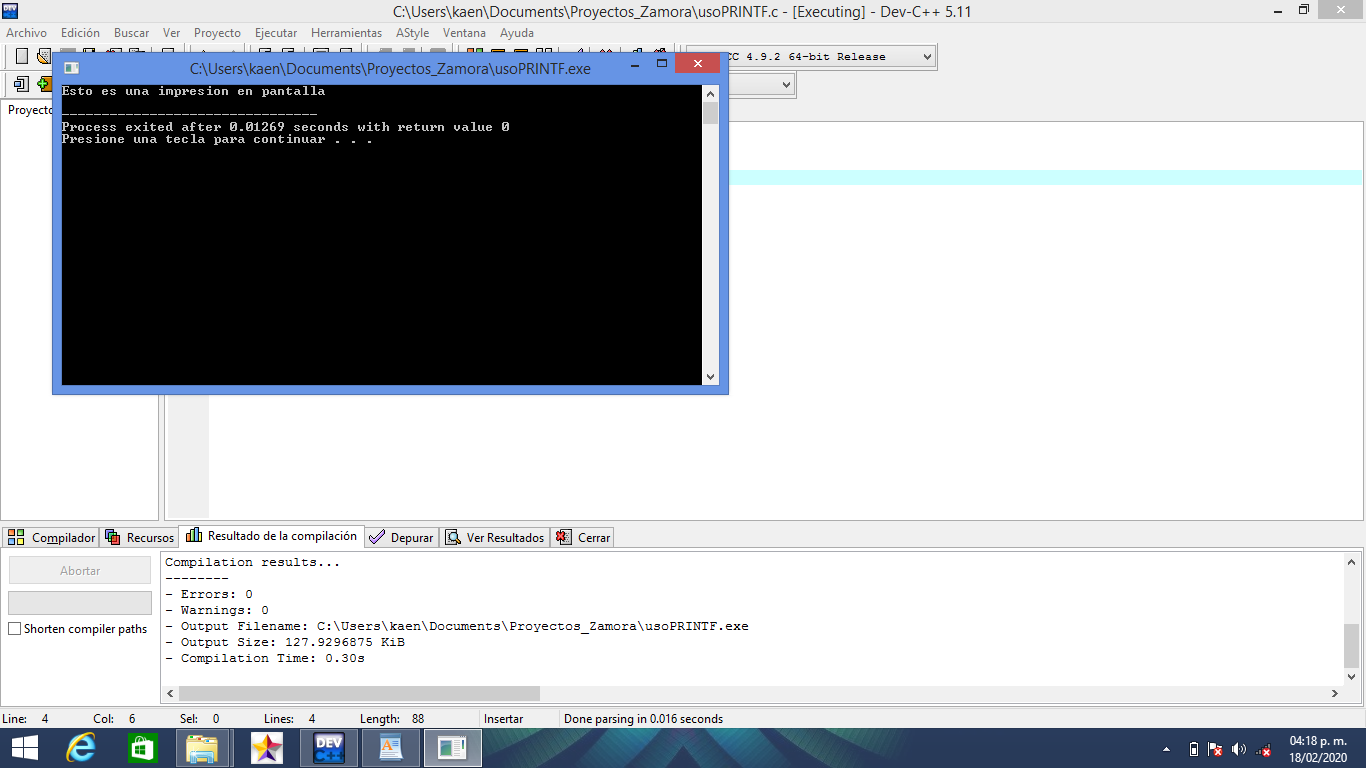
Ejercicio 1

En este ejercicio se muestra como hice el uso de printf que me permite imprimir mensajes en pantalla, quise mostrar un mensaje para que se vea su uso.Para esto importe las librerias que me permiten hacer uso de estos componentes, asi como tambien colocar el metodo principal que todo programa debe de llebar para poder ejecutarse debe de llevar llaves y dentro de lla meti en mensaje.

Como se muestra en la pantalla de abajo:

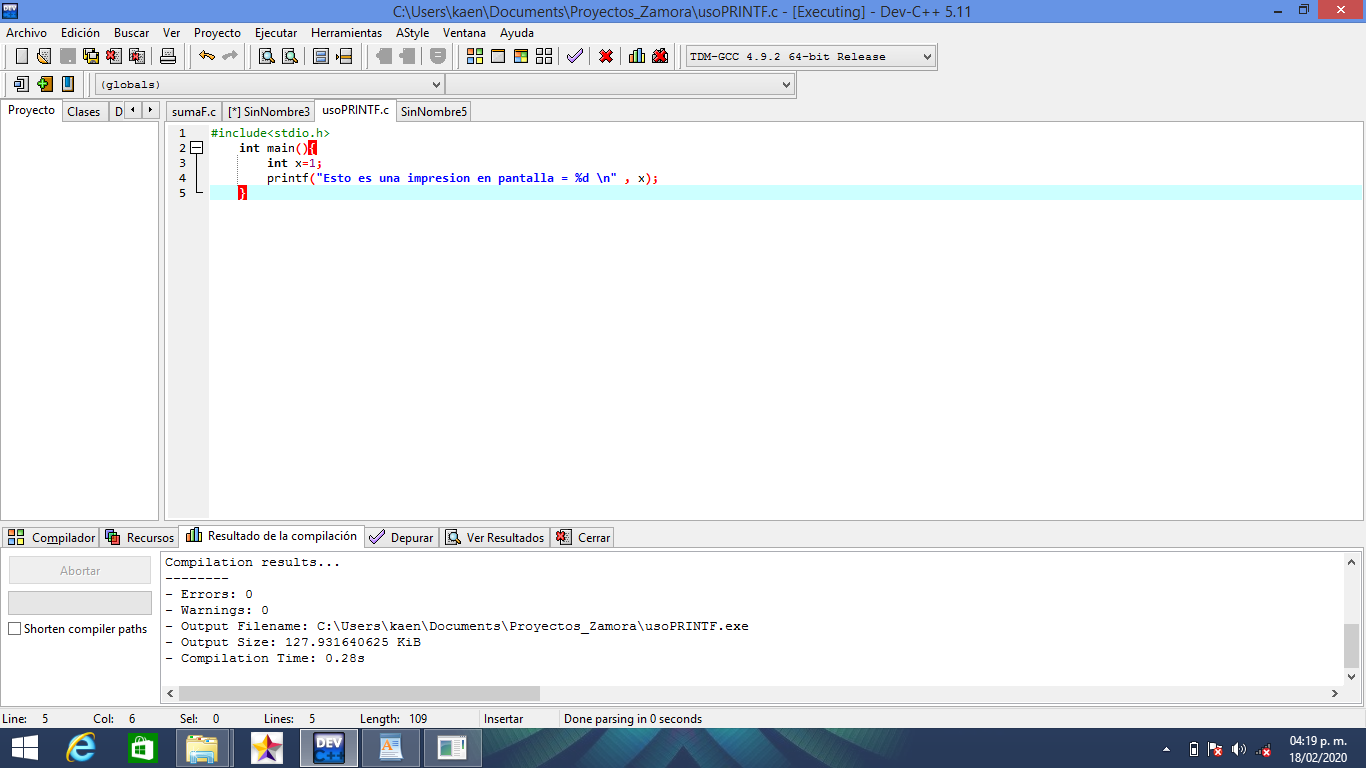


En la pantalla de abajo se muestra cuando lo corro y como es que aparece en la consola.



Ejercicio 2

En este ejercicio pude checar como es que también con ayuda de printf podemos imprimir valores de variables, en la pantalla de abajo se muestra un código que realicé para poder mostrar cómo es que se muestra en la pantalla.

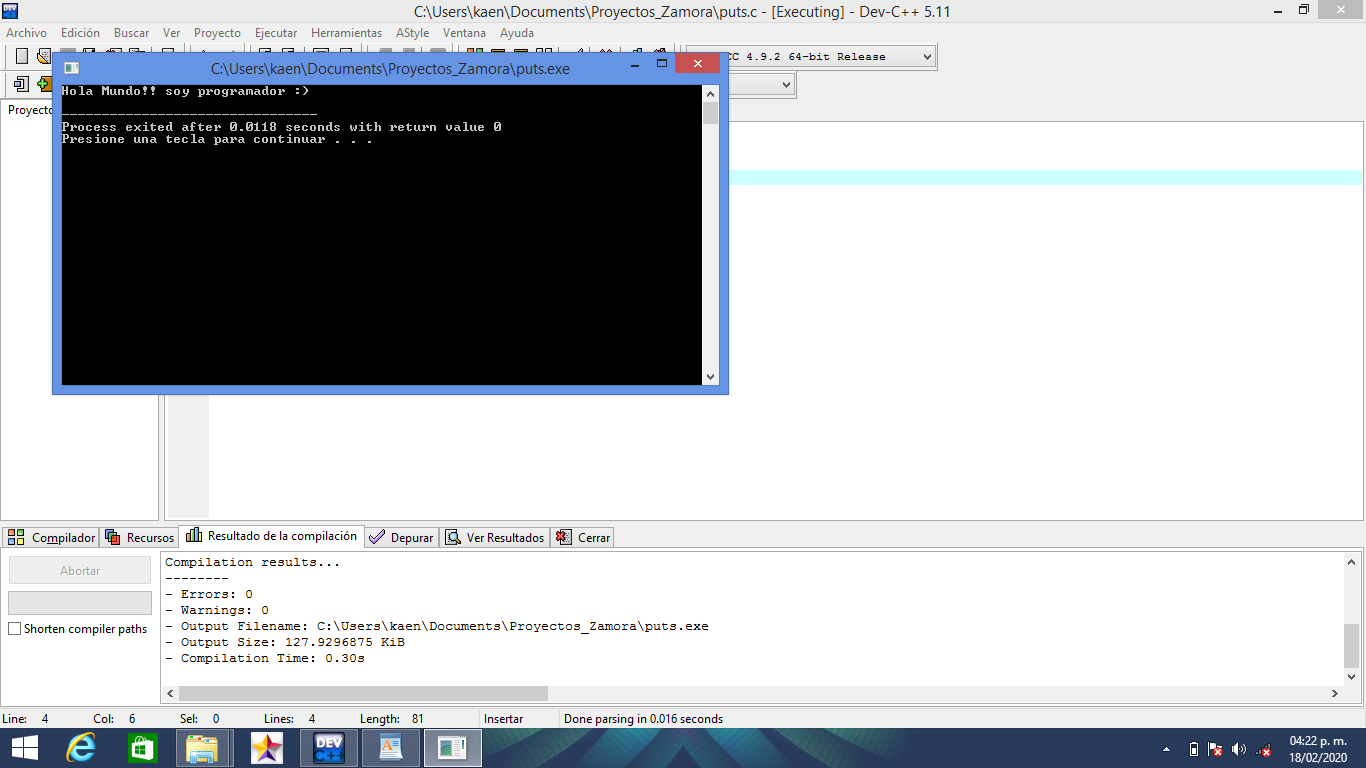


Ejercicio 3

En este ejercicio se muestra el uso de puts que nos sirve para poder imprimir mensajes sin variables, solo hice uso de la librería #include<stdio.h> y con ayuda del método principal para poder ejecutar lo que tenga dentro que en este caso es solo un mensaje.



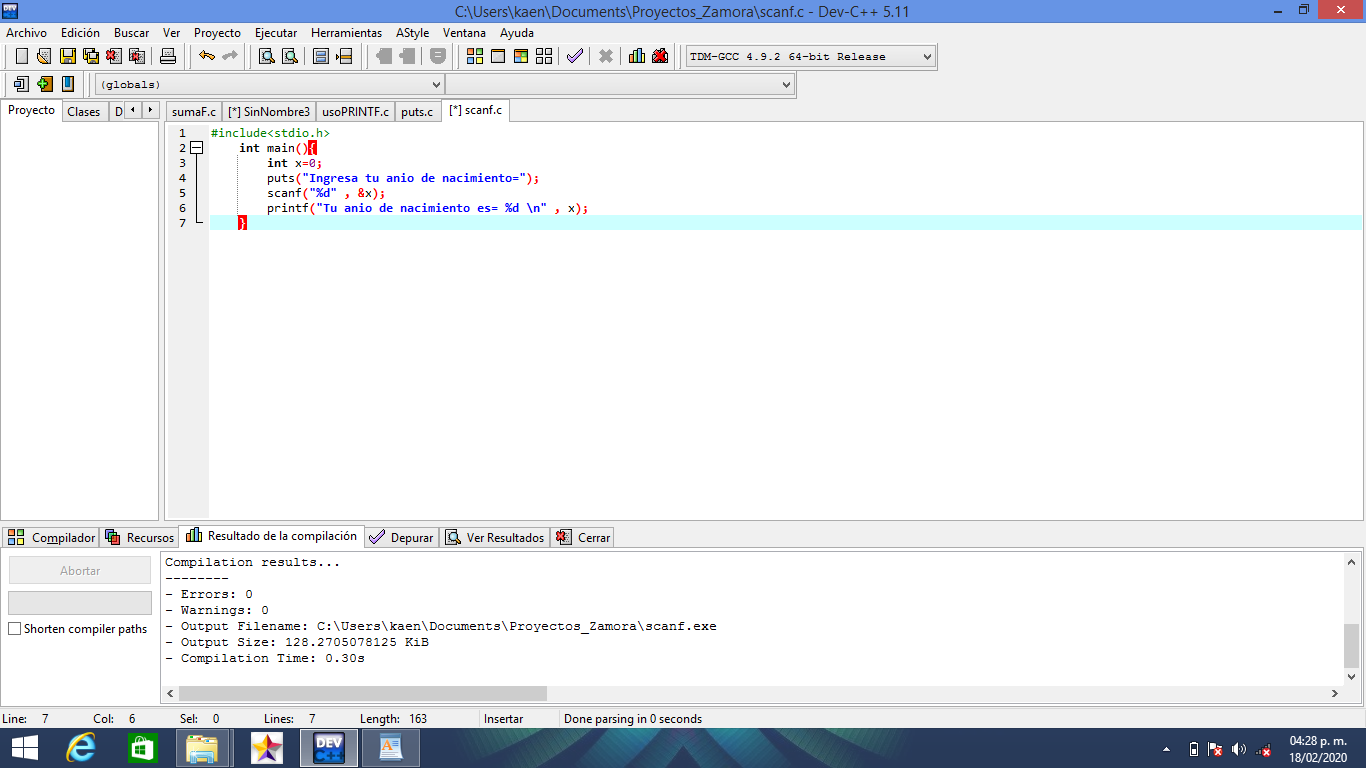
Cuando lo compilo se muestra el mensaje que coloque dentro del método principal y asi quedo como se muestra en la pantalla de abajo.



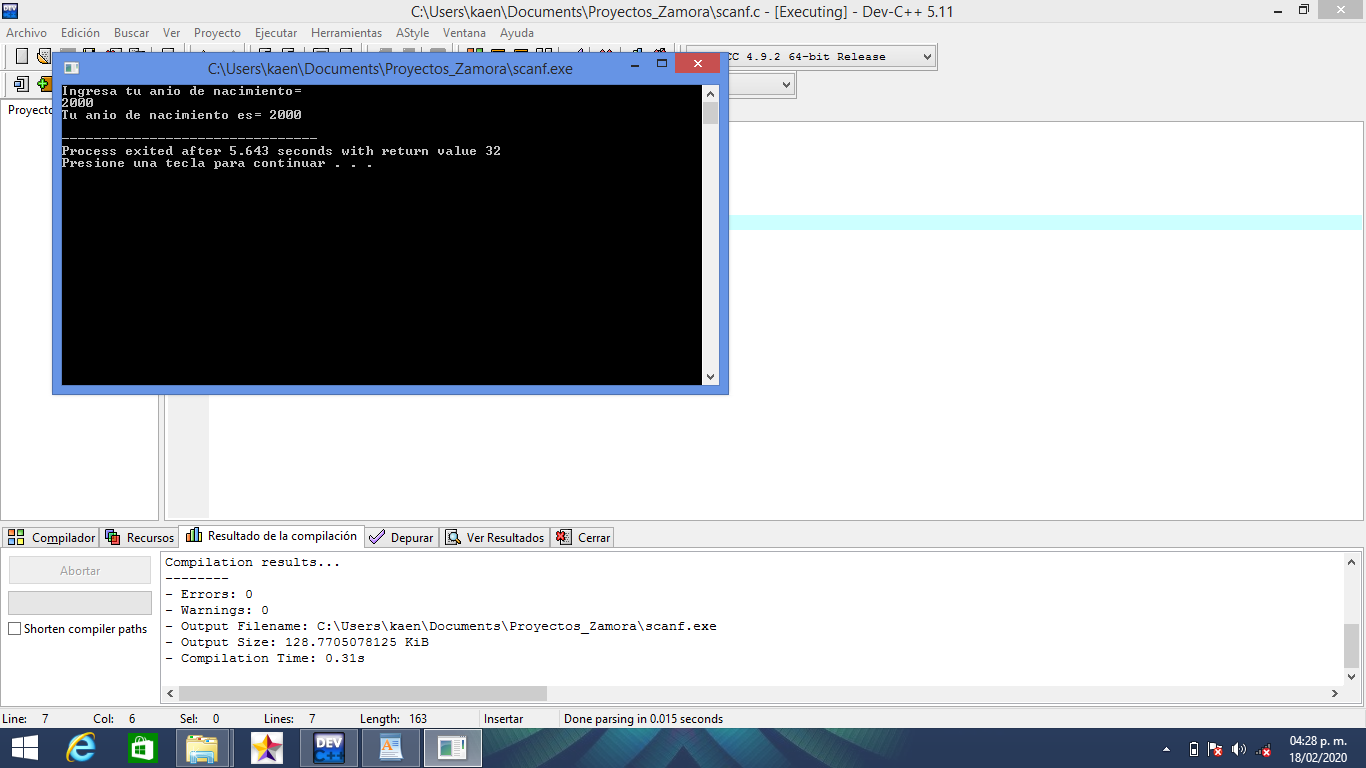
Ejercicio 4

En el ejercicio de abajo se muestra el uso de las funciones printf que es para mandar un mensaje junto con una variable, puts para mandar solo mensajes y la nueva funcionalidad es scanf que nos permite que se lean datos del teclado de acuerdo a un formato especificado, y asigna los datos de entrada a una o más variables del programa.

El ejemplo es que ingrese



En la pantalla de abajo se muestra cómo es que se ejecuta el programa y como es que te pide tu año de nacimiento y como te lo imprime en pantalla.



* **IV. CONCLUSIONES:** En mi punto de vista se me hiso interesante trabajar con este lenguaje de programación ya que nunca lo había usado es entendible y fácil de aprender ya que pude realizar cosas interesantes en el de una manera muy fácil, se maneja casi igual de java, pero su estructura en algunas cosas si cambian así que hay que irnos despacio para entenderle. Mi experiencia con este lenguaje fue buena porque yo antes había encontrado programas que estaban escritos en lenguaje C y ni los vea al 100% por que no les entendía y no podía saber que era lo que mostraban, pero con lo básico que aprendí sé que es bueno saber programar aquí.