|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Laminas 12-13** | | | **No.** | **2** |
| **Asignatura:** | **Métodos numéricos** | **Carrera:** | **Ing. Sistemas Computacionales** | **Duración de la práctica (Hrs)** |  |

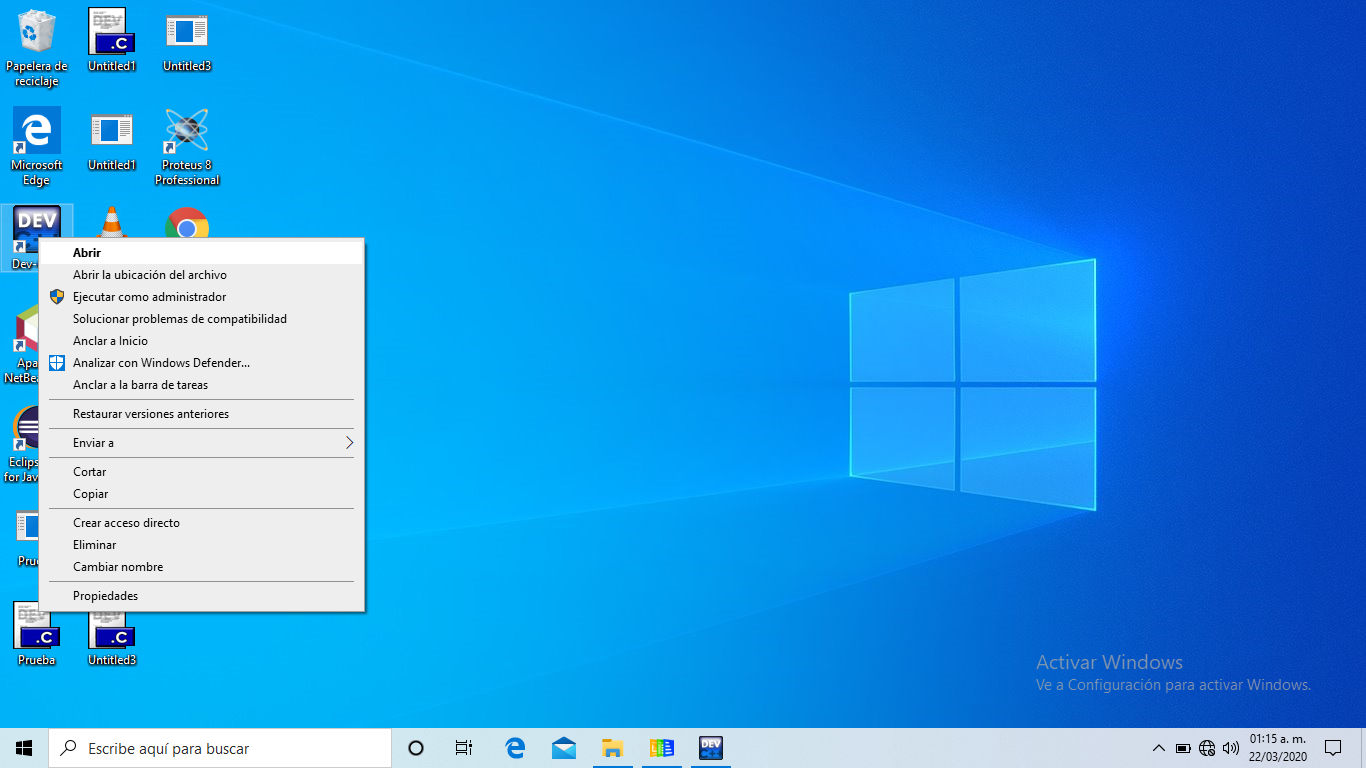
**II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):** Otro

**III. Material empleado:** laptop y Dev-c

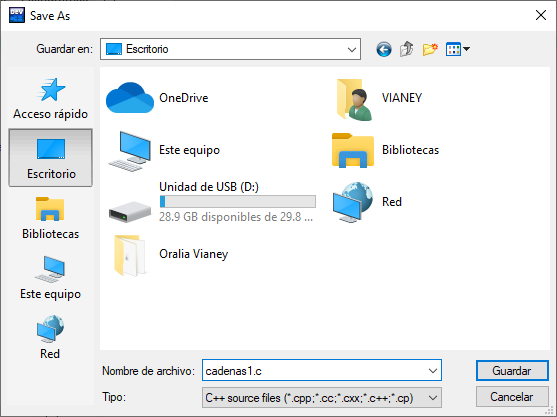
**IV. Desarrollo de la práctica:**

# Lamina 12 Cadenas

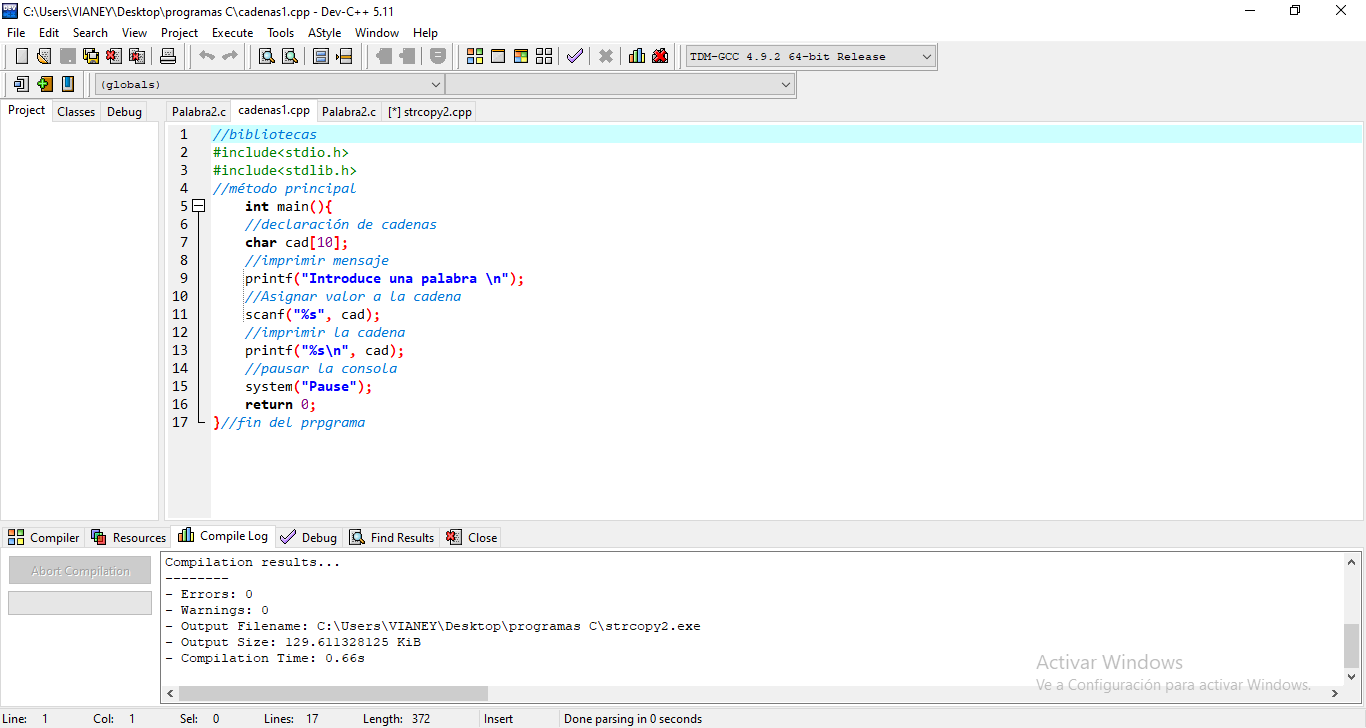
Primero abrir mi software por medio del icono del icono de acceso directo como se muestra en captura dando clic derecho.



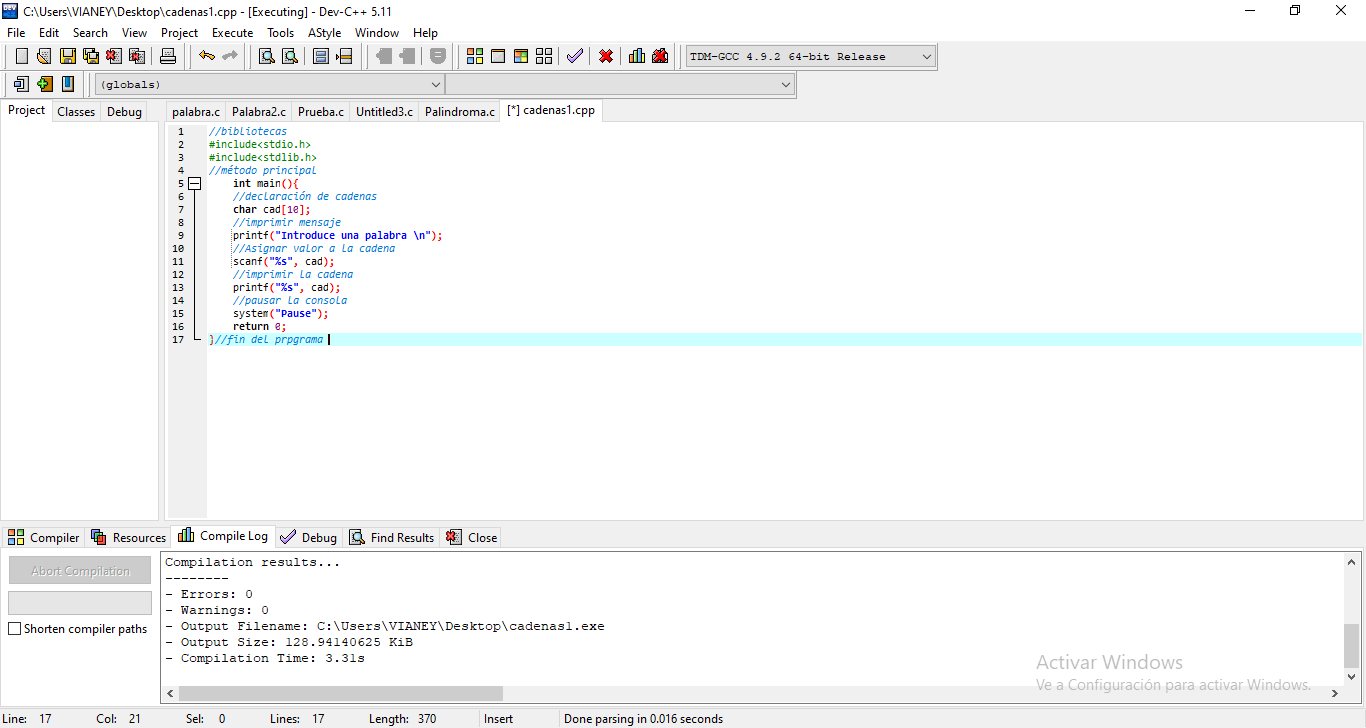
Posteriormente abrimos un nuevo programa al cual nombrare cadenas1 como se muestra en la siguiente captura.



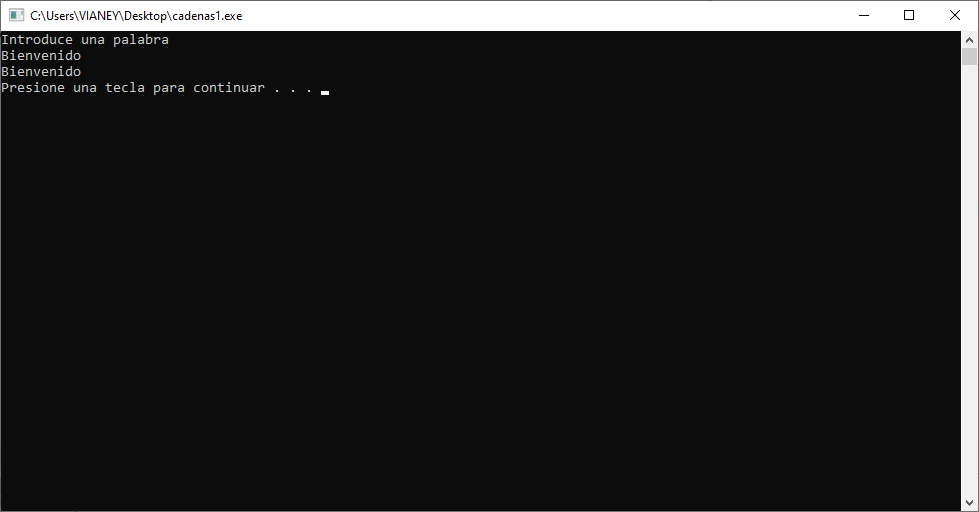
Después de ello declare una cadena con un tamaño de 10, es similar a un arreglo como lo podemos ver, a continuación, solicito al usuario por medio de un mensaje que ingrese una palabra la cual se almacenara en dicha cadena y finalmente la imprime en consola.



Di clic sobre el siguiente icono que se encuentra en la parte frontal de la pantalla en la barra de herramientas para compila y ejecutar el código.

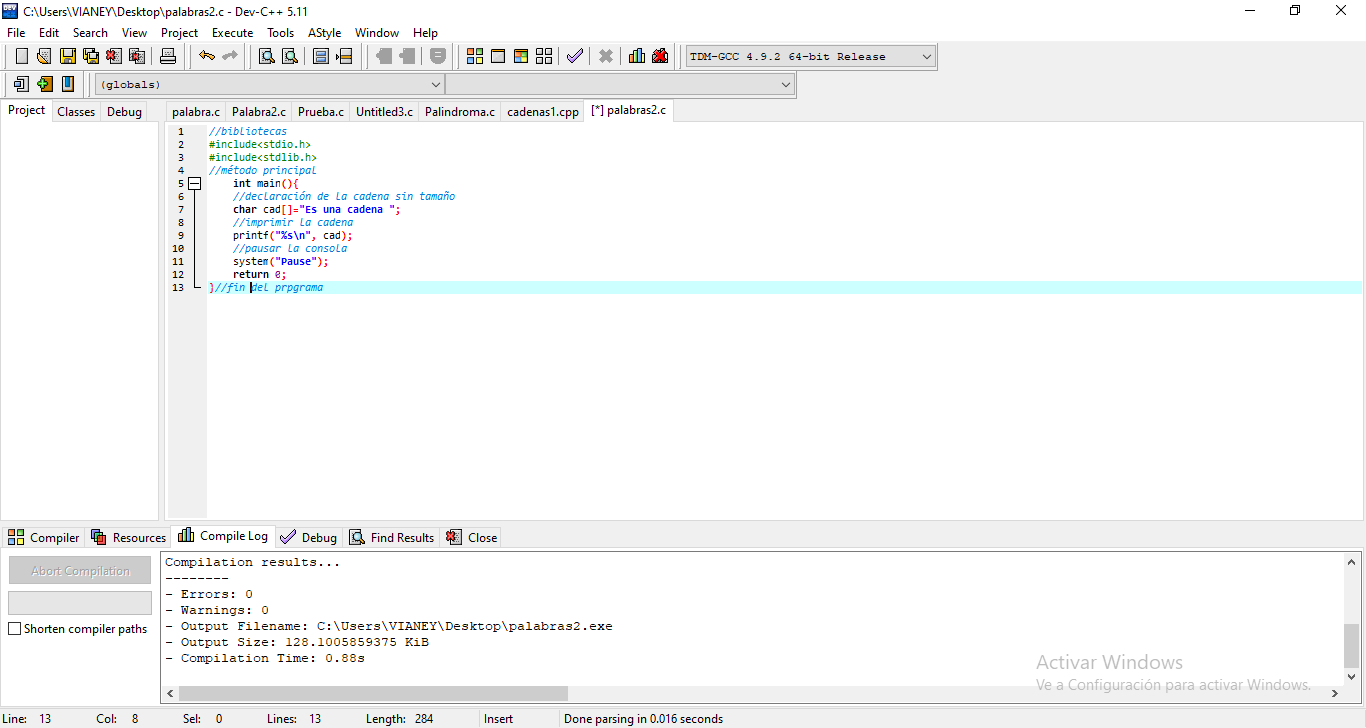


Esto es lo que se muestra en consola una vez que se ingresa la palabra y es correcto puesto que cuando se ingresa la palabra la almacena y vuelve a imprimirla.

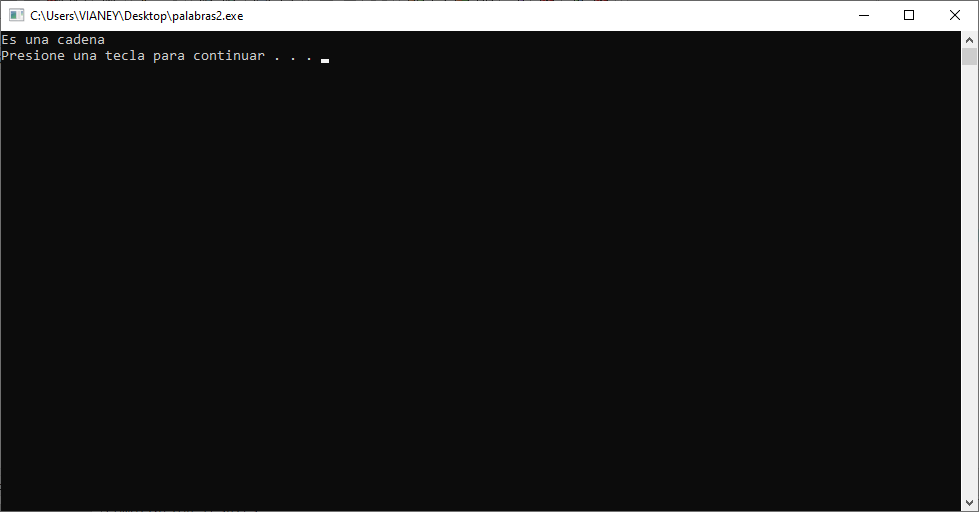


Ahora bien, como en la práctica anterior se pudo observar que nuestra cadena ya tenía un tamaño asignado, pero sin embargo las cadenas también pueden crearse cuando no se tiene un tamaño en específico y esto surge cuando se ingresara una palabra de la cual no sabemos cuántos caracteres posee, por lo tanto, para declararla se hace de la siguiente manera.

char cad[]=" ";

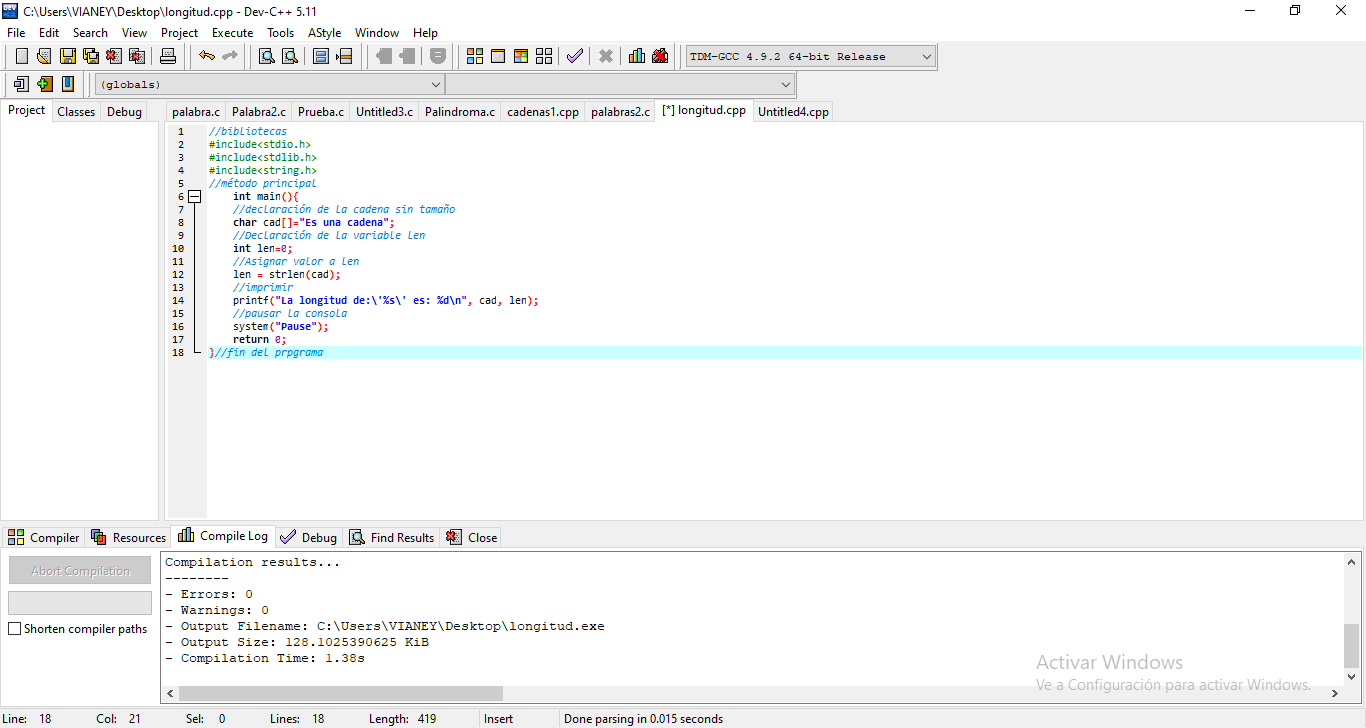


Al ejecutarlo se muestra el mensaje dentro de la cadena que anteriormente se declaró sin espacios y bien este es el resultado

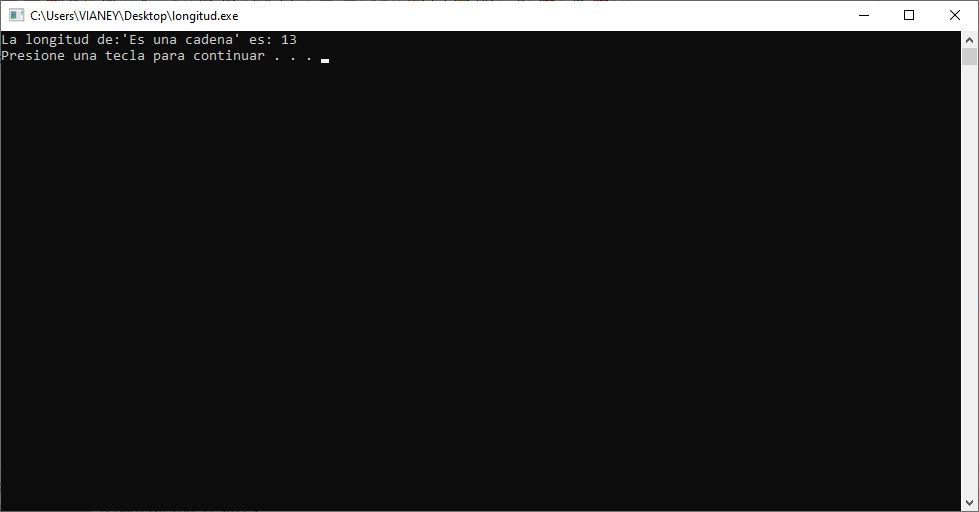


Sin embargo, existe otra solución a la problemática y esta consiste en obtener el tamaño de la palabra una vez que ya sea introducida por el usuario, por medio del método strlen

En el código una vez importadas y las librerías y dentro del método inicial, yo declare mi variable de tipo entero al igual que mi cadena posteriormente por medio del método (strlen )adquiero el tamaño de la cadena, y lo asigno como valor a la variable len menos uno y finalmente imprimo tanto la cadena como la variable len.



Finalmente, al ejecutarlo el resultado es el siguiente

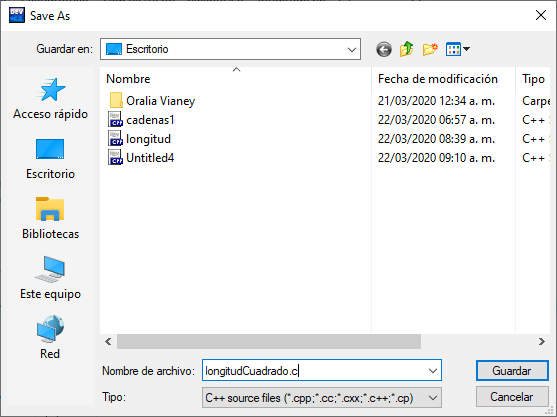
.

Ejercicio1

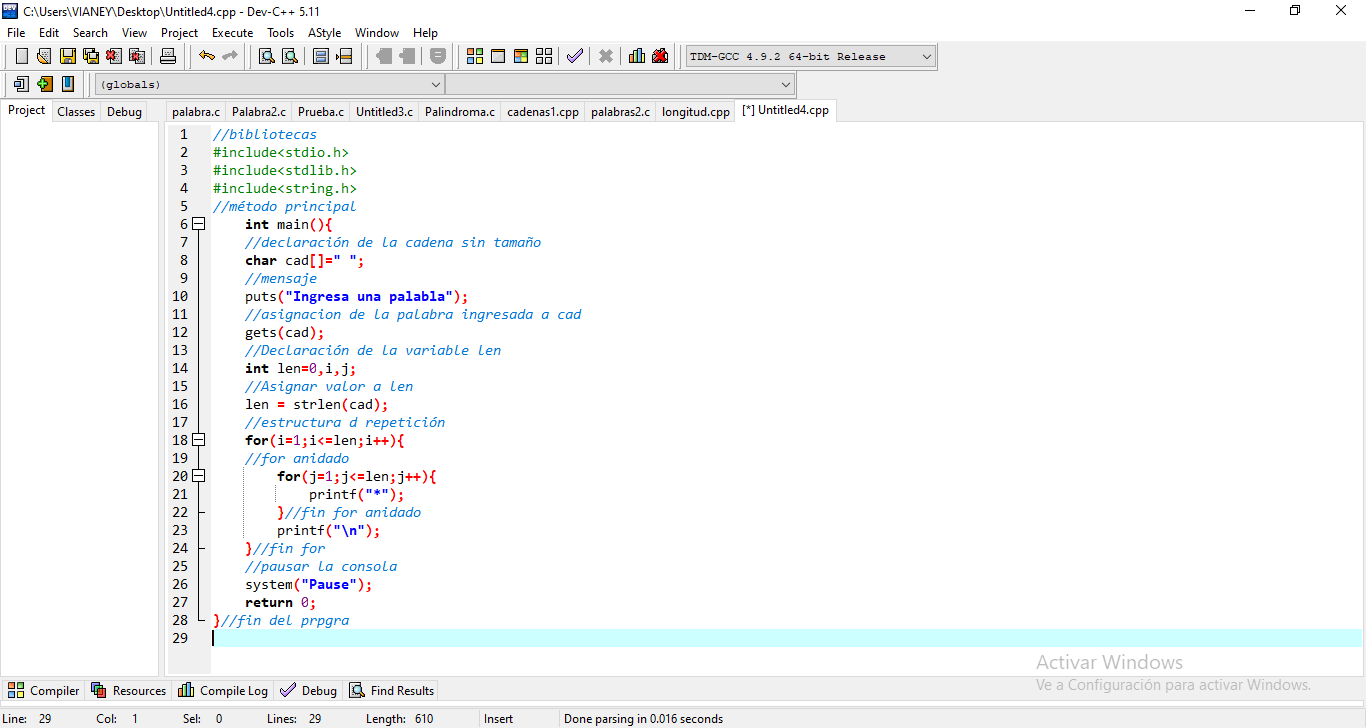
1. Escribe un programa que reciba una palabra por teclado.

De acuerdo a la longitud de la palabra (N) que se ingresó por teclado imprime un cuadrado de asteriscos de (N x N).

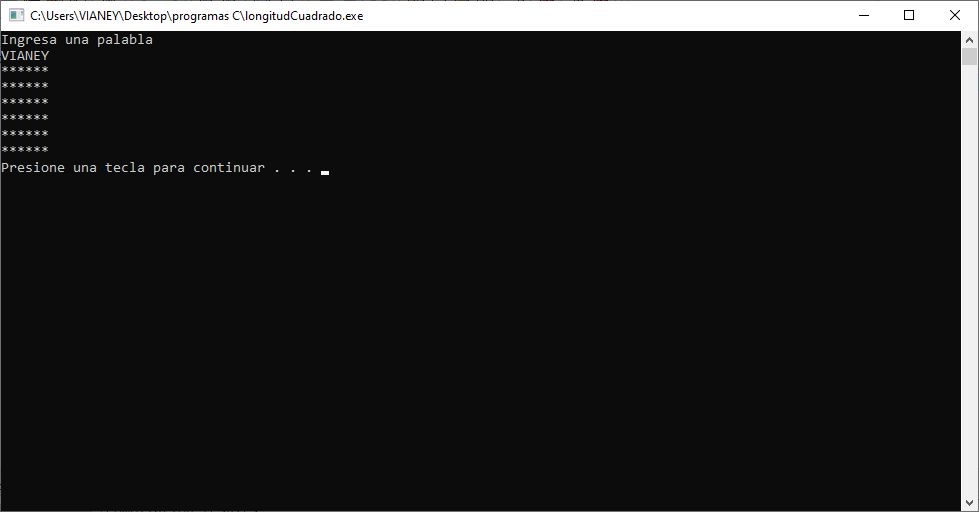
Para comenzar creo un nuevo proyecto, siguiendo los pasos anteriores, posteriormente lo guardo con el siguiente nombre,



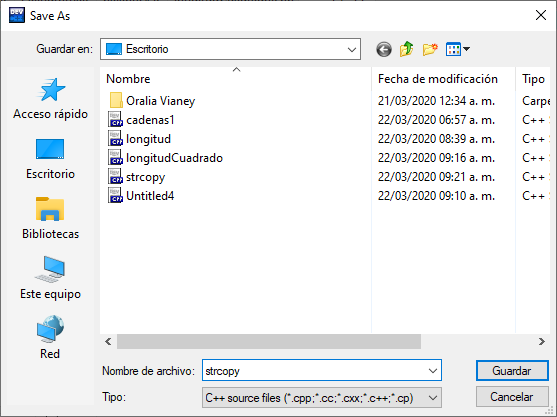
En cuanto al código desarrollado requerí de las instrucciones del programa anterior, primero solicito al usuario que ingrese una palabra, la cual asigno como valor a la cadena que cree, después de ello obtengo la longitud de dicha cadena y por medio de un ciclo for anidado obtengo la figura del cuadrado con asteriscos, lo primero que se realiza es que mientras que el for que se encuentra dentro da una línea de asteriscos vertical correspondiente al número de caracteres ingresados el de afuera da un salto de línea cada vez que se cumple un ciclo del de adentro, el código es el siguiente



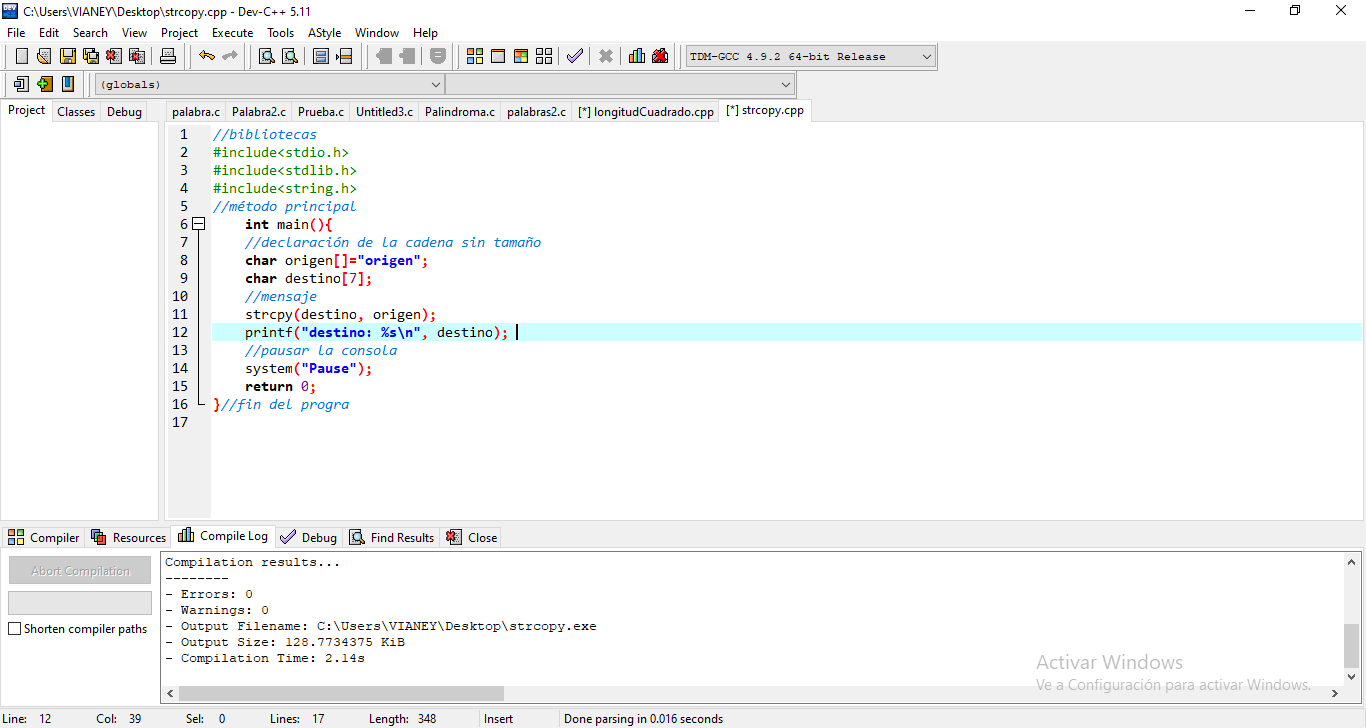
Al ejecutarlo se muestra de la siguiente manera en consola.



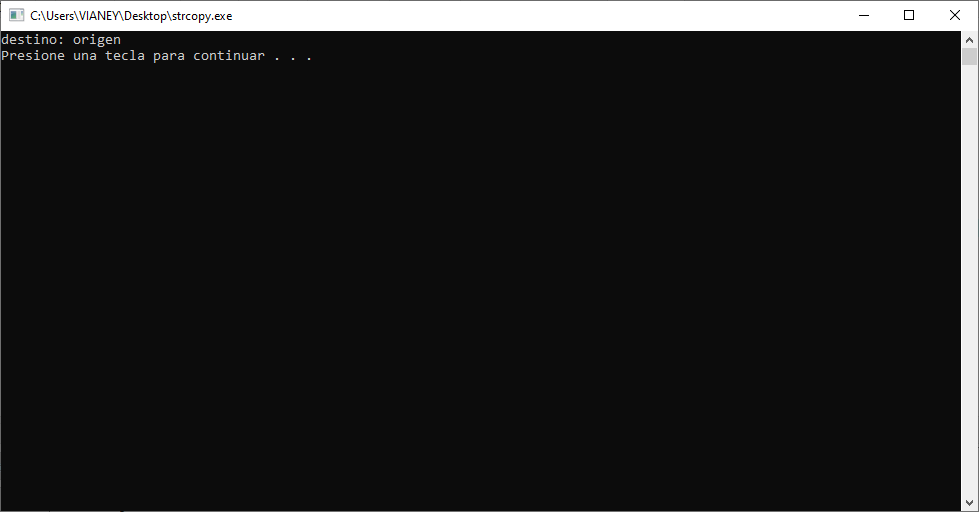
Seguido de ello se muestra el funcionamiento, así como sintaxis de strcpy el cual sirve para copiar una cadena de texto a otra y para ello cree un nuevo proyecto al cual asigne el siguiente nombre.



Mientras que, en el código, lo primero que se realiza es declarar una cadena llamada original la cual contiene el texto, posteriormente declare otra que llame destino la cual tiene un tamaño ya signado que es semejante al de la cadena origen, posteriormente por medio de (strcpy) copie la cadena origen en destino y finalmente solo imprimo la cadena destino que para entonces tiene el valor de la cadena origen, el código es el siguiente.



Mientras que al ejecutarlo se muestra de la siguiente manera.



Ejercicio 2

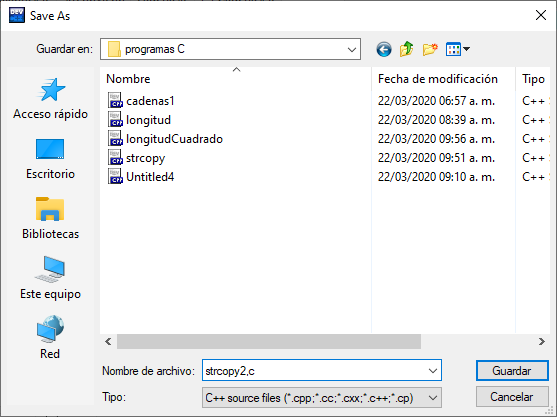
1. Escribe un programa que reciba por teclado dos palabras y cada una de ellas las almacene en un arreglo.

Después intercambia sus contenidos. Imprime el antes y el después

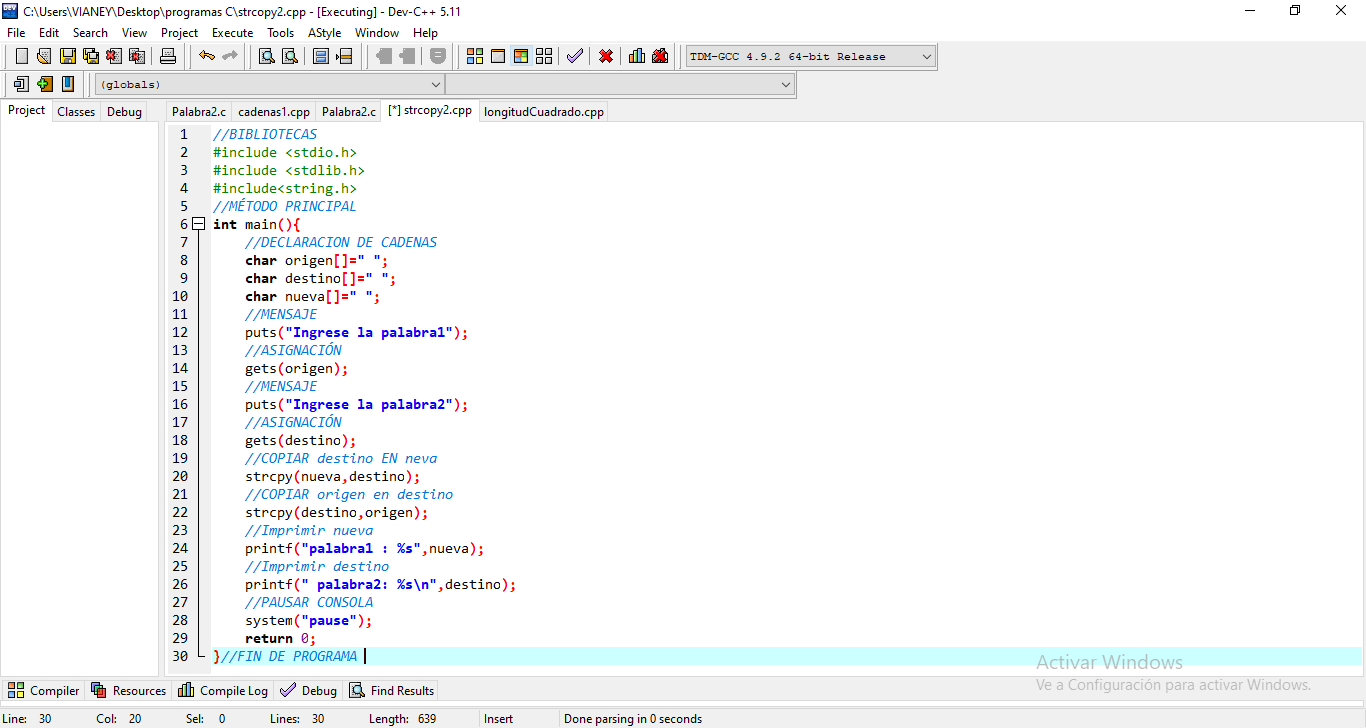
Ej.

antes después Palabra\_1 = ' Programación' Palabra\_1 = 'Computadora' Palabra \_2= 'Computadora' Palabra \_2= 'Programación'

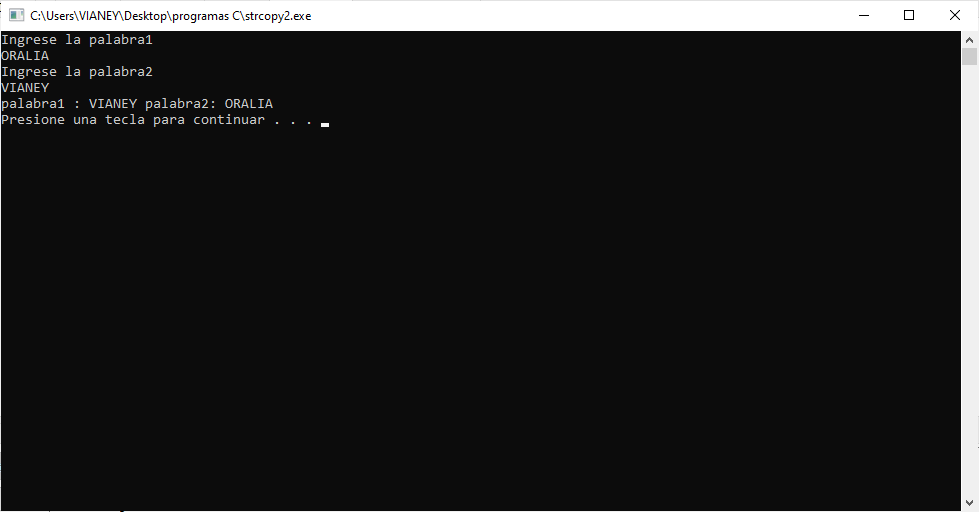
Creo un nuevo proyecto llamado



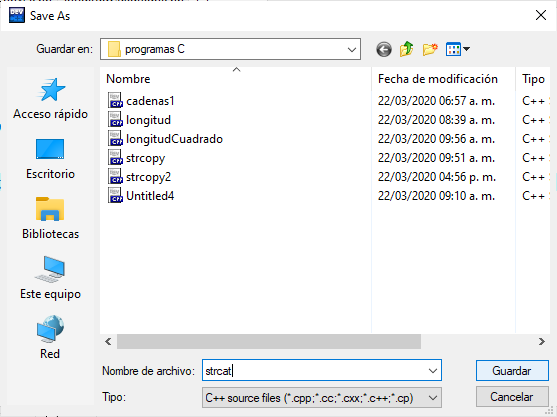
En mi programa mediante un mensaje solicita al usuario que ingrese dos palabras las cuales almacena en sus respectivas cadenas, posteriormente copio la segunda cadena en una nueva y la primera en la segunda cadena y así finalmente imprimo en pantalla con un mensaje indicando cual es cada una de las cadenas, mostrando como se copió una en otra.



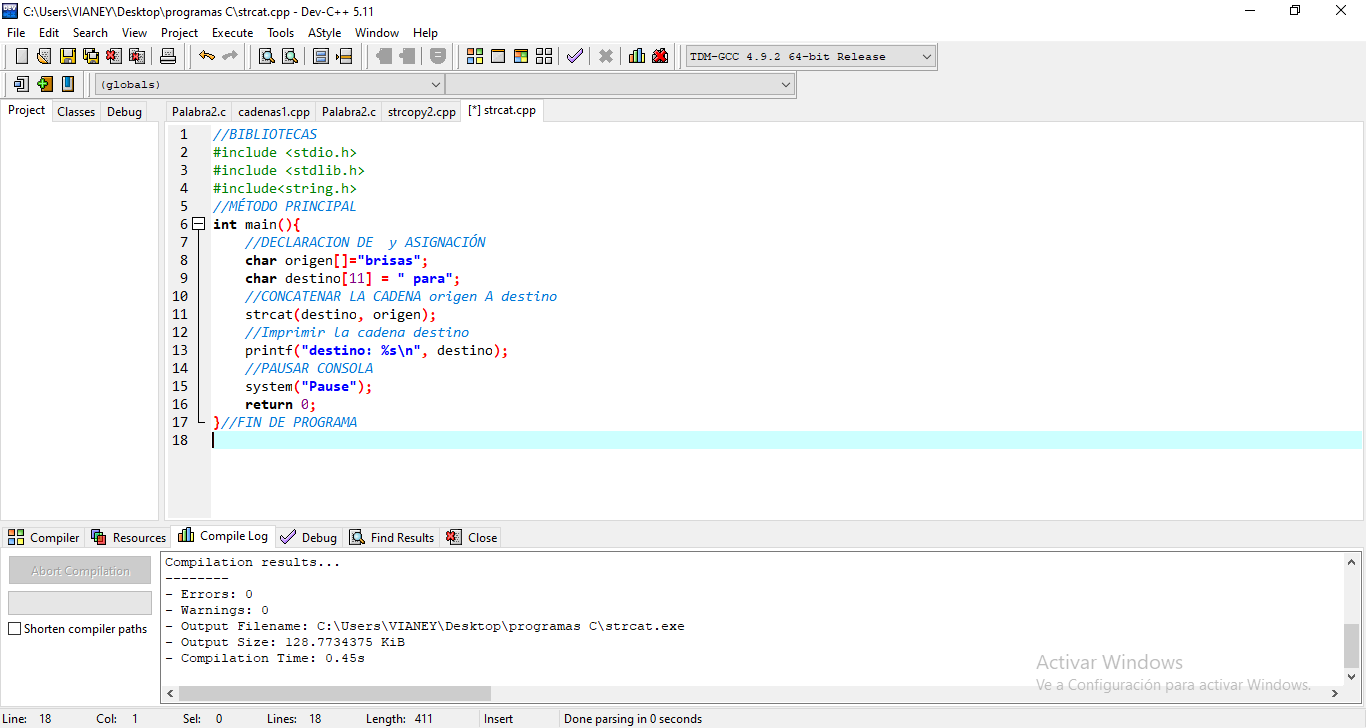
Al ejecutarlo e ingresar mis dos palabras se muestra de la siguiente forma.



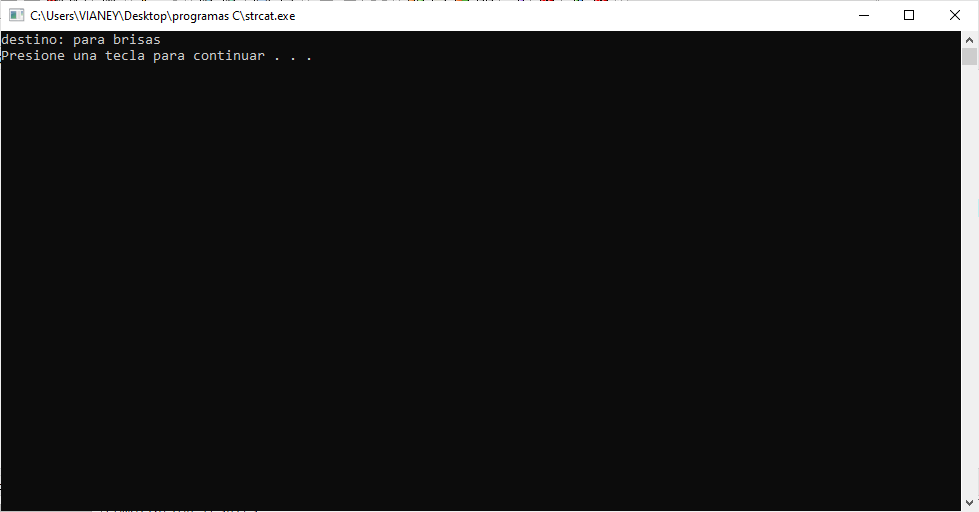
Cree un nuevo proyecto con la finalidad de mostrar la sintaxis y aprender el funcionamiento de strcat la cual permite concatenara cadenas y a mi proyecto lo nombre de la siguiente manera para poder ejecutarlo, para guardarlo en la pestaña frontal doy clic derecho sobre F**ile** y seleccione **Save As**, coloco el nombre, sucesivamente presiono el botón **Guardar**



En este código describí que se realiza en cada una de las líneas para lograr hacer más entendible el funcionamiento de la sintaxis



En el proceso de ejecución y compilación este es el resultado.



Ejercicio 3

1. Escribe un programa que reciba por teclado dos palabras.

Y concatene N veces la segunda palabra a la primera palabra.

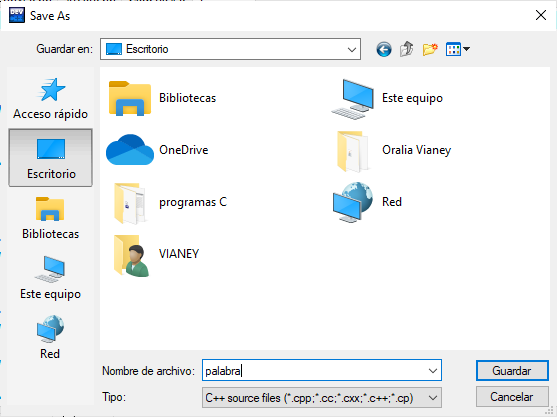
Donde N es la longitud de la primera palabra.

Ej.

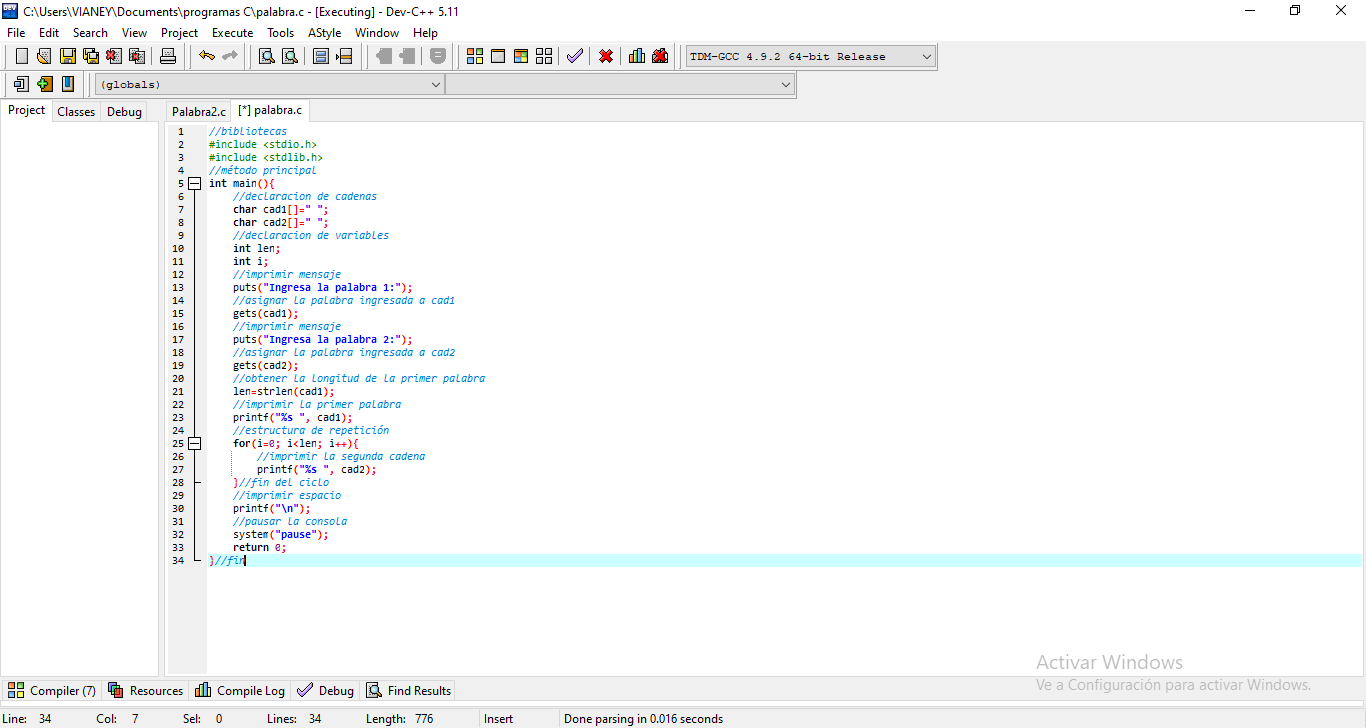
Palabra1: 'para' Palabra1: parabrisasbrisasbrisasbrisas

Palabra2: 'brisas‘

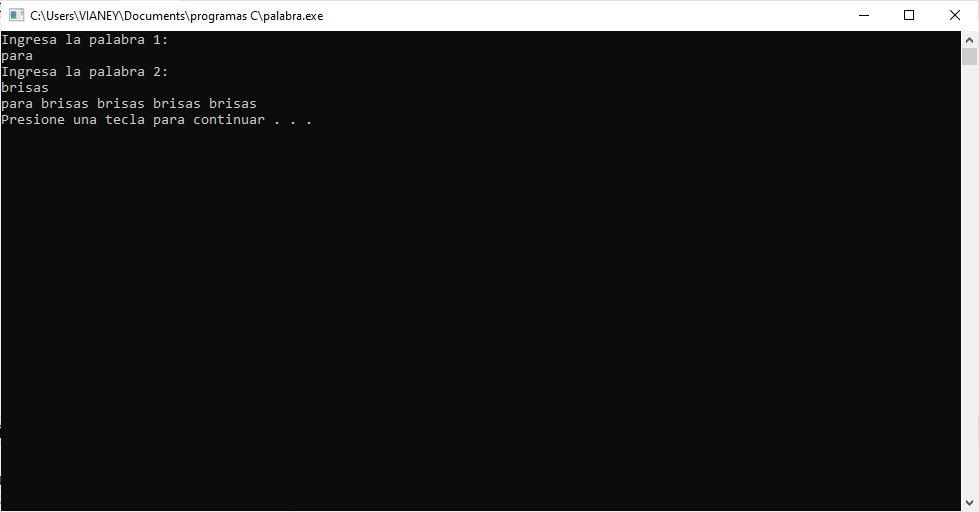
Abrimos un nuevo proyecto al cual llamamos de la siguiente manera, con la extensión (.c )



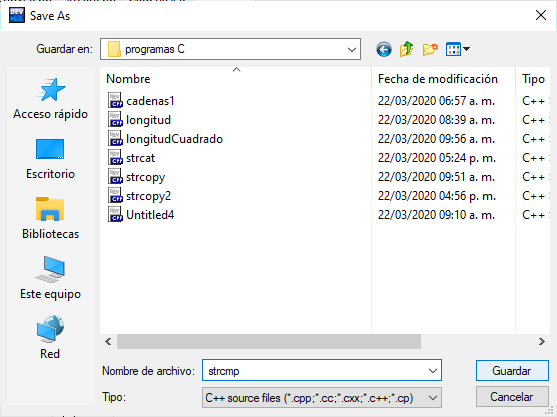
desarrolle el siguiente código en el cual solicitamos al usuario que ingrese primero que ingrese dos palabras, las cuales son asignadas a las cadenas respectivamente, después obtenemos el numero de caracteres que posee la primer palabra, después repetimos la impresión de la segunda palabra, y para cuando el número de veces que se imprime sea menor a la longitud de la primera palabra y al finalizar el ciclo imprimimos el salto de línea



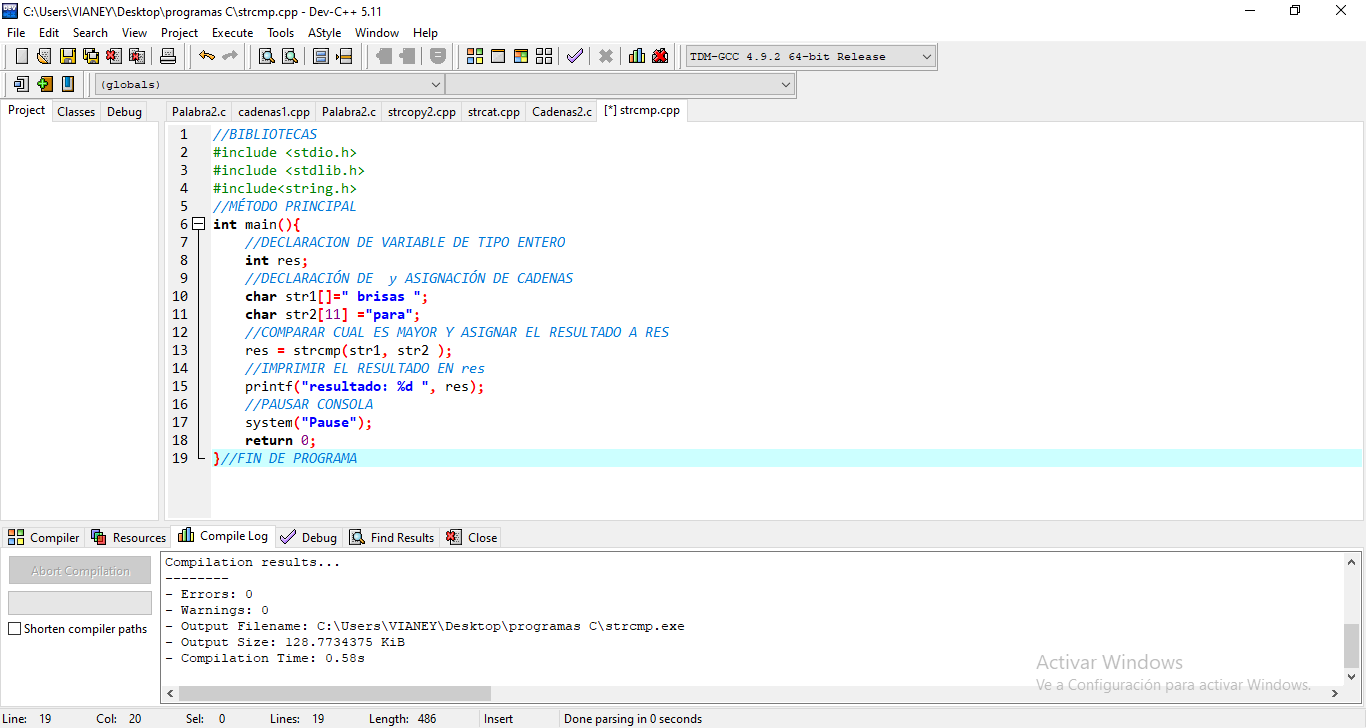
Finalmente, al ejecutar el programa el resultado a mostrar en pantalla es el siguiente



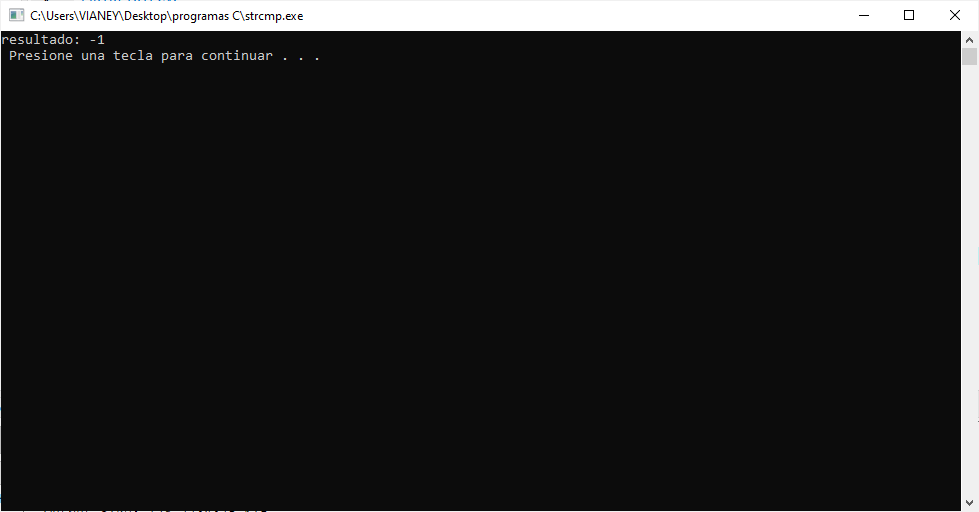
Esta nueva sintaxis nos sirve para comparar cadenas, para mostrar su funcionamiento guardo mi nuevo proyecto donde desarrollare mi código como en los ejemplos anteriores



El código compara dos cadenas estáticas, es decir ya declaradas y el resultado a obtener es un -1 en caso de que la <cadena1> es mayor que <cadena2>, un número positivo distinto a 0 si <cadena1> es menor que <cadena2> y un cero si son iguales y dicho valor se asigna a la variable de tipo entero llanada res y finalmente se imprime dicho valor.



Por lo tanto el resultado a esperar es menos uno.

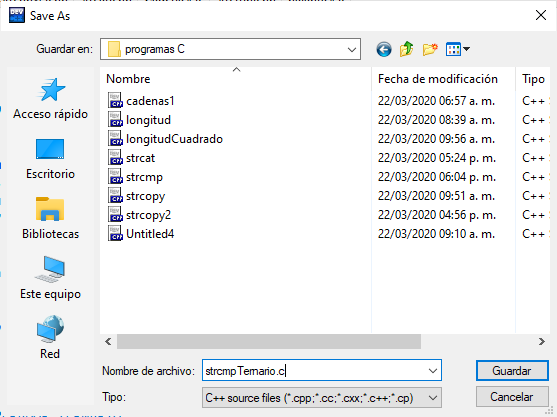


Ejercicio 4

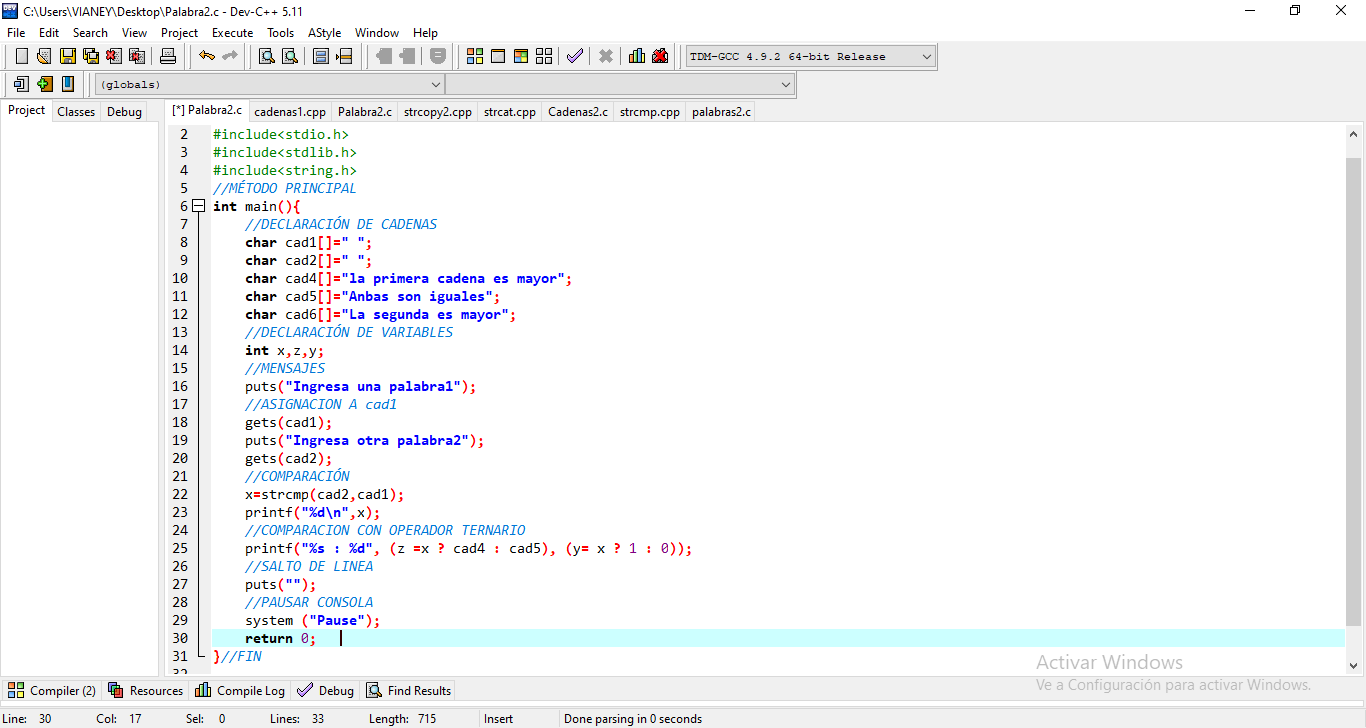
1. Escribe un programa que reciba por teclado dos palabras y te indique cuál de ellas es mayor y cuál es la menor.

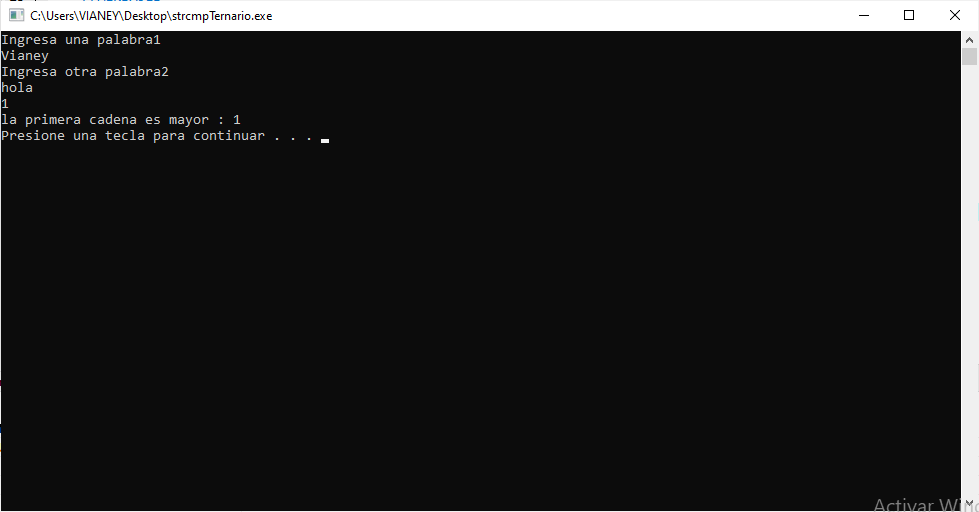
En caso de ser iguales, que imprima la leyenda 'ambas palabras son iguales'.

Di pauta a la creación de un nuevo proyecto.

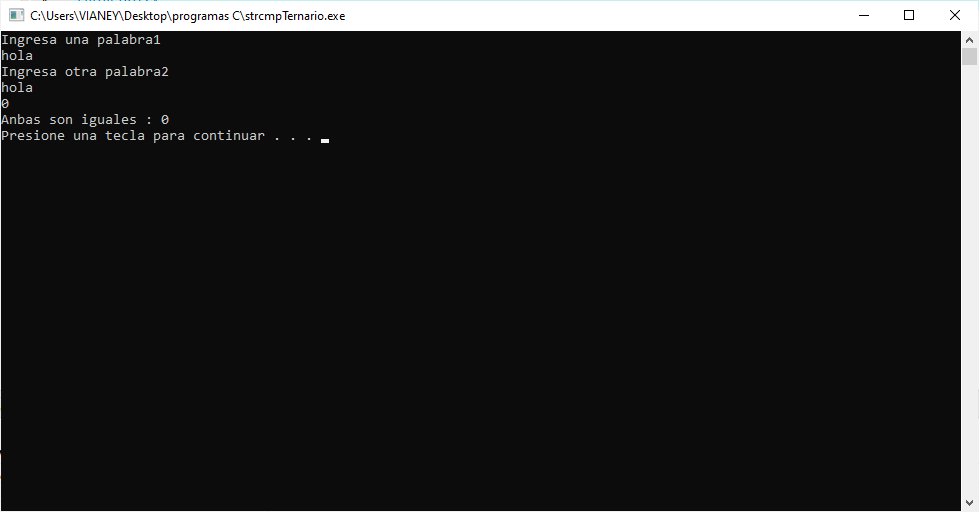


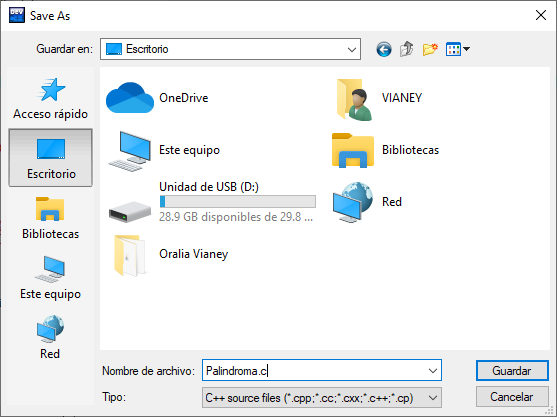
En el declare dos cadenas sin tamaño y vacías, puesto que son las que se llenaran con las palabras que ingrese el usuario, posteriormente cree tres más que serán las que indicaran las instrucciones, por medio de un mensaje solicito al usuario ingrese dos palabras y las asigno a las respectivas cadenas, mediante strncmp comparo cuál de ellas es la mayor o si son iguales y así mismo asigno el resultado en la variable x, posteriormente mediante los operadores ternarios logro imprimir las cadenas según sea el caso.



Al ejecutar mi programa este es el resultado, el cual es correcto.

A continuación, se introducen dos palabras que son iguales



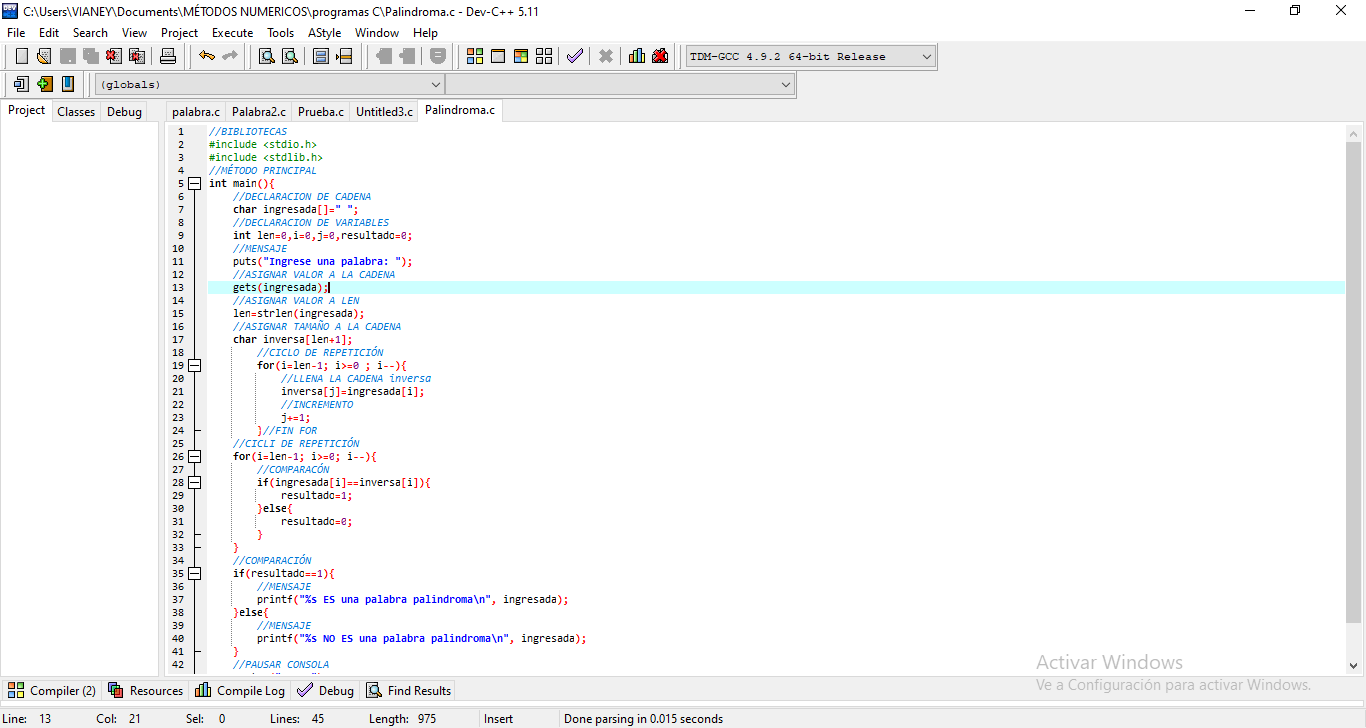
Cree un nuevo programa llamado polindromas y lo guarde con (.c)

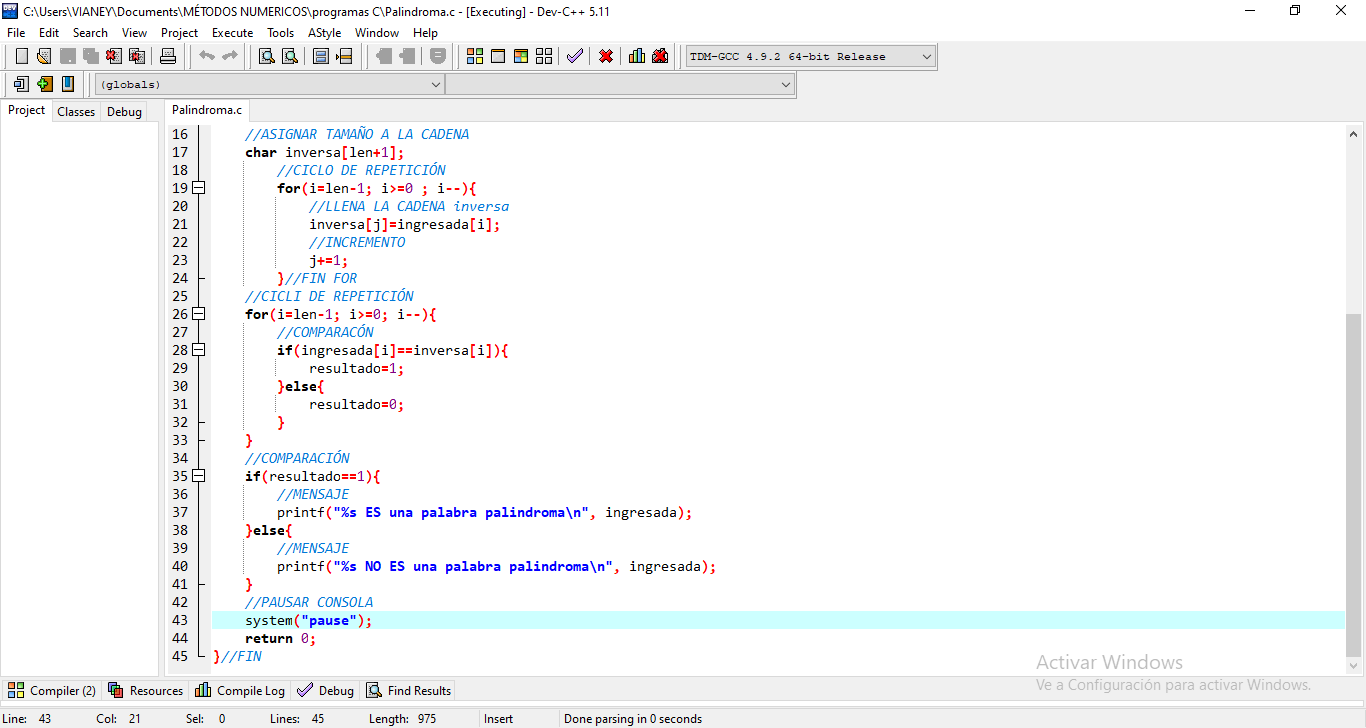
Ejercicio 5

a) Crea un programa que detecte una palabra palíndroma.

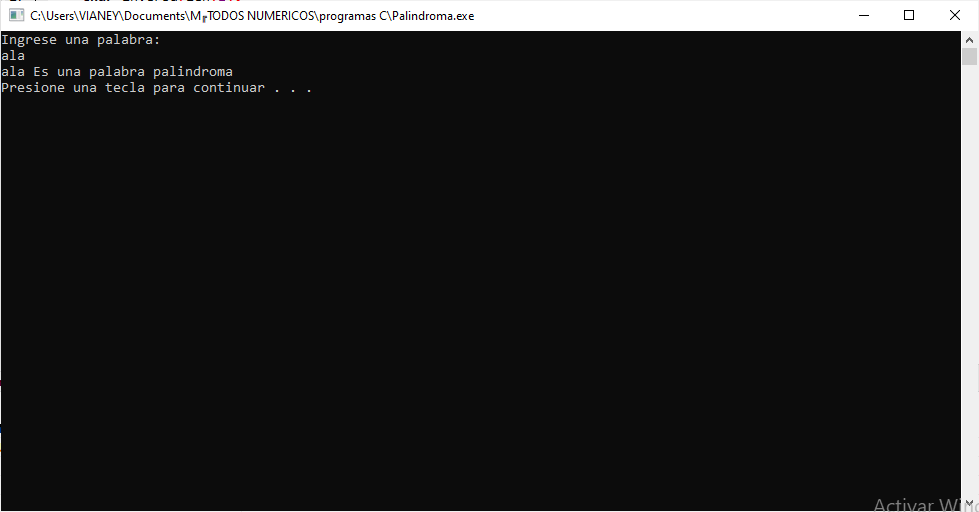
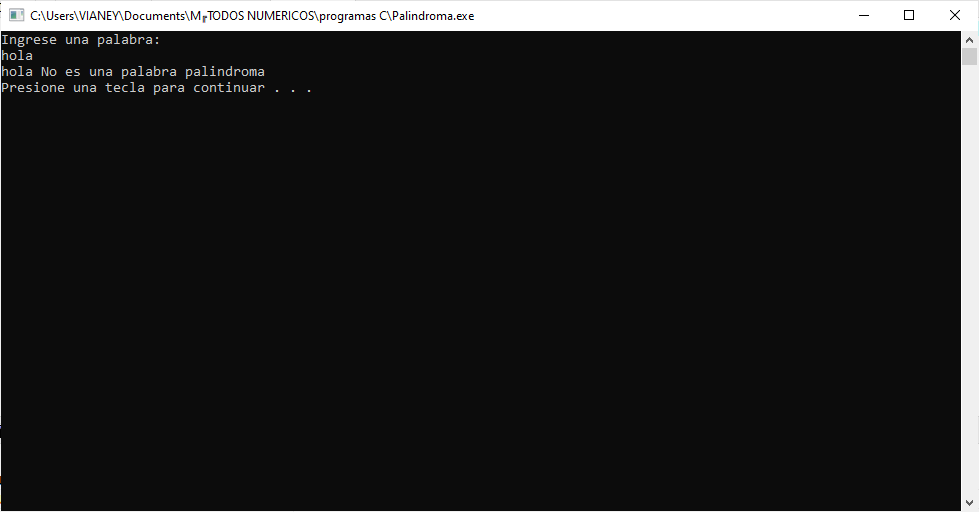
b) Los polindromas son palabras que se leen igual de izquierda a derecha que de derecha a izquierda.

A continuación desarrollo mi código conde primero solicito que se ingrese una palabra por medio de un mensaje en pantalla y esta palabra la asigno a la cadena llamada ingresada[], posteriormente obtengo su longitud de esta palabra y este lo asigno como amaño a la cadena inversa[], a continuación por medio del ciclo repetitivo for asigno el valor a la cadena inversa de tal manera que cada vez que se recorre el ciclo se comienza a signara el valor de la última posición de la cadena ingresada a la inversa y así sucesivamente el penúltimo carácter en la segunda posición de la cadena inversa y el ante penúltimo carácter de la cadena ingresada en la tercera posición de la cadena inversa hasta terminar, mientras que con el segundo for compara estas dos cadenas posiciona por posición y si son iguales imprime sus respectivos mensajes





Al ejecutar el programa e ingresar una palabra este es el resultado según sea el caso.

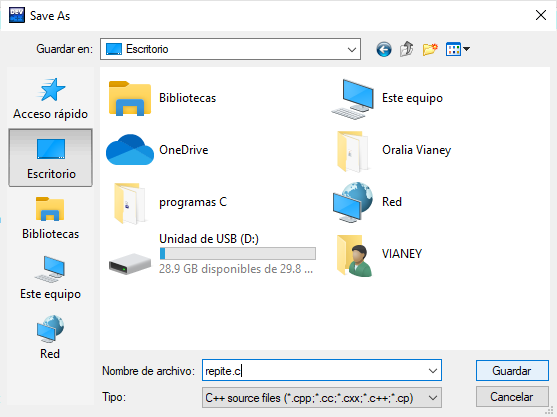


Ejercicio 6

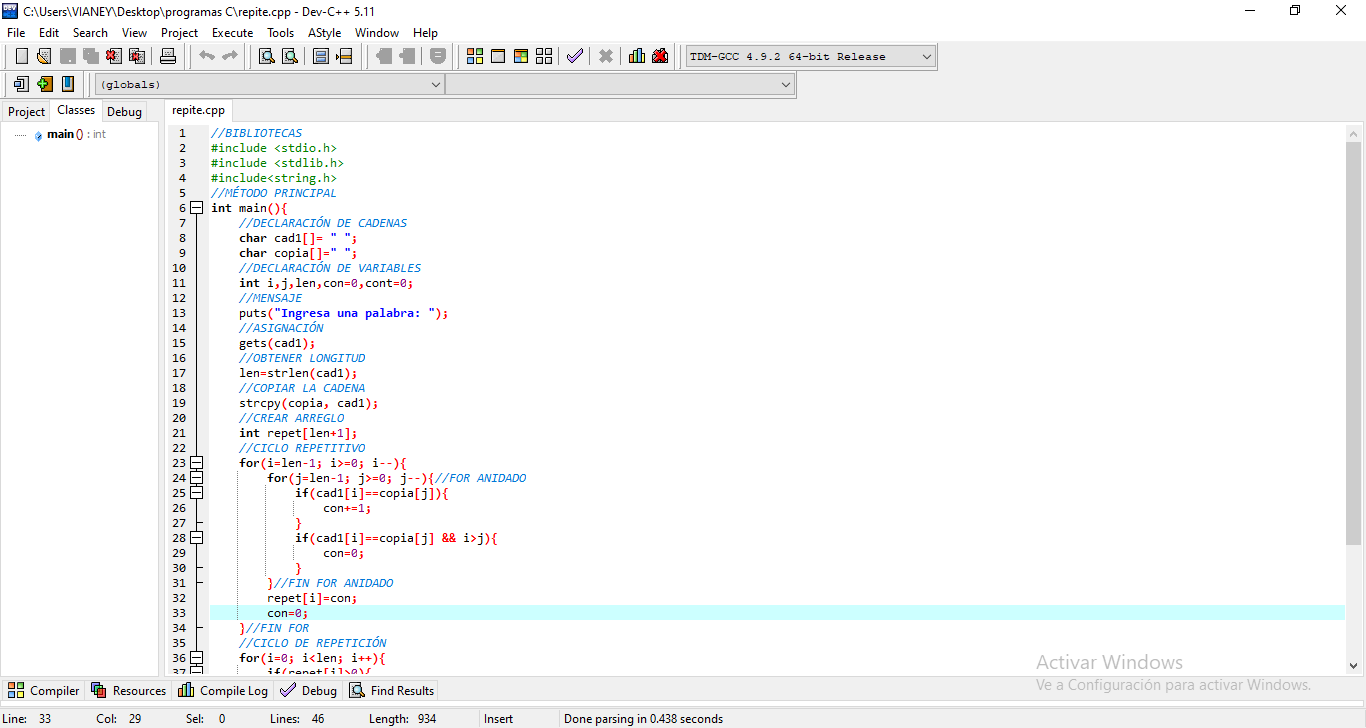
Crea un programa que cuente cuantas ocurrencias de cada letra contiene una palabra.

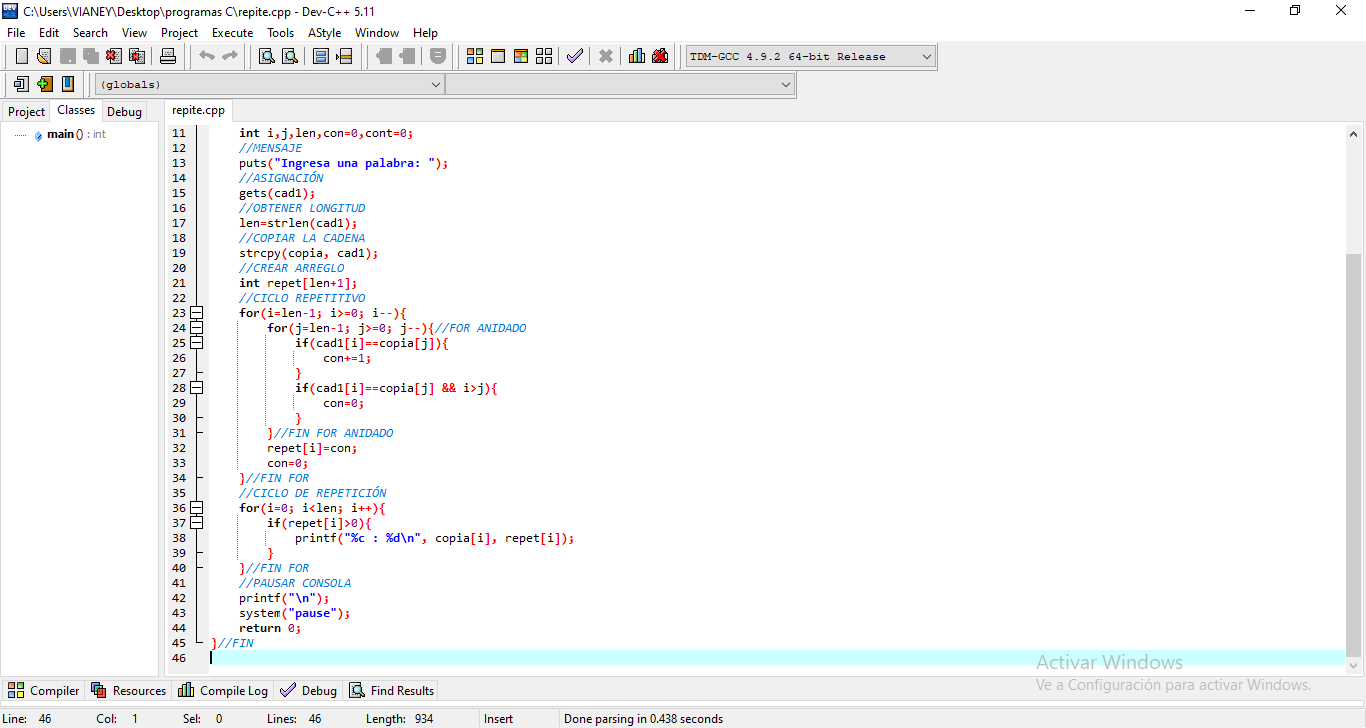
Ejemplo: Palabra P→1 a →3 l →1 b →1 r →1

Para demostrar su funcionamiento cree un nuevo proyecto nombrado de la siguiente manera

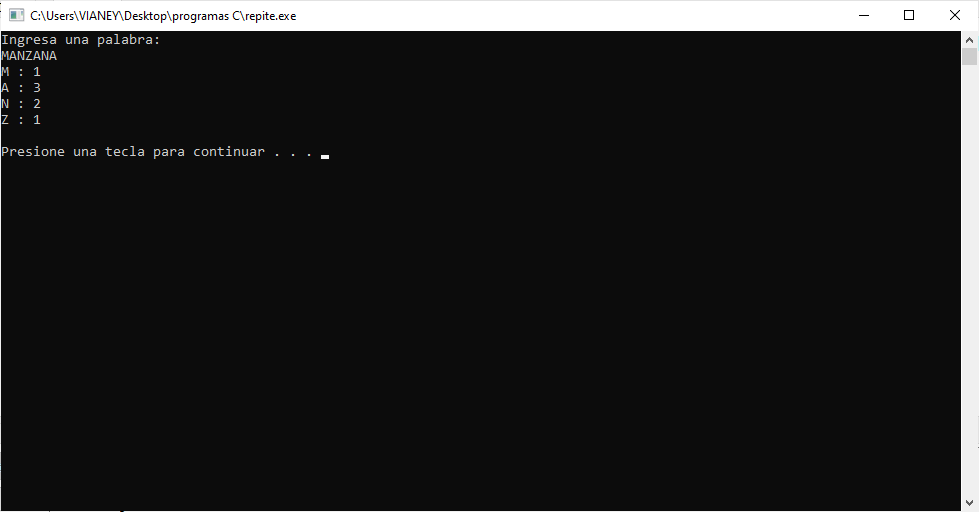


En el funcionamiento del código primero se solicitan que se ingrese una palabra y esta es almacenada, además de que hacemos una copia de ella en otra cadena, posteriormente obtenemos la longitud de esta cadena original, mientras que con el for anidado se evaluó posición por posición de las cadenas de atrás hacia delante en caso de que la labra se repita el contador aumenta de uno en uno y este se va almacenando en el arreglo repet, además de que por medio del obtenemos que carácter es el que se repitió, según sea el que se encuentre en la posición de i y finalmente con el ultimo for se imprime, la cadena copia que en ese momento solo contiene los caracteres que se repiten y repet que es el arreglo que contiene el número de veces que se repite dicho carácter.





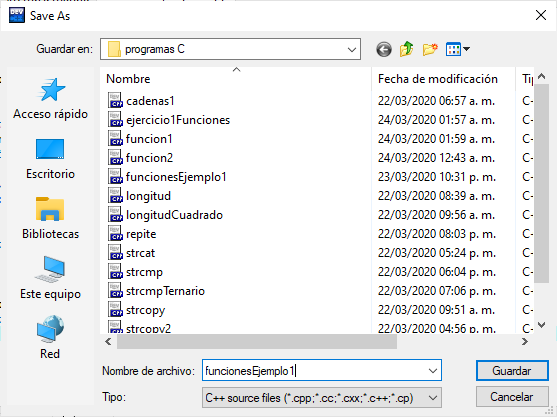
Al ejecutar el programa este es el resultado a mostrar en pantalla.



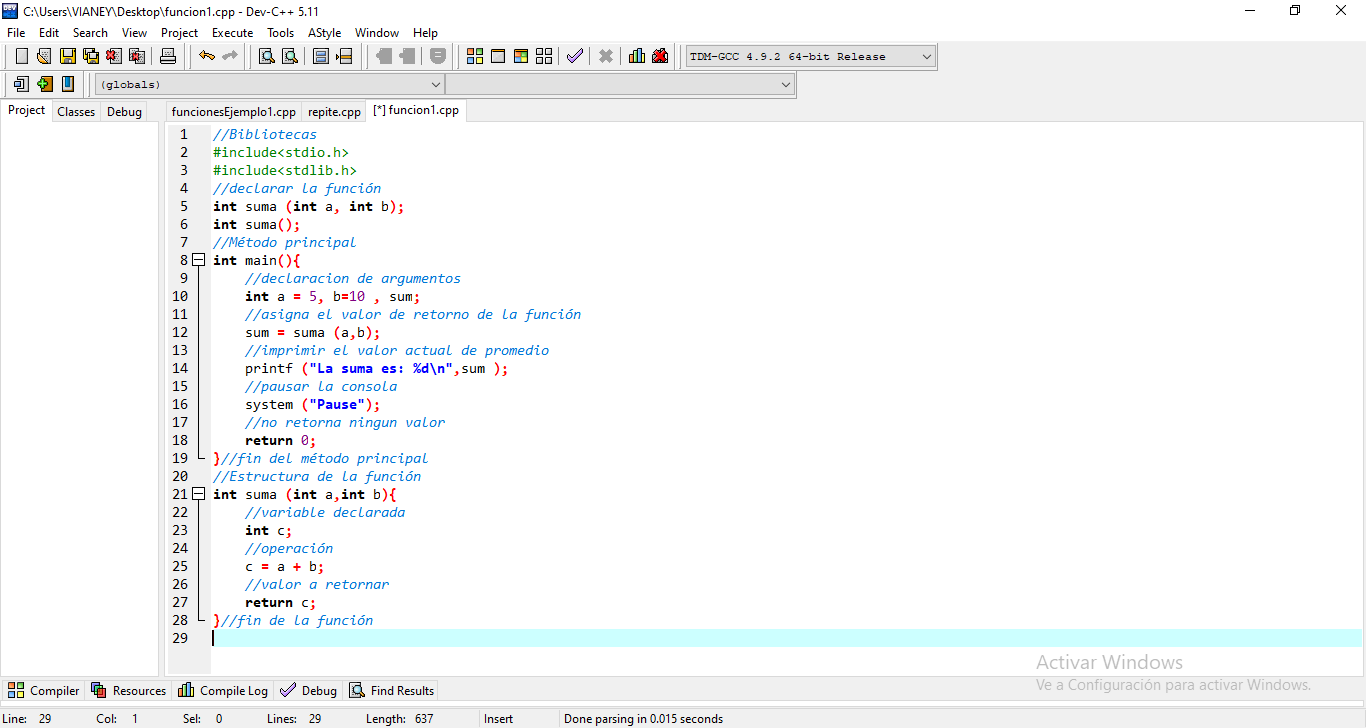
# Lamina13 Funciones

Funciones

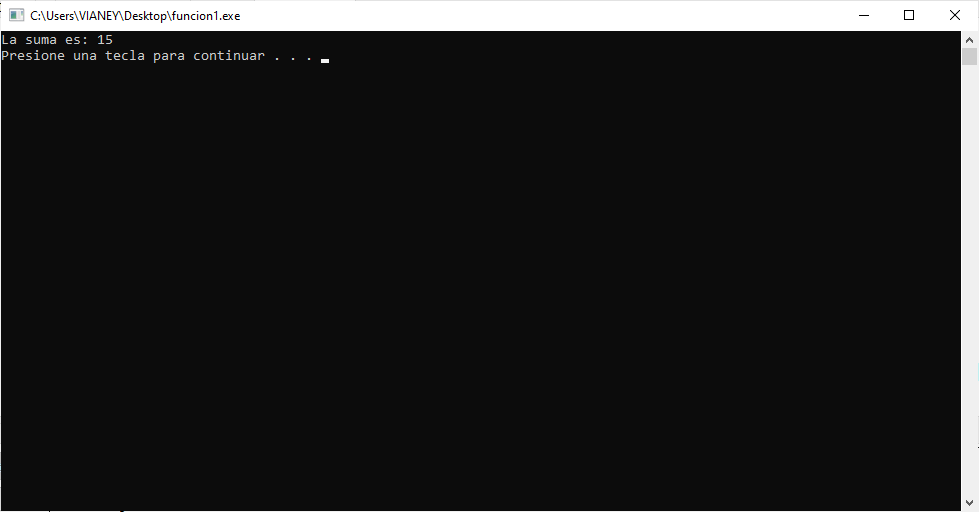
Para dar pauta a los ejercicios y una vez comprendidos y analizados los conceptos, así como funcionamiento básico de las funciones cree este nuevo programa dando clic derecho sobe File>New>Source File y posteriormente para guardarlo nuevamente doy clic en File>Save As, coloco el nombre de mi proyecto y doy clic sobre el botón guardar.



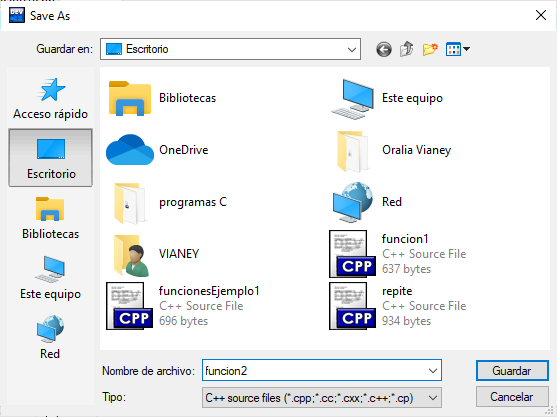
En este código se presentan una función que es de tipo entero, por lo tanto su valor a retornar se sabe que es un numero entero, por medio del método principal declaramos los argumentos los cuales se utilizaran en el desarrollo tanto de la función, como del resto del código, ya que al entrar en la función son declarados dichos valore como parámetros, es decir los valores que se establecen en el método principal, es decir los valores asignados en a=5 y b=10 pasan a la función y dentro de ella realiza la operación correspondiente y el resultado de dicha operación lo retorna, en este caso en el método principal dicho valor de retorno, se manda a llamar y es asignado al argumento mus, el cual posteriormente se imprime en pantalla, al igual que un mensaje que hace alucino al calor obtenido



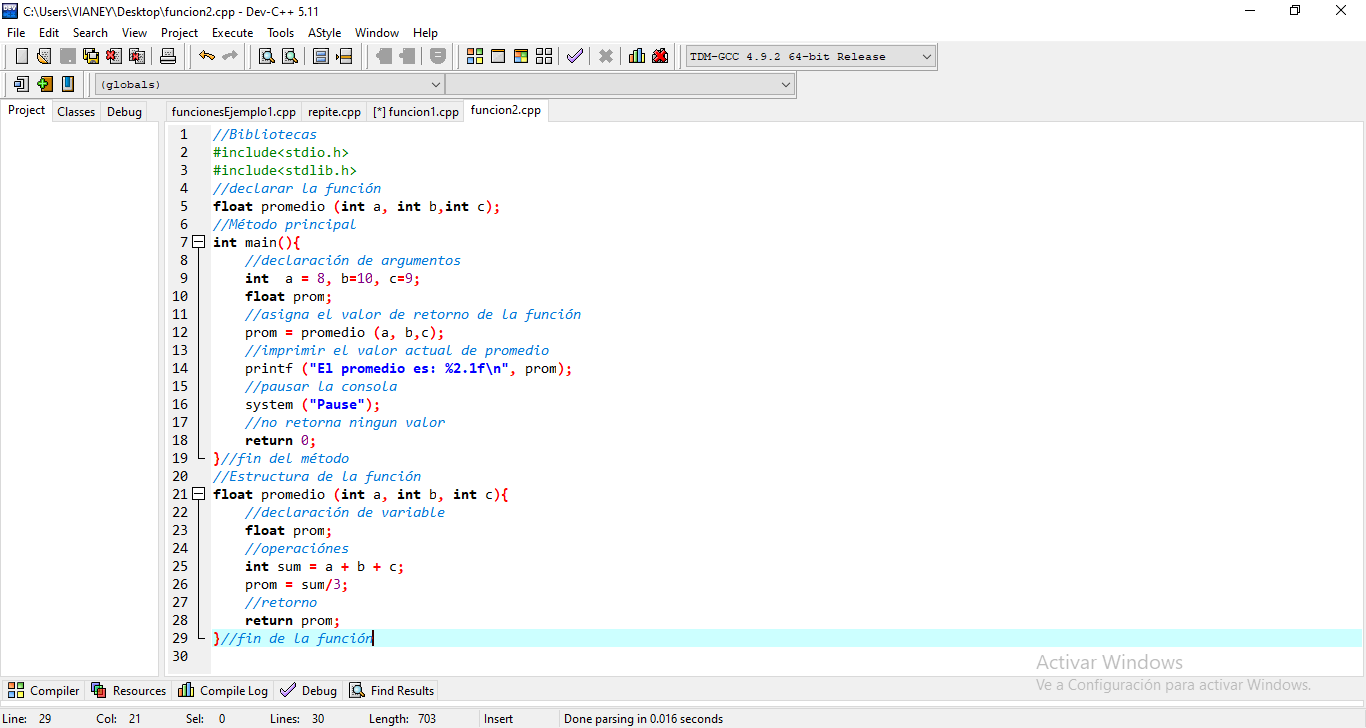
Al ejecutar el resultado es el siguiente.



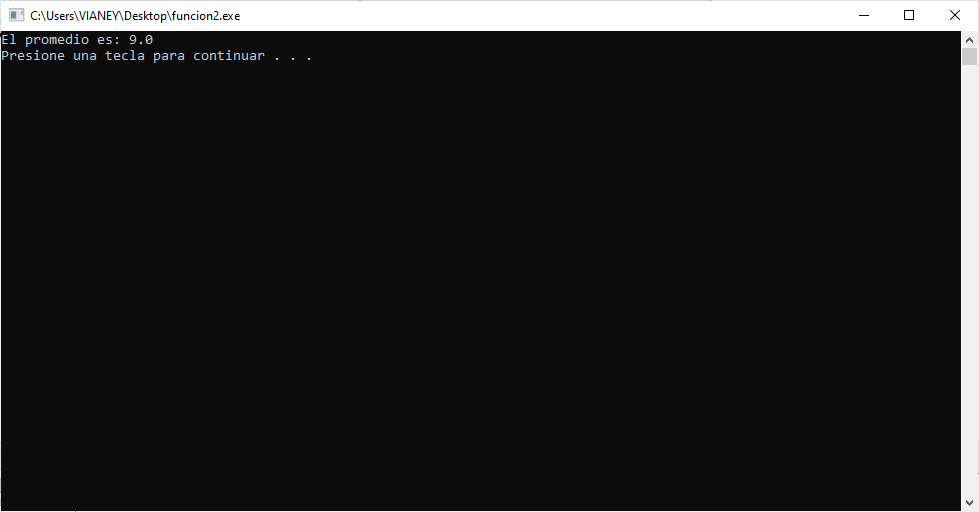
Para el siguiente ejercicio lo guarde con el presente nombre siguiendo los pasos correspondientes al ejercicio anterior.



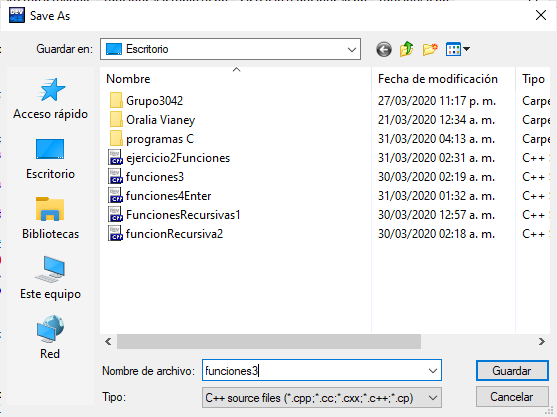
La función que realiza es similar a la anterior, solo que en su caso lleva un parámetro más, además de ello una de las partes más interesante es que una función de cierto tipo de dato puede tener parámetros de otro tipo de dato diferente, en esta función se realiza algo similar a la anterior puesto que suma todos los valores que poseen los argumentos y además los divide entre la cantidad de datos, para obtener el promedio y una vez que se obtiene lo manda como retorno de la función, para posteriormente asignarlo a un argumento distinto dentro del método principal y finalmente imprimirlo.



Al ejecutar el programa el resultado es correcto. Puesto que 8 más 10 más 9 es igual a 27 y se divide entre 3, esto da como resultado 9 en promedio y es el que se muestra en pantalla

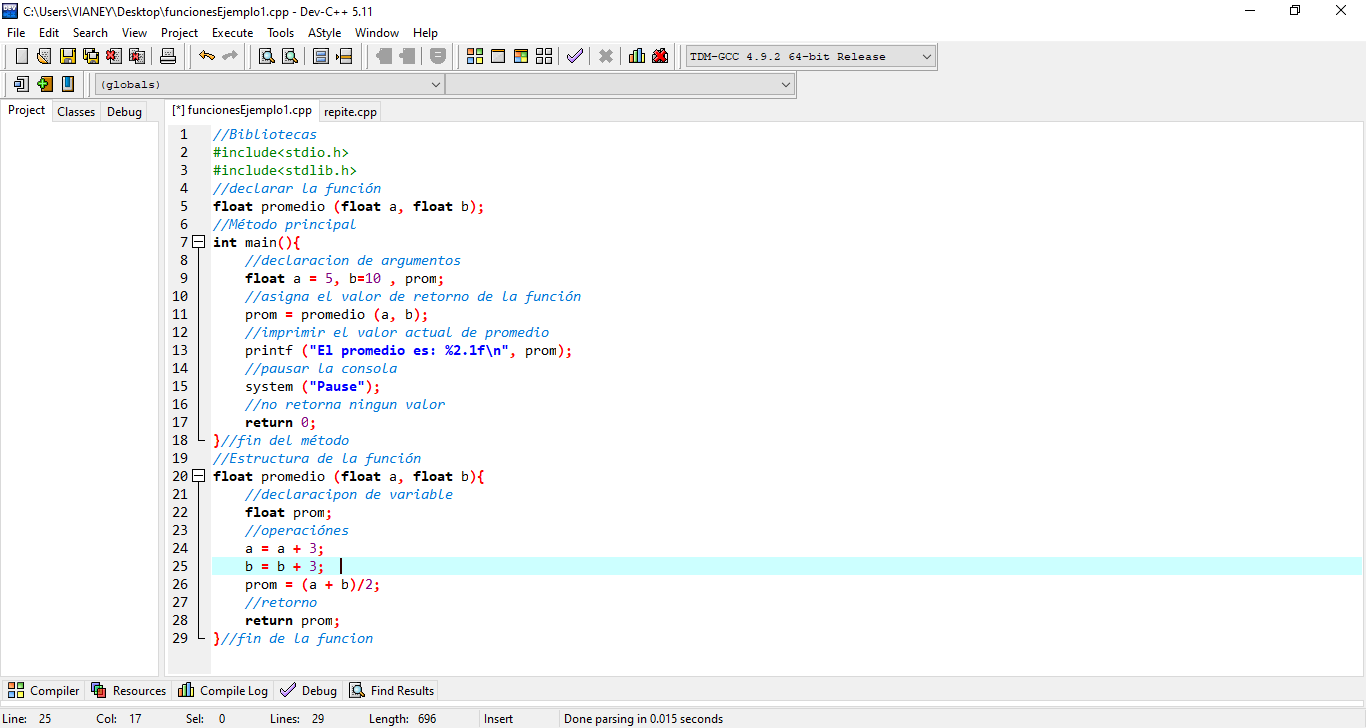


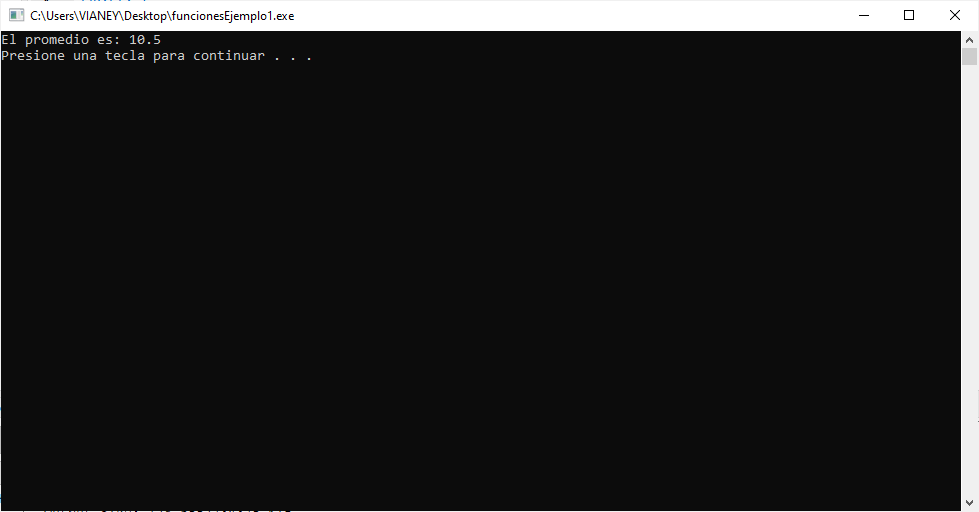
Cree el siguiente programa siguiendo los pasos anteriores y nombre el programa de la siguiente manera



Posteriormente ingreso al área de trabajo donde desarrolle el siguiente código el cual funciona de la siguiente manera.

Primero se declara la función para que esta pueda ser utilizada tanto en el método principal como en la función como tal, posteriormente declaro dentro de mi método principal los argumentos, los cuales tienen una diferencia a comparación de las variables, puesto que los argumentos pueden ser utilizados en una o varias funciones, mientras que las variables locales no, posteriormente asignamos el valor de este retorno de nuestra función al argumento prom, por lo tanto al mandar a llamar a la función, esta se ejecuta de la siguiente manera si la variable a que pasa como parámetro a la función vale 5 y se le suman 3 termina valiendo 8, mientras que b bale 10 y le suman 3 ahora vale 13 y por medio de la tercera operación suma ambas variables y las divide entre 2, retornando este valor y es ahora el valor actual de prom, finalmente por medio de un mensaje se manda a imprimir dicho mensaje

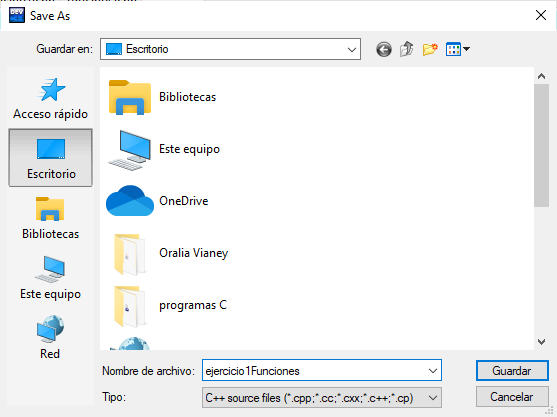


Al ejecutar el programa este es el resultado obtenido en pantalla.

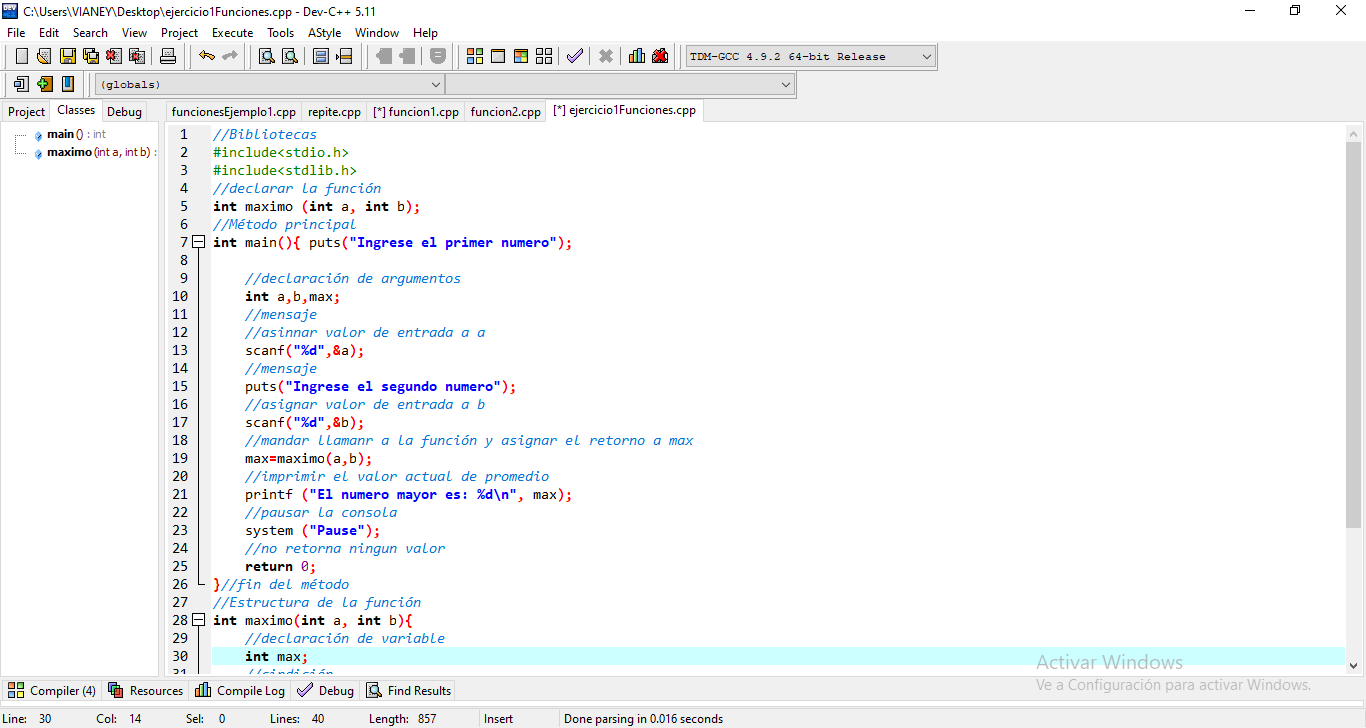
Ejercicio 1

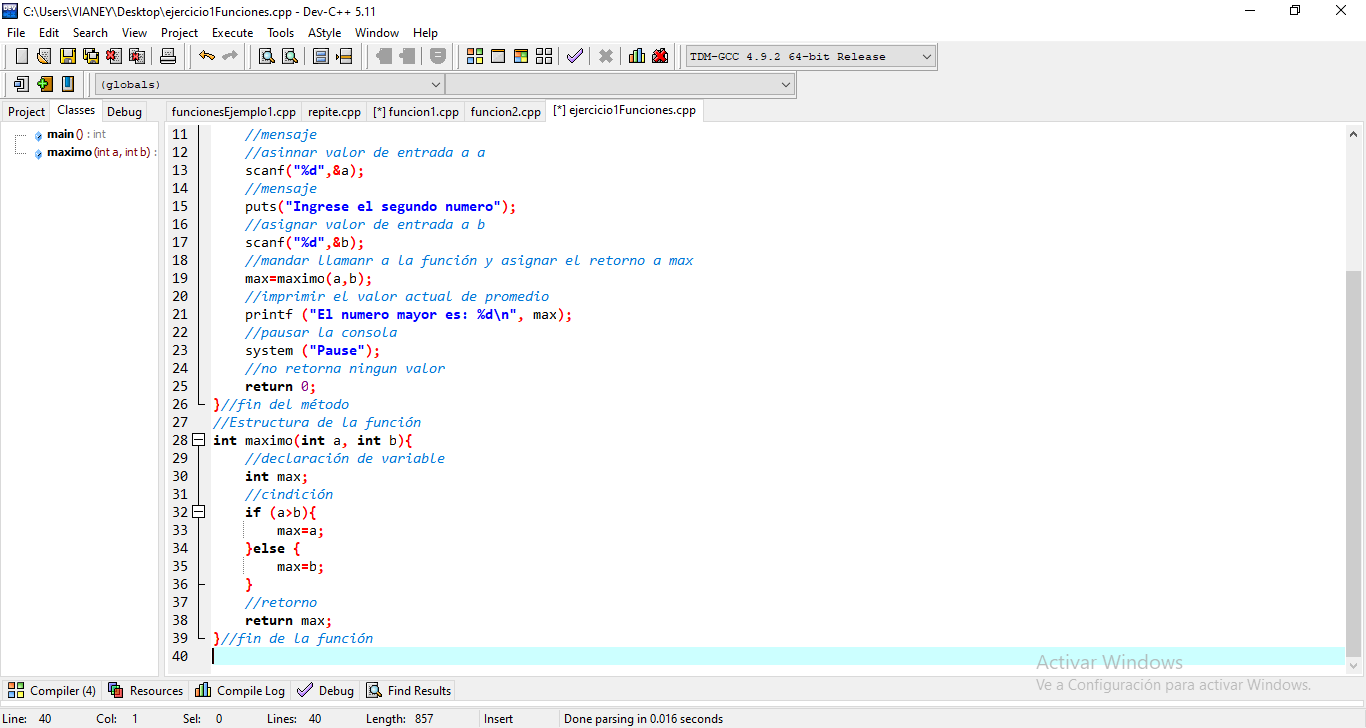
Escribir una función que se llame **máximo** que reciba dos número por parámetros y que regrese el mayor de ellos.

A continuación creo mi nuevo programa al cual nombre ejercicio1Funciones, posteriormente di clic sobre el botón guardar.

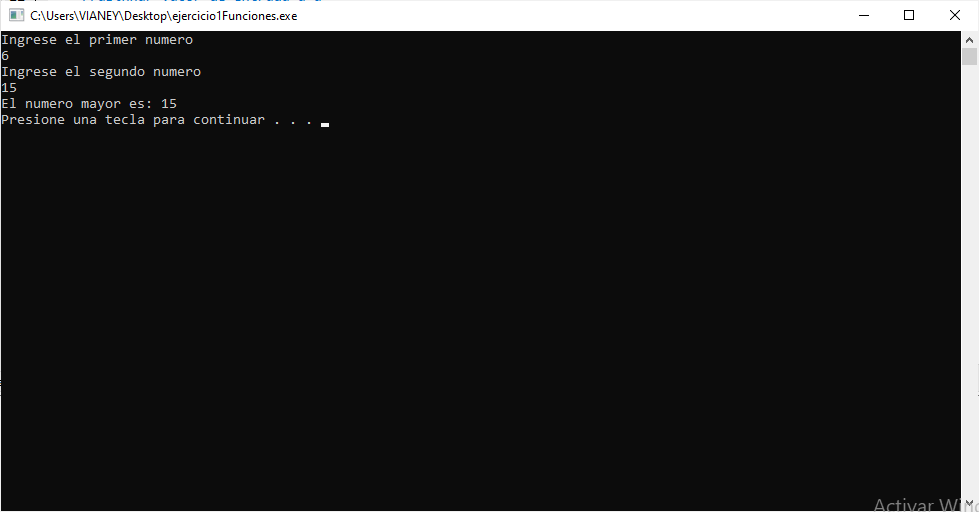


Una vez ubicados dentro del área de trabajo se muestra la función que requerir, primero solicito al usuaria que ingrese nos números, los cuales amaceno en una variable respectivamente, posteriormente se diseña la función, la cual es de tipo entero ya que solo requiero que devuelva uno de los dos valores ingresados, por lo tanto estos argumentos son los parámetros, y por medio de un if evaluo si el primer numero ingresado es mayor entonces la variable max toma ese valor, en caso contrario se asigna el valor del segundo valor introducido a la variable max y finalmente se retorna el valor de max, dicho valor de retorno es asignada al argumento max y finalmente se imprime dicho valor,





Al ejecutar el programa este es el resultado una ves que se han introducido dos valores.



Ejercicio2

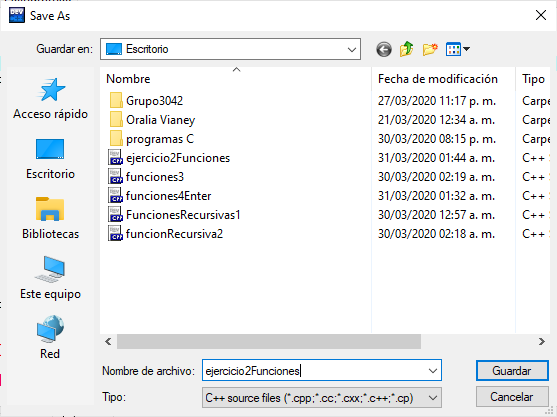
Escribir una función que reciba caracteres del teclado hasta recibir un espacio o un salto de línea (enter) y a continuación mostrar todos los caracteres en orden inverso.

Ejemplo:

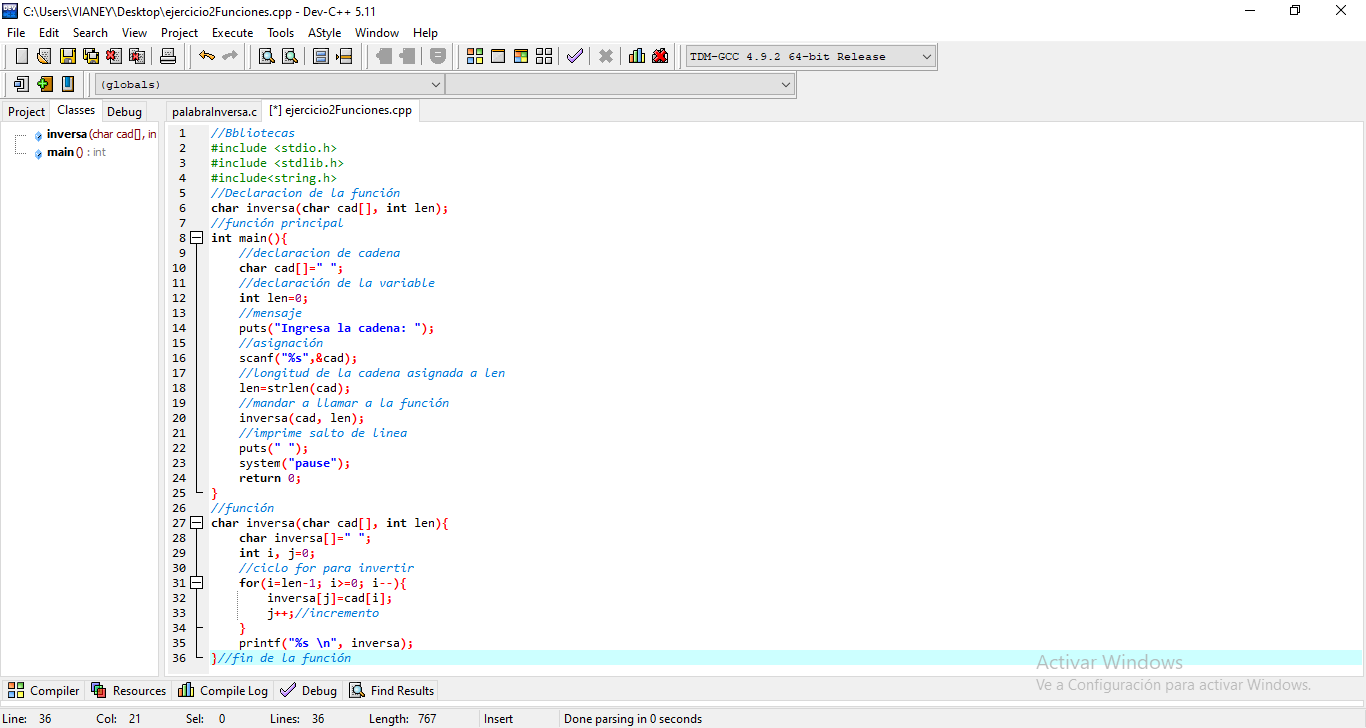
Entrada: Hola

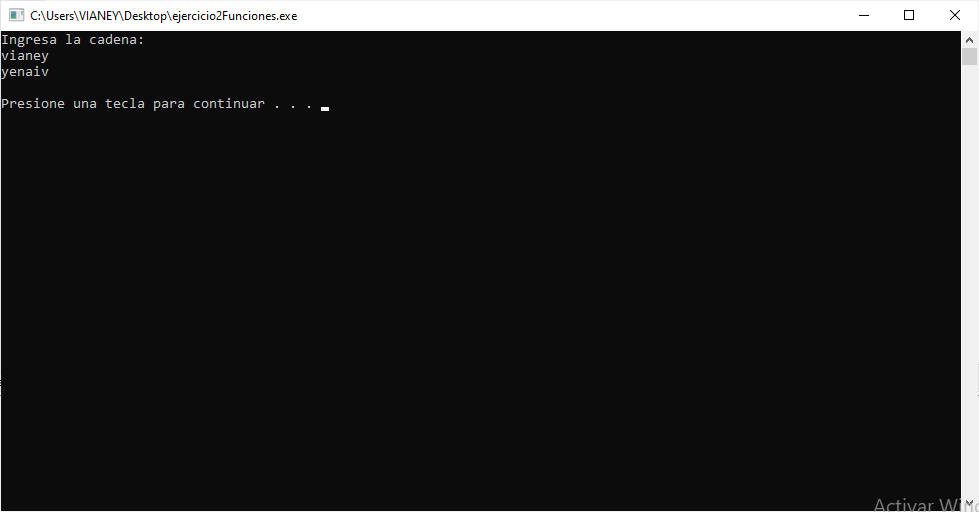
Salida: aloH

Abrí y cree mi nuevo archivo con el siguiente nombre.



Mediante el siguiente código,se solicita la cadena por medio de un mensaje, posteriormente se obtiene la longitud de dicha cadena, la cual se almacena en la variable len que pasa como parámetro a la función en la cual permite que se invierta la cadena por medio del ciclo for y es almacenada en una nueva cadena, la cual es impresa al finalizar el for y esta función finalmente es mandada a llamar en cuanto damos enter o se detecta un espacio vació en la cadena.

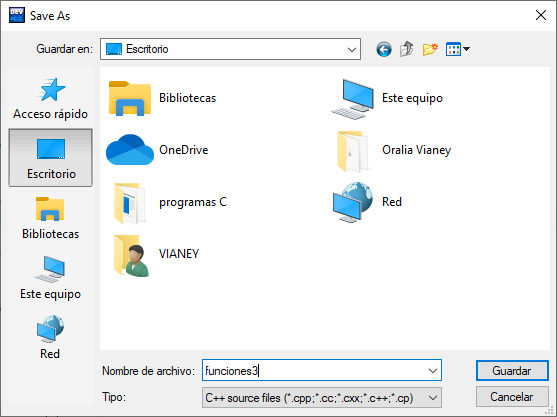


Al ejecutar este es el resultado que aparece en pantalla

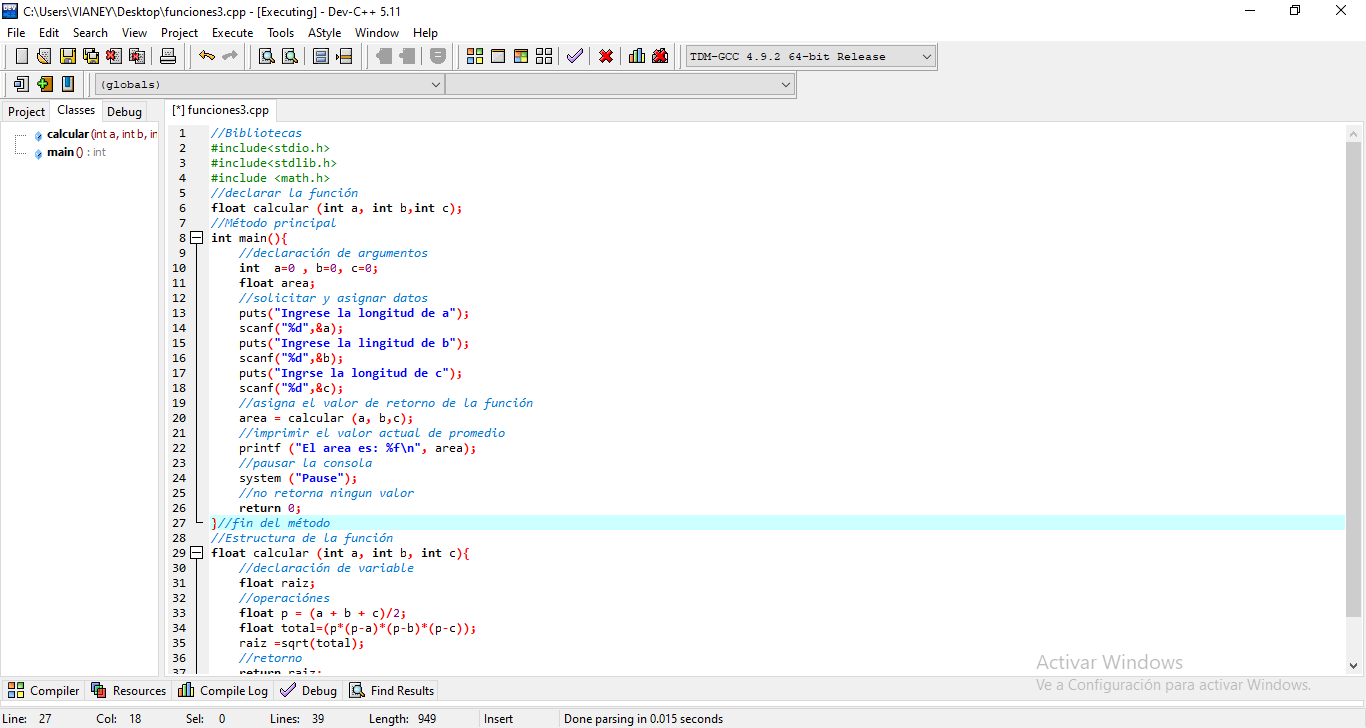
Ejercicio3

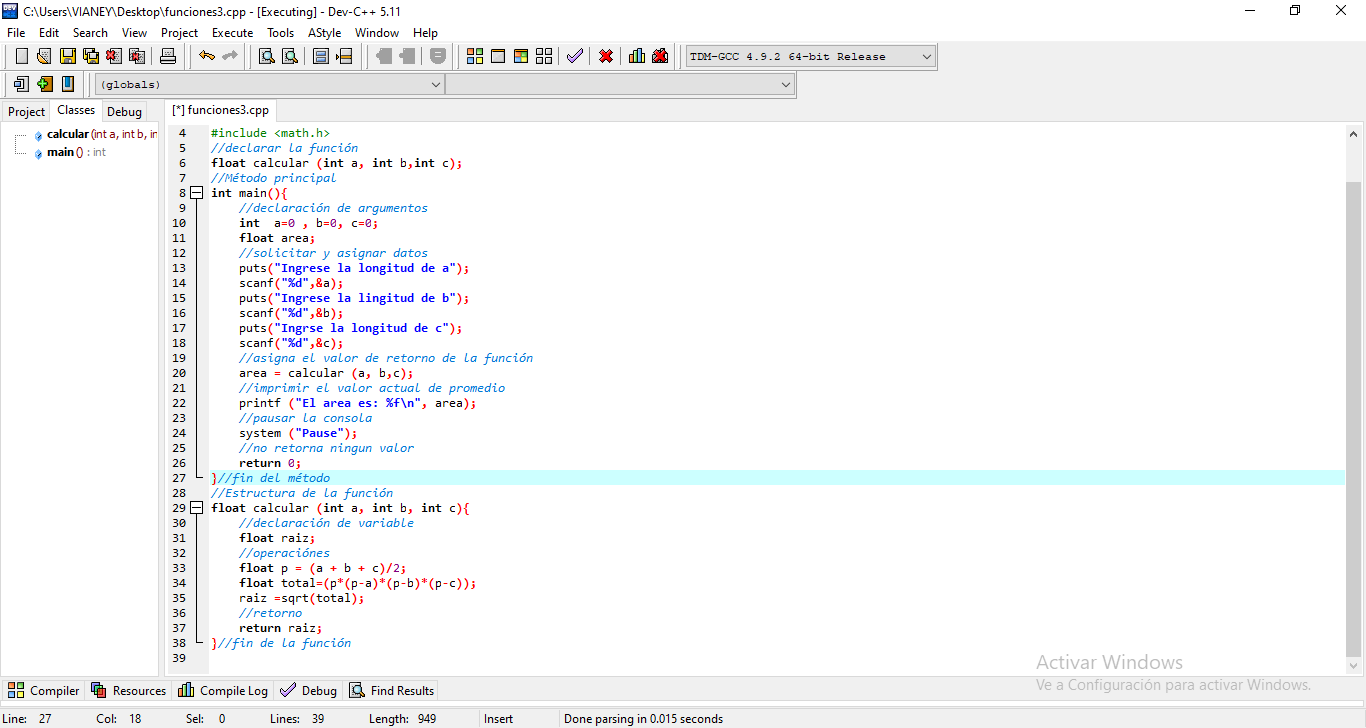
Escribir una función que tome como parámetros las longitudes de los tres lados de un triángulo (a, b, c) y devuelva el área del triángulo.

Creo mi nuevo proyecto, al cual nombre de la siguiente manera, como aparece en la imagen, con la extensión .c

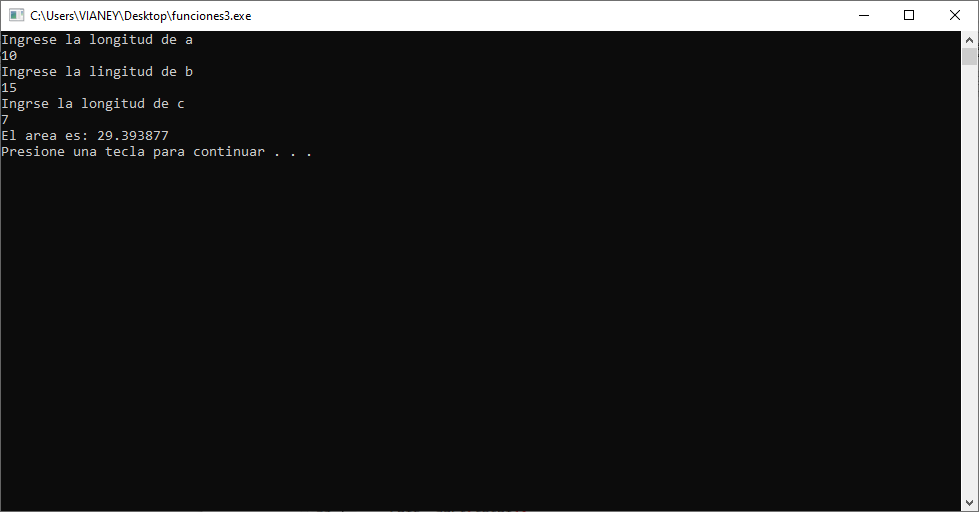


Posteriormente en nuestro programa lo que se realiza la declaración de nuestra función con sus respectivos parámetros, posteriormente por medio de puts(“”), podemos enviar un mensaje para solicitar los dados y con scanf asignamos dichos datos ingresados a sus argumentos respectivos, una ves que se tiene estos datos en la función calcular realice las operaciones necesarias para obtener el área del triangulo, tomando en cuenta que en este ejercicio se requirió de un nueva biblioteca la cual es math que permite realizar operaciones como lo es la raíz cuadrada, posteriormente este valor que se retorna de la función es asignado a la variable área y finalmente se imprime.

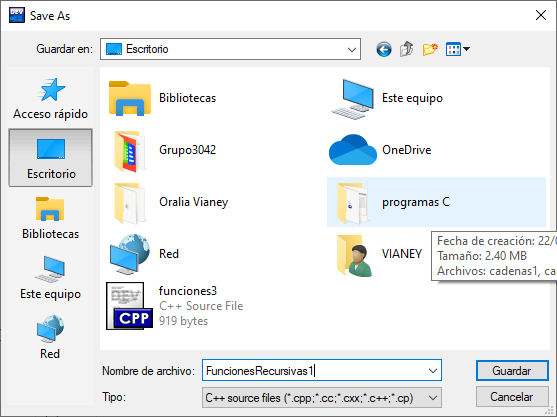




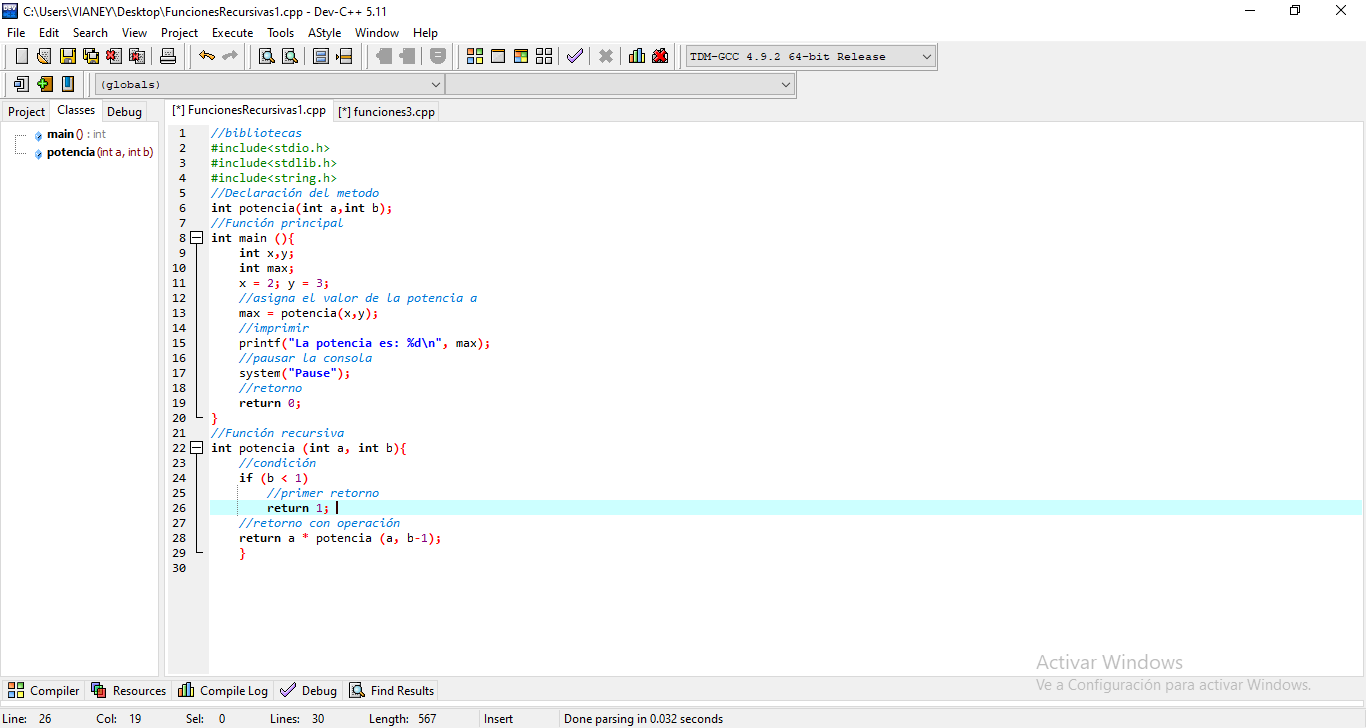
Al ingresar nuestros datos este es el valor resultante, equivalente al área del triángulo.



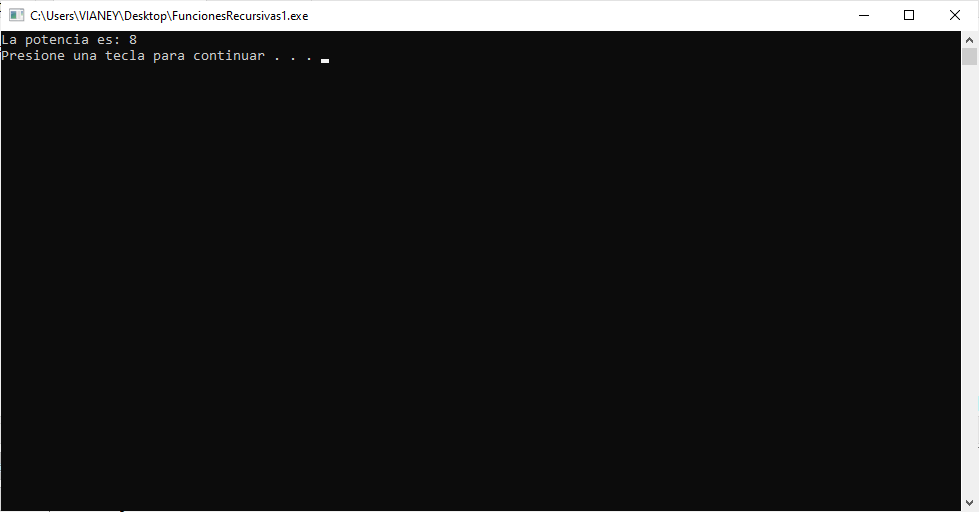
Este ejemplo se basa en una función recursiva la cual se basa en.



En este ejemplo muestra una función la cual se vuelve a llamara si misma, en base a si cumple o no una condición que en este casi es b, otra parte interesante en este ejemplo que fue explicado, es que los parámetros de la función no es necesario que sean llamados de la misma manera que los argumentos de los cuales obtienen su valor como lo es x y y y los parámetros son a y b, de igual manera como se puede observar las funciones recursivas pueden tener dos retornos uno por si se cumple la condición y otro si no.



En base a la función explicada este es el resultado.



Ejercicio 4

Haz un programa con funciones recursivas que calcule el factorial de un número N ingresado desde teclado.

EJEMPLO:

N = 5

5! = 4! \* 5

4! = 3! \* 4

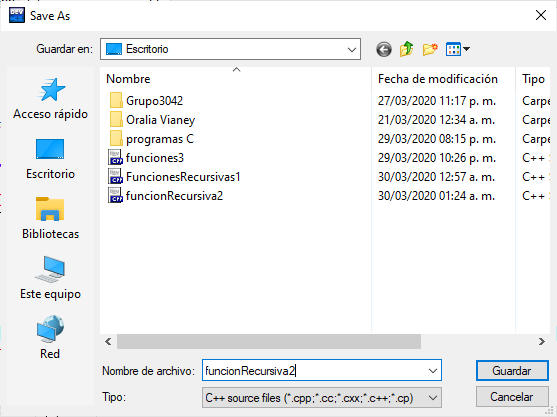
3! = 2! \* 3

2! = 1! \* 2

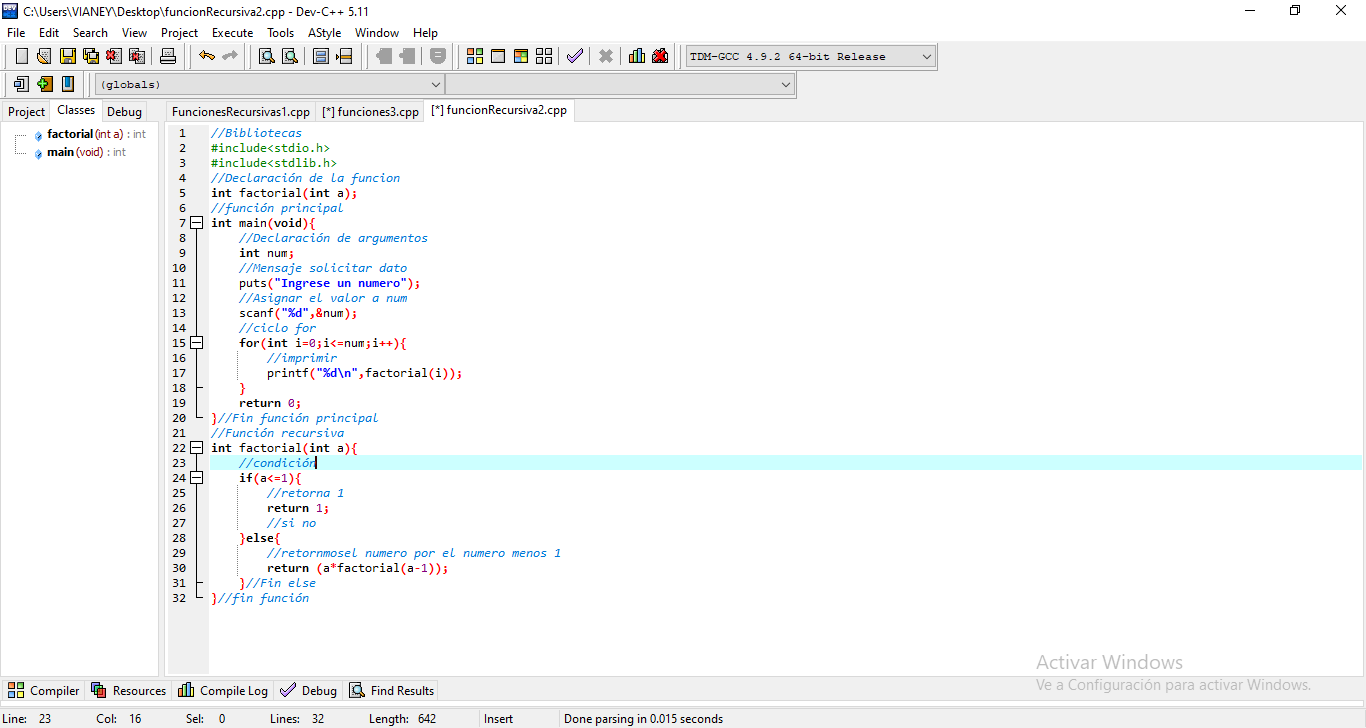
1! = 0! \* 1

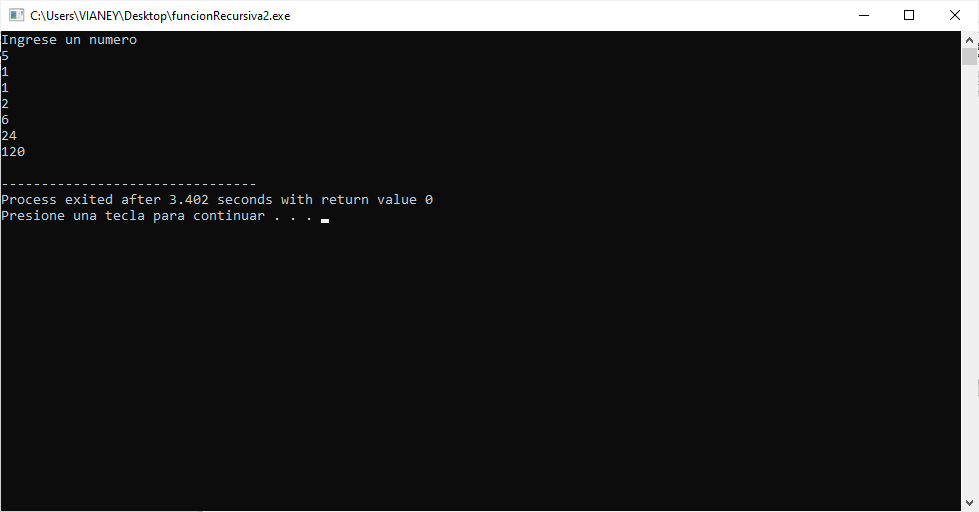
0! = 1

Por lo tanto lo primero a realizar es crear mi nuevo proyecto con el siguiente nombre.



A continuación en mi programa declaro las bibliotecas necesarias a utilizar,posteriormente declaro mi función recursiva, en esta ovación con un solo parámetro, posteriormente solicito al usuario ingrese un numero por medio de un mensaje y con el scanf, el valor ingresado lo asigno a el argumento num, el cual pasa como valor al parámetro de la función, en la cual nos dice que si el numero a evaluar es menor o igual a 1, entonces el retorno será 1, puesto que el factorial de 1 y o es 1, en caso contrario entonces se multiplica el número actual por nuevamente el parámetro de la función menos uno, por lo tanto al recorrer la primera vez es 5 y al llamar al parámetro menos 1 lo multiplica por cuatro ahora entra cuatro al condicional y así repite el proceso, posteriormente por medio del for se imprime, cada vez que se manda a llamar a la función por medio de la posición I



Al ejecutar el programa el resultado es el siguiente.

**V. Conclusiones:**

Por medio de esta práctica aprendía a identificar el funcionamiento de las cadenas, desde una perspectiva diferente a la que se había tomado anteriormente, puesto que se tomó como un vector, de esta manera también se pudieron aplicar nuevas funciones a una cadena, como lo fue el copiar, concatenar , concatenar e incluso comparar cual es mayor que otra aplicándolas en diferentes programas analizando el funcionamiento de estas, además de que se retroalimento el uso del for tanto simple como anidado.

Por otra parte, por medio de las funciones pude aprender su uso para resolver problemas utilizando parámetros y reutilizando variables, lo cual fue desde mi perspectiva el tema más interesante al asignar valores por medio de parámetros.