|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Claudia Rodriguez Espino |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 4 |
| *No de Práctica(s):* | 7 |
| *Integrante(s):* | Sanchez Escamilla Hector |
|  |  |
| *No. de Equipo de cómputo empleado* | 35 |
| *Semestre:* | 2 |
| *Fecha de entrega:* | 29/03/2019 |
| *Obervaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Práctica de estudio 07: Fundamentos de Lenguaje C

**Objetivo:**

Elaborar programas en lenguaje C utilizando las instrucciones de control de tipo *secuencia*,

para realizar la declaración de variables de diferentes tipos de datos, así como efectuar

llamadas a funciones externas de entrada y salida para asignar y mostrar valores de

variables y expresiones.

**Actividades:**

􀂃 Crear un programa en lenguaje C que tenga definidas variables de varios tipos, se

les asigne valores adecuados (por lectura o asignación directa) y muestre su valor

en la salida estándar.

􀂃 En un programa en C, asignar valores a variables utilizando expresiones

aritméticas; algunas con uso de cambio de tipo (cast)

􀂃 Elaborar expresiones relacionales/lógicas en un programa en C y mostrar el

resultado de su evaluación.

**Introducción**

Una vez que un problema dado ha sido analizado (se identifican los datos de entrada y la

salida deseada), que se ha diseñado un algoritmo que lo resuelva de manera eficiente

(procesamiento de datos), y que se ha representado el algoritmo de manera gráfica o

escrita (diagrama de flujo o pseudocódigo) se puede proceder a la etapa de codificación.

La codificación se puede realizar en cualquier lenguaje de programación estructurada,

como lo son Pascal, Python, Fortran o PHP. En este curso se aprenderá el uso del lenguaje

de programación C.

**Lenguaje de programación C**

El proceso de desarrollo del lenguaje C se origina con la creación de un lenguaje llamado

BCPL, que fue desarrollado por Martin Richards.

BCPL tuvo influencia en un lenguaje llamado B, el cual se usó en 1970 y fue inventado por

Ken Thompson, esto permitió el desarrollo de C en 1971, el cual lo inventó e implementó

Dennis Ritchie.

C es un lenguaje de programación de propósito general que ofrece como ventajas

economía de expresión, control de flujo y estructuras de datos y un conjunto de

operadores.

C es un lenguaje de propósito general basado en el paradigma estructurado. El teorema

del programa estructurado, demostrado por Böhm-Jacopini, dicta que todo programa

puede desarrollarse utilizando únicamente 3 instrucciones de control:

􀂃 Secuencia

􀂃 Selección

􀂃 Iteración

Por otro lado, C es un lenguaje compilado, es decir, existe un programa (llamado

compilador) que, a partir de un código en lenguaje C, genera un código objeto (ejecutable).

Para crear un programa en C se siguen tres etapas principales: edición, compilación y

ejecución.

* Edición: Se escribe el código fuente en lenguaje C desde algún editor de textos.
* Compilación: A partir del código fuente (lenguaje C) se genera el archivo en lenguaje

máquina (se crea el programa objeto o ejecutable).

* Ejecución: El archivo en lenguaje máquina se puede ejecutar en la arquitectura

correspondiente.

Un programa en C consiste en una o más funciones, de las cuales una de ellas debe

llamarse main( ) y es la principal.

**Declaración de variables**

Para declarar variables en C se sigue la siguiente sintaxis:

[modificadores] tipoDeDato identificador [= valor];

Por lo tanto, una variable puede tener modificadores (éstos se analizarán más adelante y

son opcionales), debe declarar el tipo de dato que puede contener la variable, debe

declarar el identificador (nombre o etiqueta) con el que se va a manejar el valor y se puede

asignar un valor inicial a la variable (opcional).

También es posible declarar varios identificadores de un mismo tipo de dato e

inicializarlos en el mismo renglón, lo único que se tiene que hacer es separar cada

identificador por comas.

tipoDeDato identificador1[= valor], identificador2[= valor];

**Tipos de datos**

Los tipos de datos básicos en C son:

* Caracteres: codificación definida por la máquina.
* Enteros: números sin punto decimal.
* Flotantes: números reales de precisión normal.
* Dobles: números reales de doble precisión.

**Identificador**

Un identificador es el nombre con el que se va a almacenar en memoria un tipo de dato.

Los identificadores siguen las siguientes reglas:

Debe iniciar con una letra [a-z].

􀂃 Puede contener letras [A-Z, a-z], números [0-9] y el carácter guión bajo (\_).

NOTA: A pesar de que variables como 'areadeltriangulo' o 'perimetro\_del\_cuadrado' son

declaraciones válidas como identificadores, es una buena práctica utilizar la notación de

camello para nombrar las variables como convención.

La biblioteca 'stdio.h' contiene diversas funciones tanto para imprimir en la salida estándar

(monitor) como para leer de la entrada estándar (teclado).

**printf** es una función para imprimir con formato, es decir, se tiene que especificar entre

comillas el tipo de dato que se desea imprimir, también se puede combinar la impresión

de un texto predeterminado:

**printf("El valor de la variable real es: %lf", varReal);**

**scanf** es una función que sirve para leer datos de la entrada estándar (teclado), para ello

únicamente se especifica el tipo de dato que se desea leer entre comillas y en qué variable

se quiere almacenar. Al nombre de la variable le antecede un ampersand (&), esto indica

que el dato recibido se guardará en la localidad de memoria asignada a esa variable.

**scanf ("%i", &varEntera);**

**Moldeo o cast**

El resultado de una operación entre dos tipos de datos iguales puede dar como resultado

un tipo de dato diferente, en esos casos es necesario moldear el resultado. A este proceso

se le conoce como *cast*.

**Expresiones lógicas**

Las expresiones lógicas están constituidas por números, caracteres, constantes o variables

que están relacionados entre sí por operadores lógicos. Una expresión lógica puede tomar

únicamente los valores verdadero o falso.

Los operadores de relación permiten comparar elementos numéricos, alfanuméricos,

constantes o variables.

Los operadores lógicos permiten formular condiciones complejas a partir de condiciones

simples.

**Depuración de programas**

Cuando un programa falla (no termina su ejecución de manera correcta) y la información

enviada por el compilador es muy general, se puede ejecutar el programa en un contexto

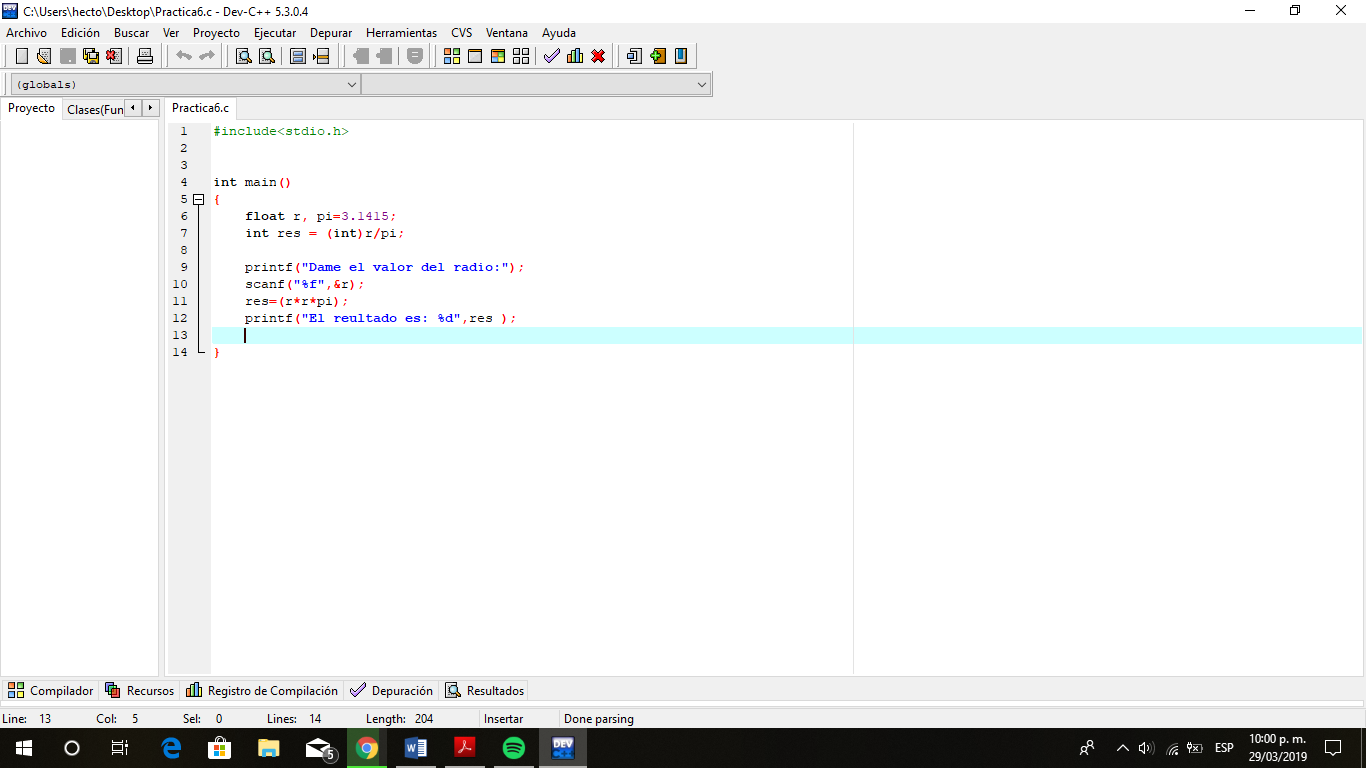
controlado para saber, exactamente, dónde está fallando. Se revisará este tema en la guía

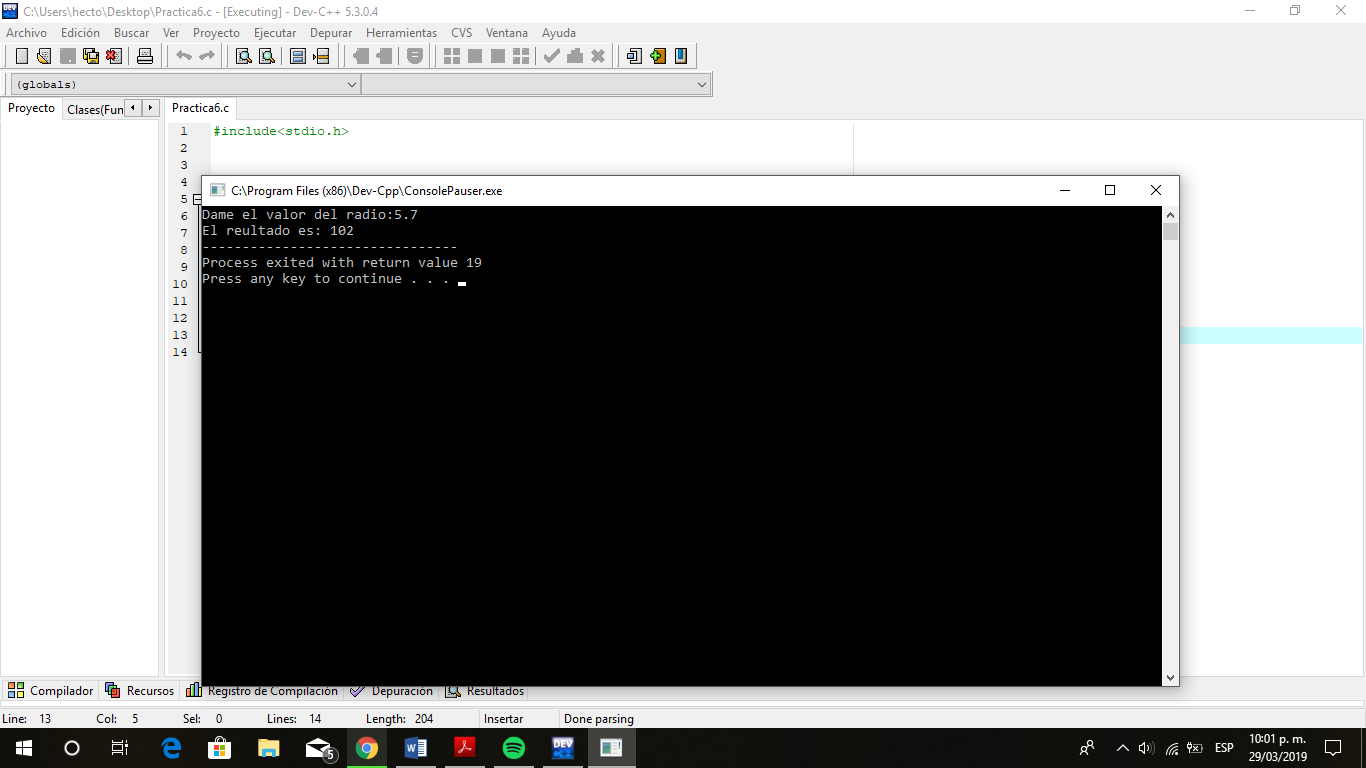
práctica de estudio “Depuración de programas” para conocer las diferentes herramientas

que nos ayudan a encontrar los errores de un programa.

**Actividades:**

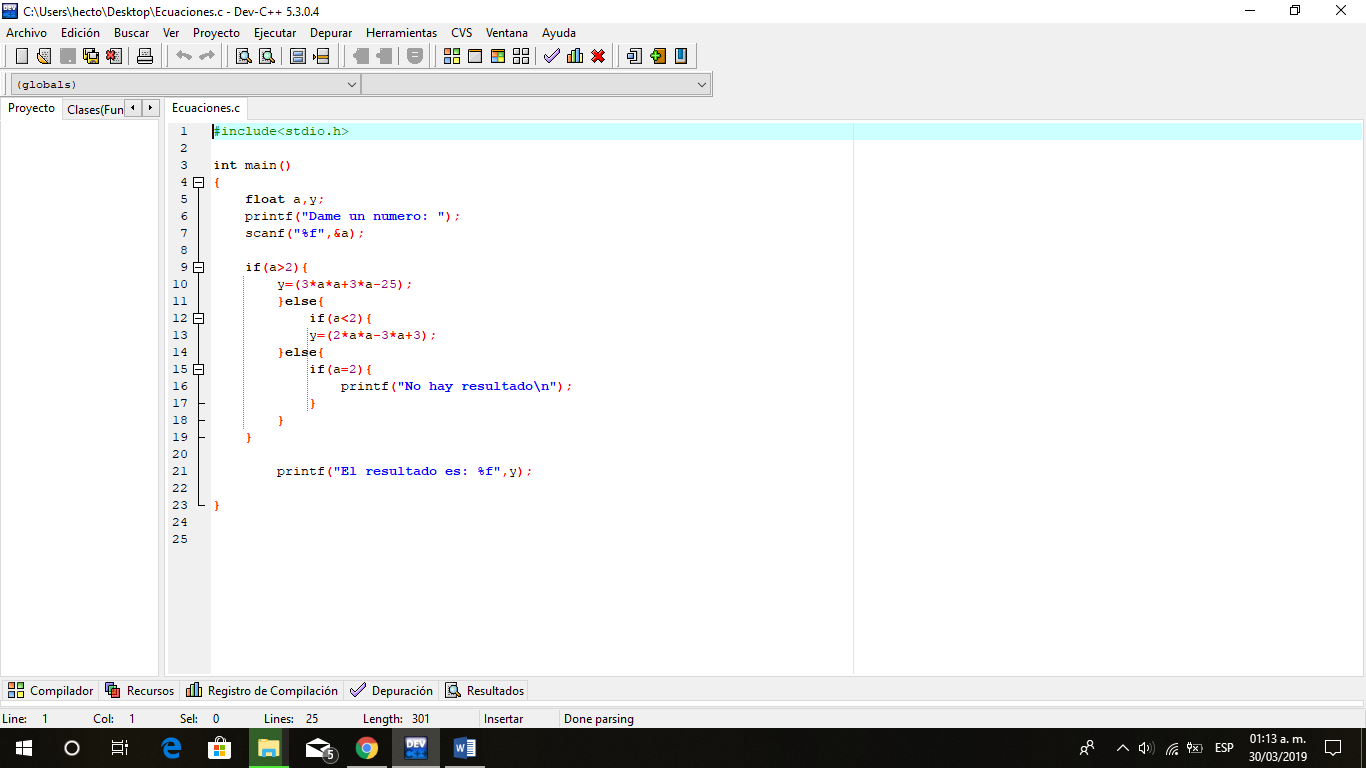
* Realizar un programa que calculi el area de un circulo utilizando el modelo cast

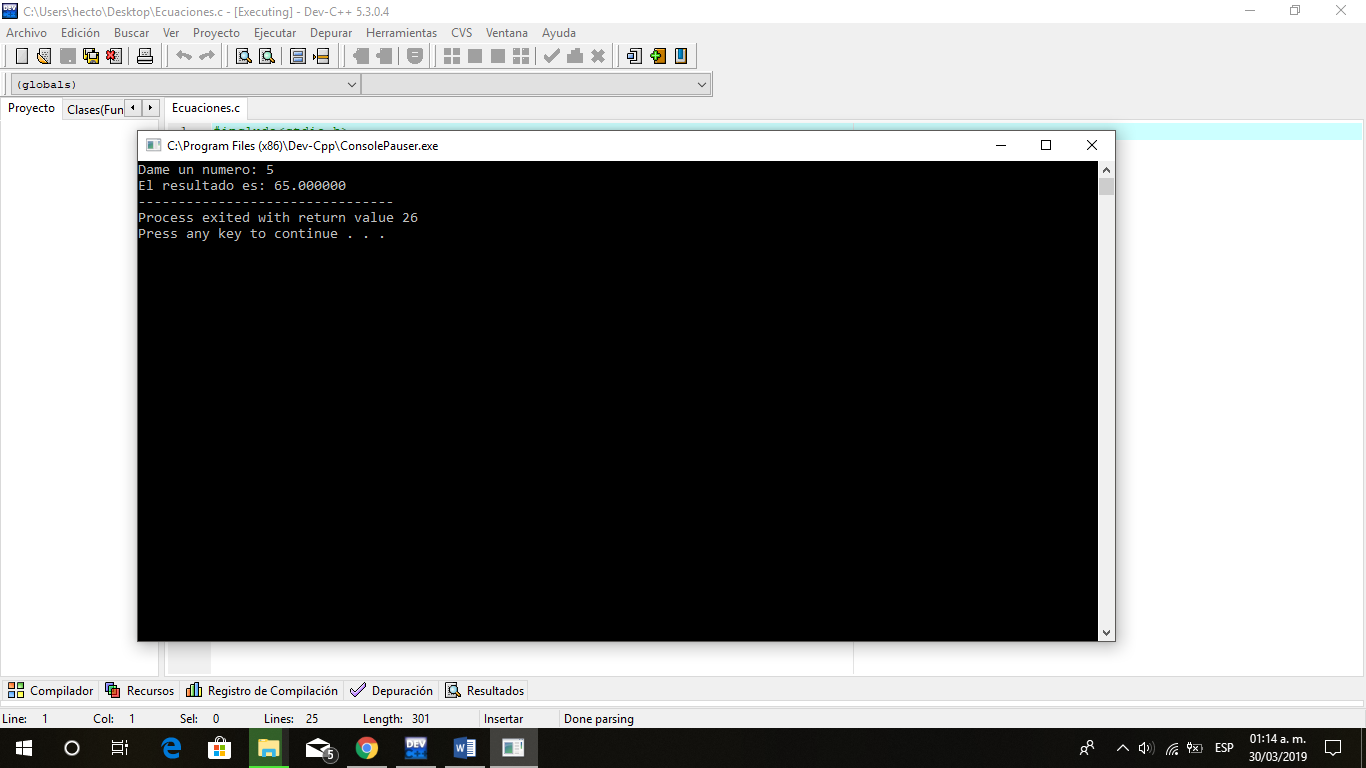


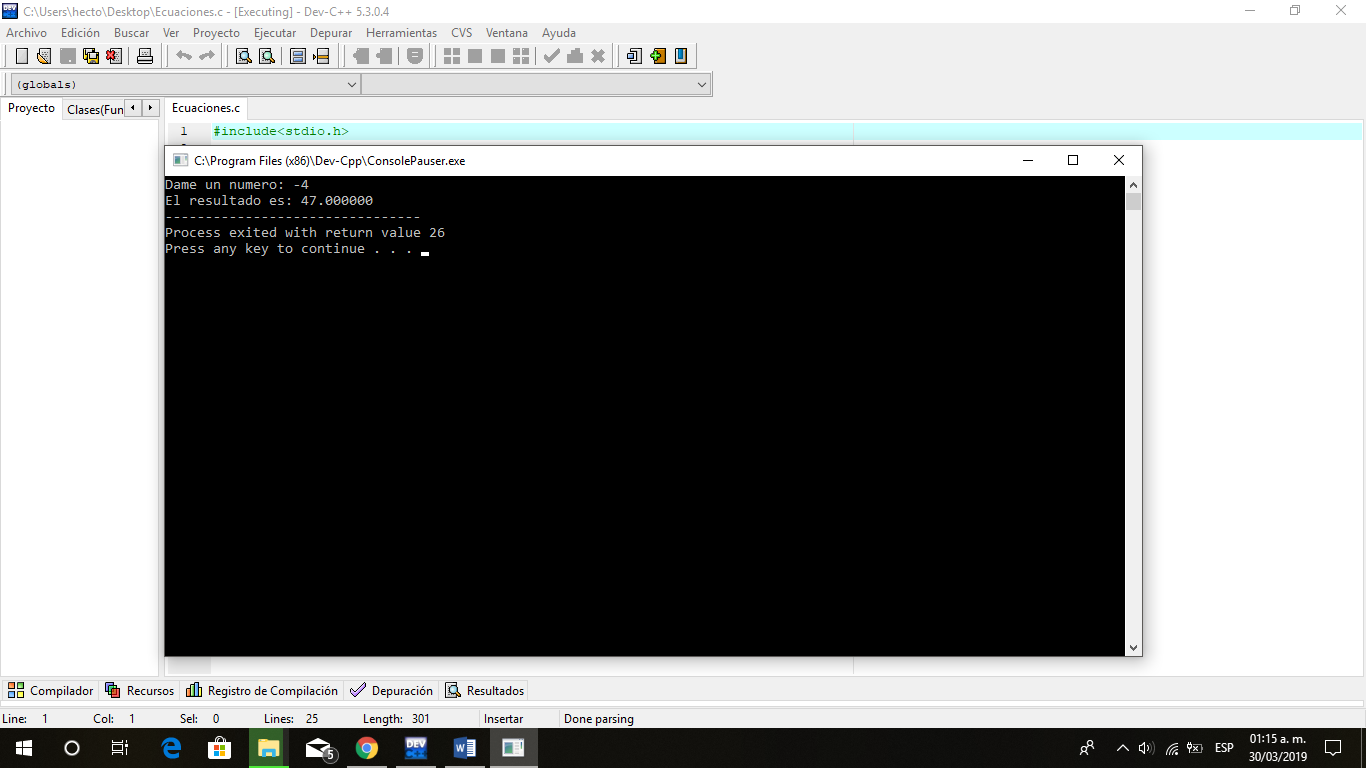


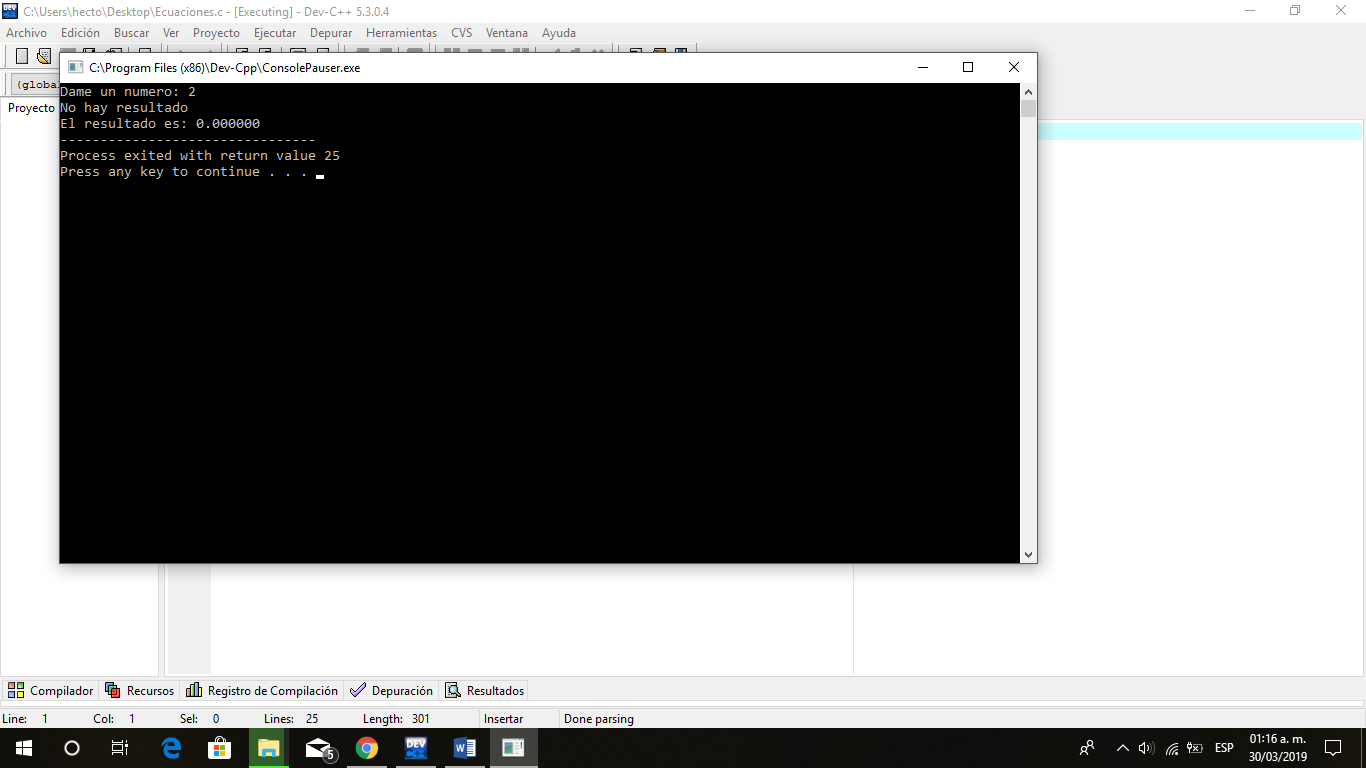
* Realizar un programa que el usuario introduzca un numero mayor a dos y resuelva la ecuacion: y=3x^2+3x-25.

Sí el numero introducido es menor a dos entonces que resuelva y=2x^2-3x+3

En caso de ser igual a dos no resolver nada.







**Conclusiones:**

El lenguaje se tiene una gran importancia hoy en dia aunque muchos ya lo consideran muy obsoleto por el simple hecho de que no se puede lograr aplicaciones grandes como un juego en 3d, pero si es un lenguaje de programacion muy basico para poder programar aplicaciones, utilerias, commandos y herramientas. Ademas que podemos decir que el lenguaje c a tenido sus influencias enotros lenguajes como c++, objective-c, javaScript, php, entr otros.