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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Rodriguez Espino Claudia |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | Práctica #8 |
| *Integrante(s):* | Diaz Flores Diego Sergei |
| *Semestre:* | Primer semestre |
| *Fecha de entrega:* | 12/11/2017 |
| *Obervaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Guía práctica de estudio 08: Estructuras de selección



***Elaborado por:***

M.C. Edgar E. García Cano

Ing. Jorge A. Solano Gálvez Ing.

***Revisado por:***

Laura Sandoval Montaño

***Autorizado por:***

M.C. Alejandro Velázquez Mena

**Guía práctica de estudio 08: Estructuras de selección**

Objetivo:

Elaborar programas en lenguaje C que incluyan las estructuras de selección if, if-else, switch y ternaria (o condicional) para la resolución de problemas básicos.

Actividades:

• Elaborar expresiones lógicas/condicionales utilizadas en las estructuras de selección y realizar su evaluación.

• Elaborar un programa en lenguaje C para cada estructura de selección.

En las 2 practicas anteriores de laboratorio se ha trabajado con el lenguaje de programacion c, aprendimos funciones básicas como por ejemplo utilizar librerías, declarar variables y constantes, imprimir en pantalla, pedir datos de entrada, realizar algunas operaciones, etcétera.

Sin embargo en esta práctica aprendimos a utilizar más y mejores herramientas que pone a nuestra disposición este lenguaje de programacion. A mi parecer las más importantes son las estructuras de selección (o también llamadas condicionales) ya que estas permiten realizar una u otra acción con base en una expresión lógica. Además de que siguen un ciclo y nos facilitan la programacion ya que no es necesario escribir repetitivamente las acciones a realizar.

La primera estructura de selección tiene por nombre “if”

Su sintaxis es: if (expresión\_ lógica) { // bloque de código a ejecutar }, esta estructura se evalúa la expresión lógica y, si se cumple (si la condición es verdadera), se ejecutan las instrucciones del bloque que se encuentra entre las llaves de la estructura. Si no se cumple la condición, se continúa con el flujo normal del programa. Además if a mi parecer es una herramienta muy útil debido a que es muy fácil de utilizar.

La segunda estructura es una derivación o complementación de la anterior lleva por nombre if-else su sintaxis es la siguiente

if (expresión\_ lógica) { // bloque de código a ejecutar // si la condición es verdadera }

else { // bloque de código a ejecutar // si la condición es falsa }, igualmente que la estructura if, primero se evalúa la expresión lógica que se encuentra entre las llaves de la estructura (en caso de ser verdadero, pero el complemento de esta es que si la condición es falsa se ejecutaran las condiciones que se encuentran dentro de las llaves de else; para finalmente continuar con el flujo normal del programa.

La tercera estructura de selección se llama “switch” y su sintaxis es la siguiente

switch (opción\_ a\_ evaluar){

case valor1: /\* Código a ejecutar\*/ break;

case valor2: /\* Código a ejecutar\*/ break; …

case valorN: /\* Código a ejecutar\*/ break;

default: /\* Código a ejecutar\*/ }

Esta estructura evalúa la variable que se encuentra entre paréntesis después de su palabra reservada, los tipos de datos que puede evaluar son en enteros, caracteres y enumeraciones, en las clases de teoría y en la practica hemos aprendido que es este tipo de estructura siempre es necesario colocar el break ya que si no se coloca el programa correrá y ejecutara todos los casos.

Además este tipo de estructura permite al usuario darse cuenta que introduce datos erróneos ya que el default sirve para todos los casos no registrados y con el cual enviara un mensaje para la reintroducción de datos.

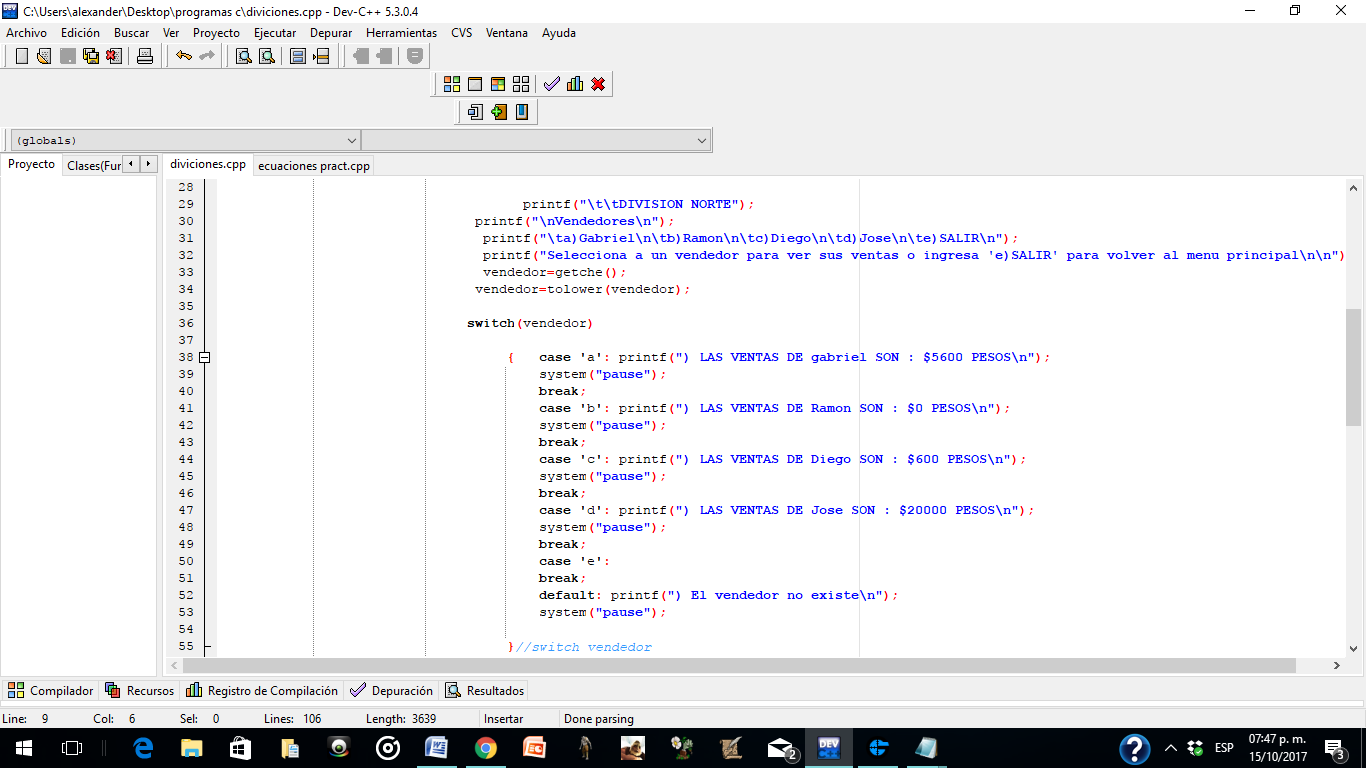
La cuarta estructura de selección que aprendimos en esta práctica fue la estructura ternaria

La estructura condicional (también llamado operador ternario) permite realizar una comparación rápida. Su sintaxis es la siguiente:

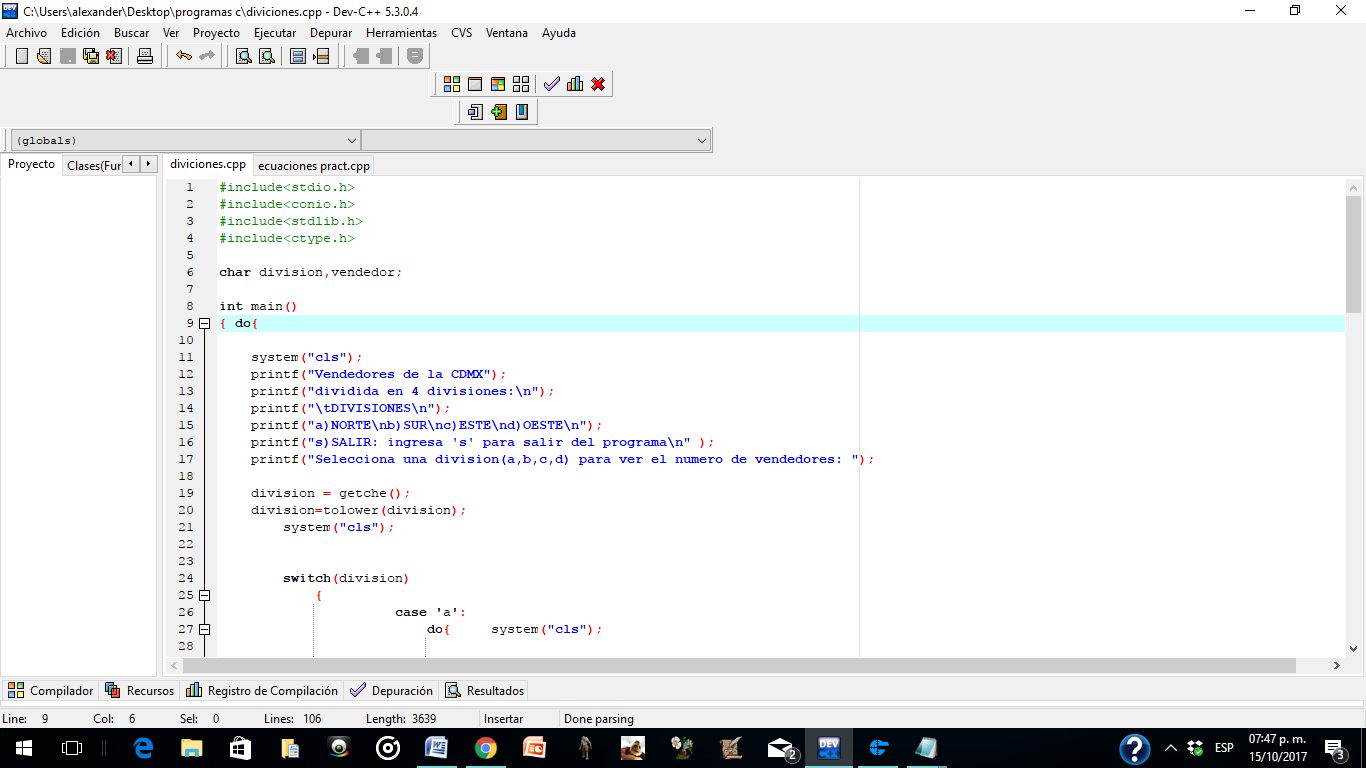
Condición? Si Se Cumple: Si No Se Cumple

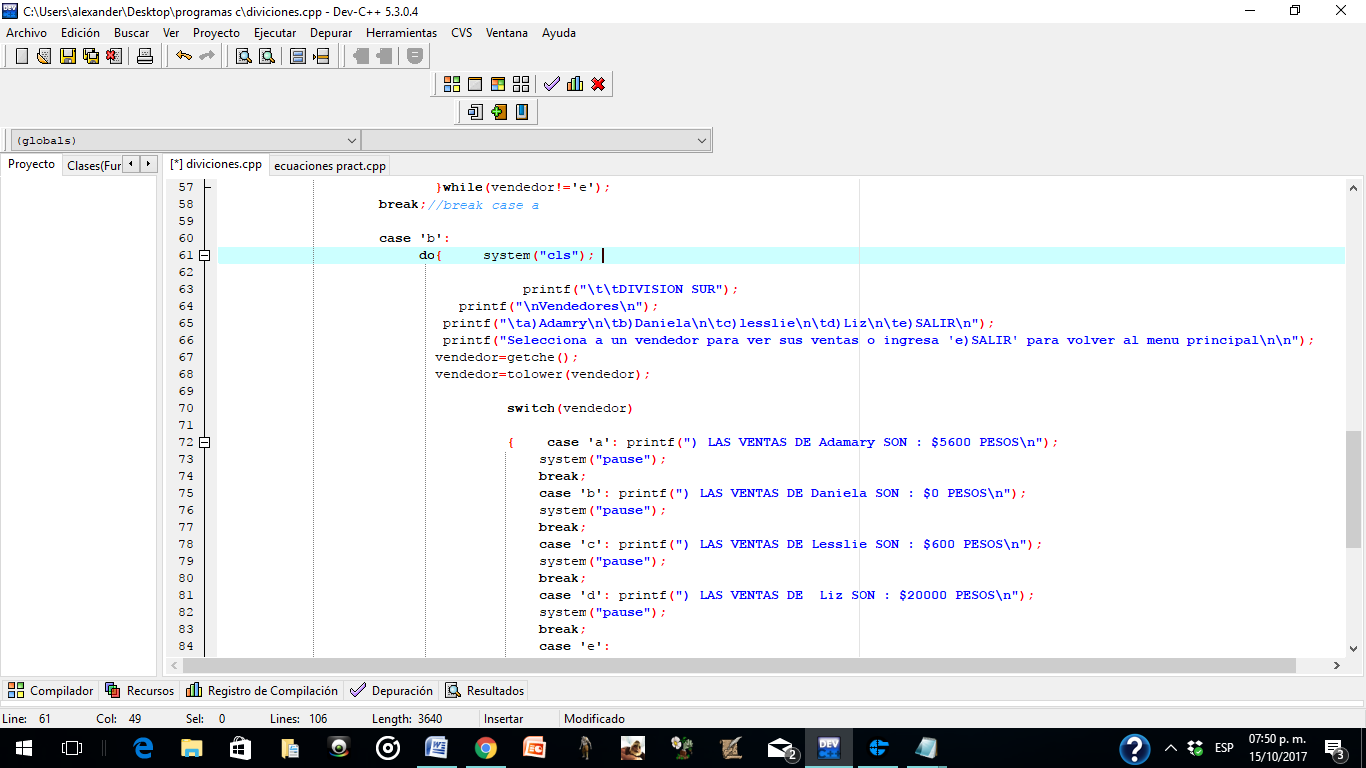
Esta estructura es muy sencilla simplemente si la condición propuesta es verdadera se ejecuta todo lo que esta antes del signo y en caso de ser falsa se ejecuta todo lo que esta después del signo, una vez ejecutado un de los dos casos continua con las líneas del programa principal.

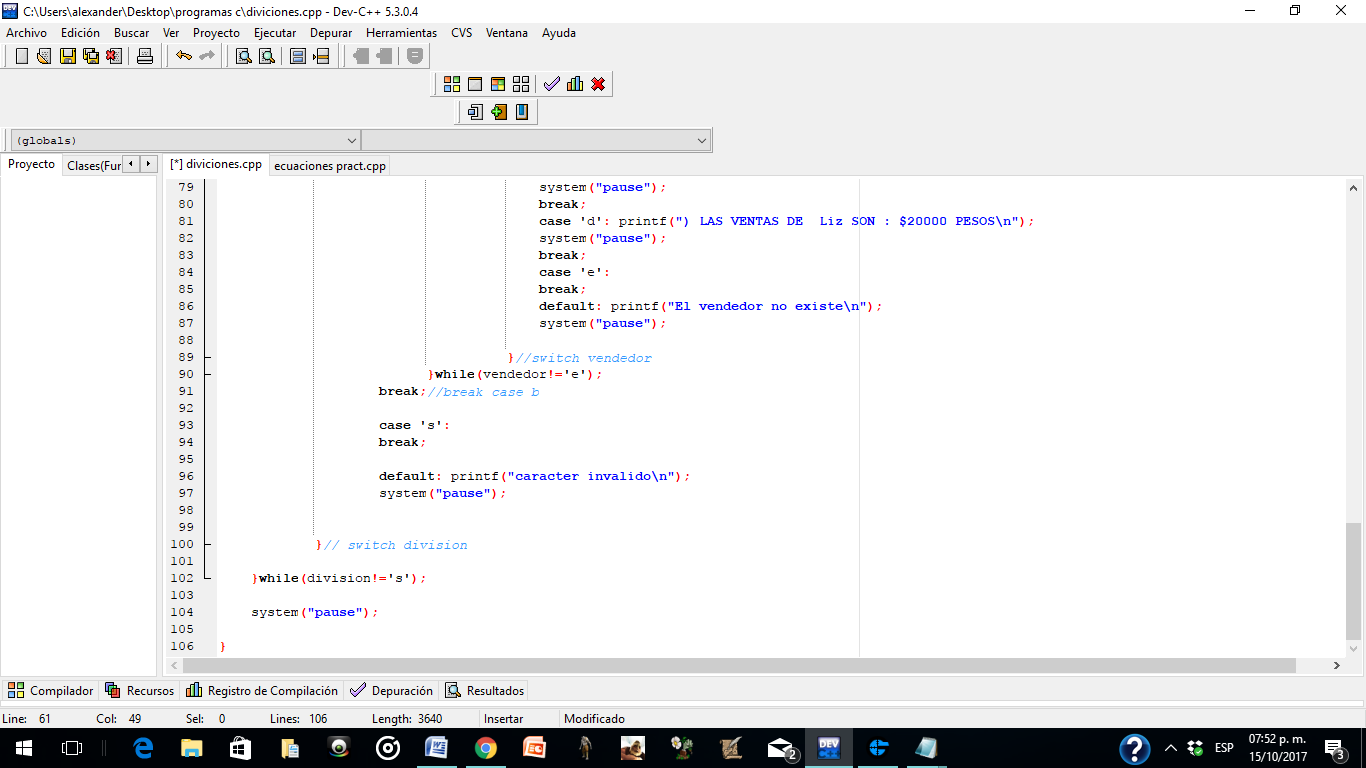
Las estructuras de selección como toda la programacion que hemos aprendido en las clases de teoría como en las practicas necesita únicamente utilizacion de librerías, declaración de variables y constantes dependiendo del programa a desarrollar, y colocar dichas estructuras dentro de la función main().

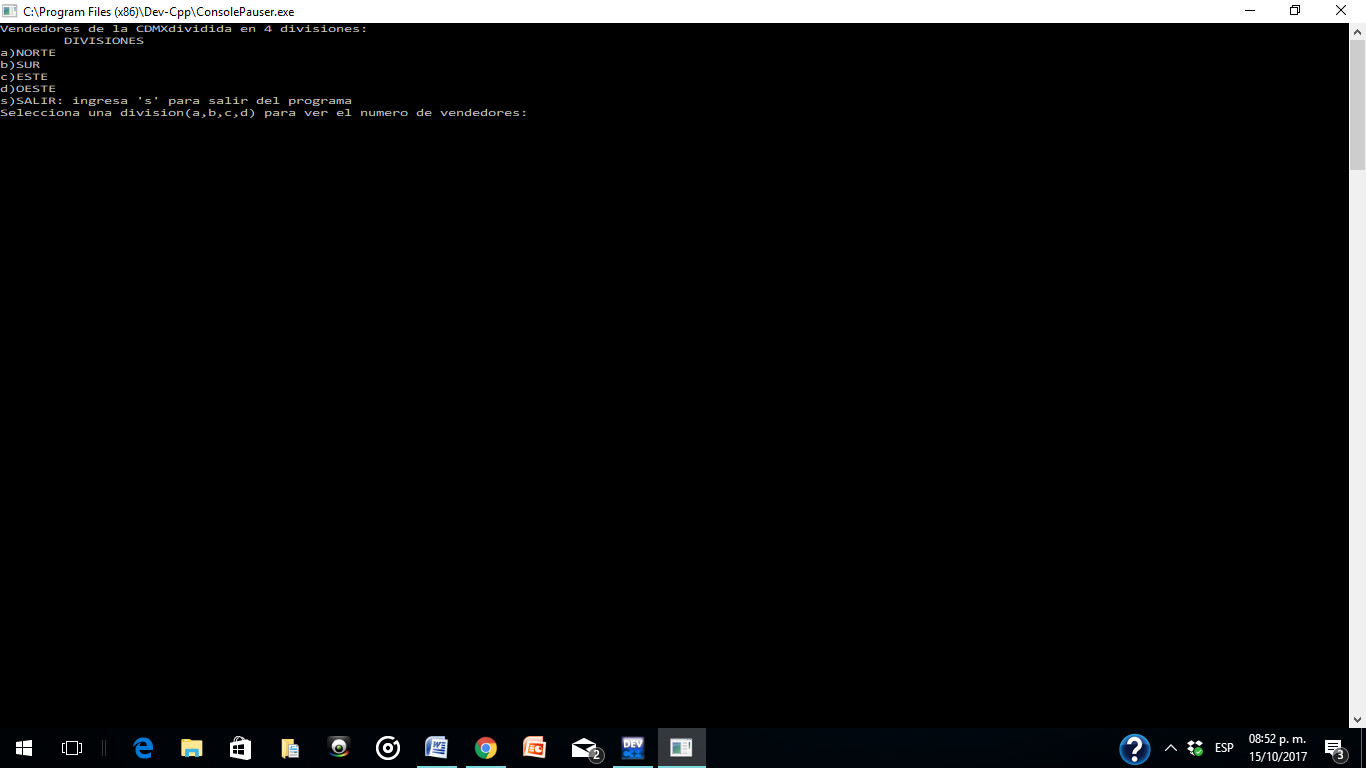
Programa numero 1.-

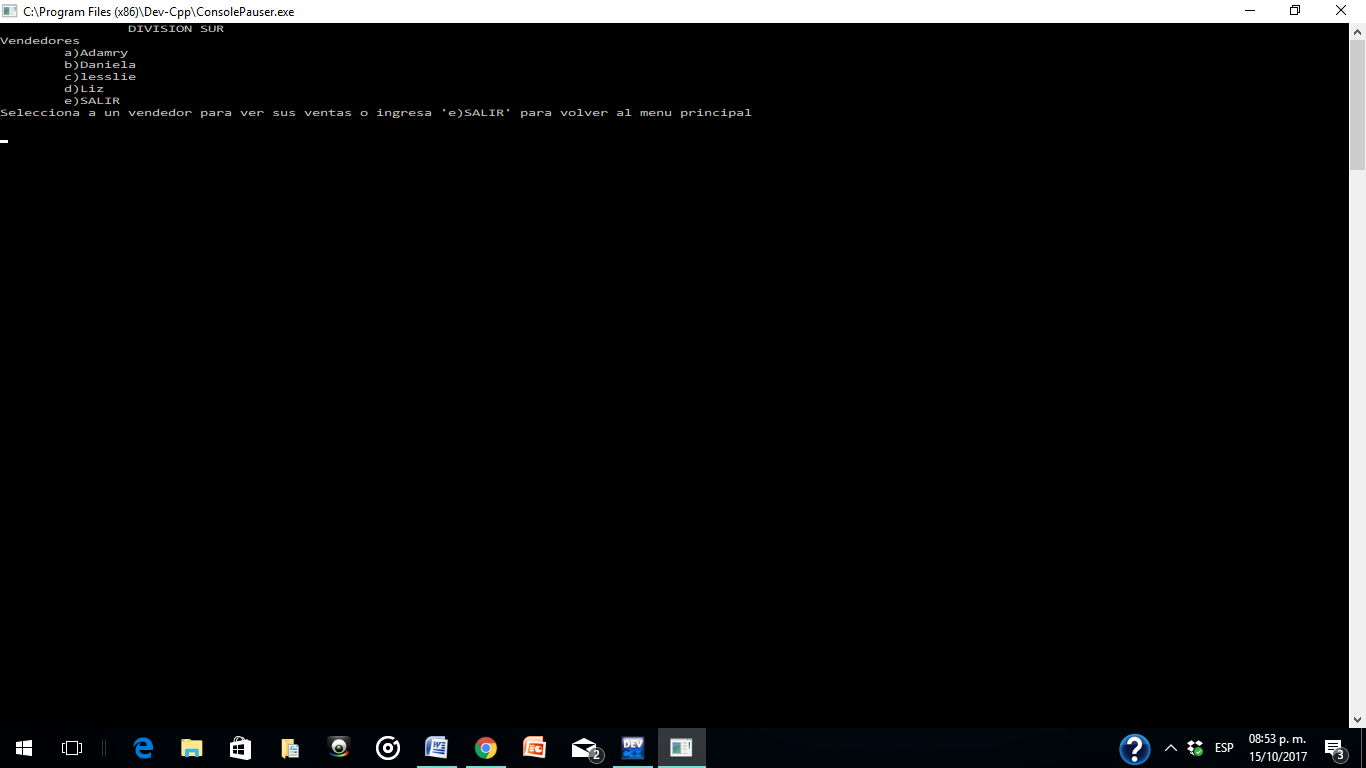
Utilizacion de la estructura de selección switch, y utilizacion de caracteres (letras a, b, c, d).

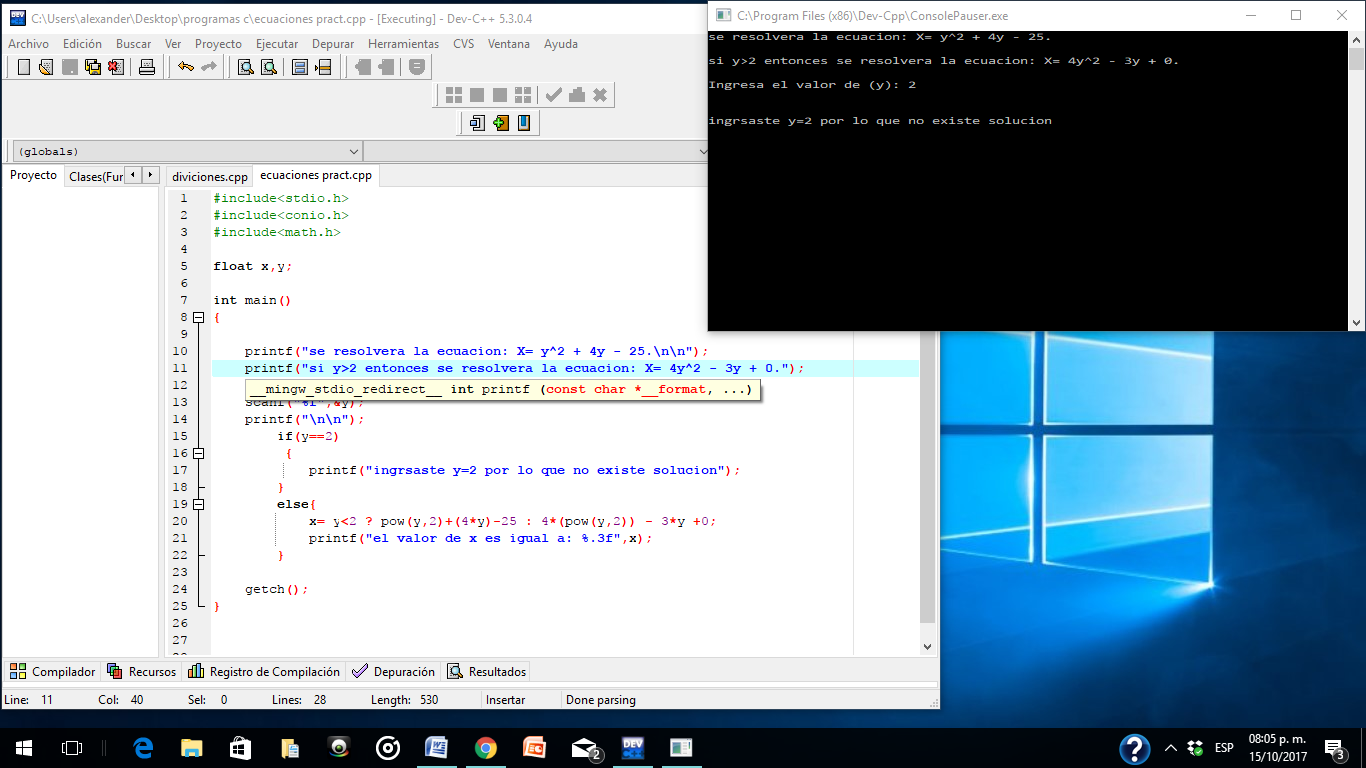
****



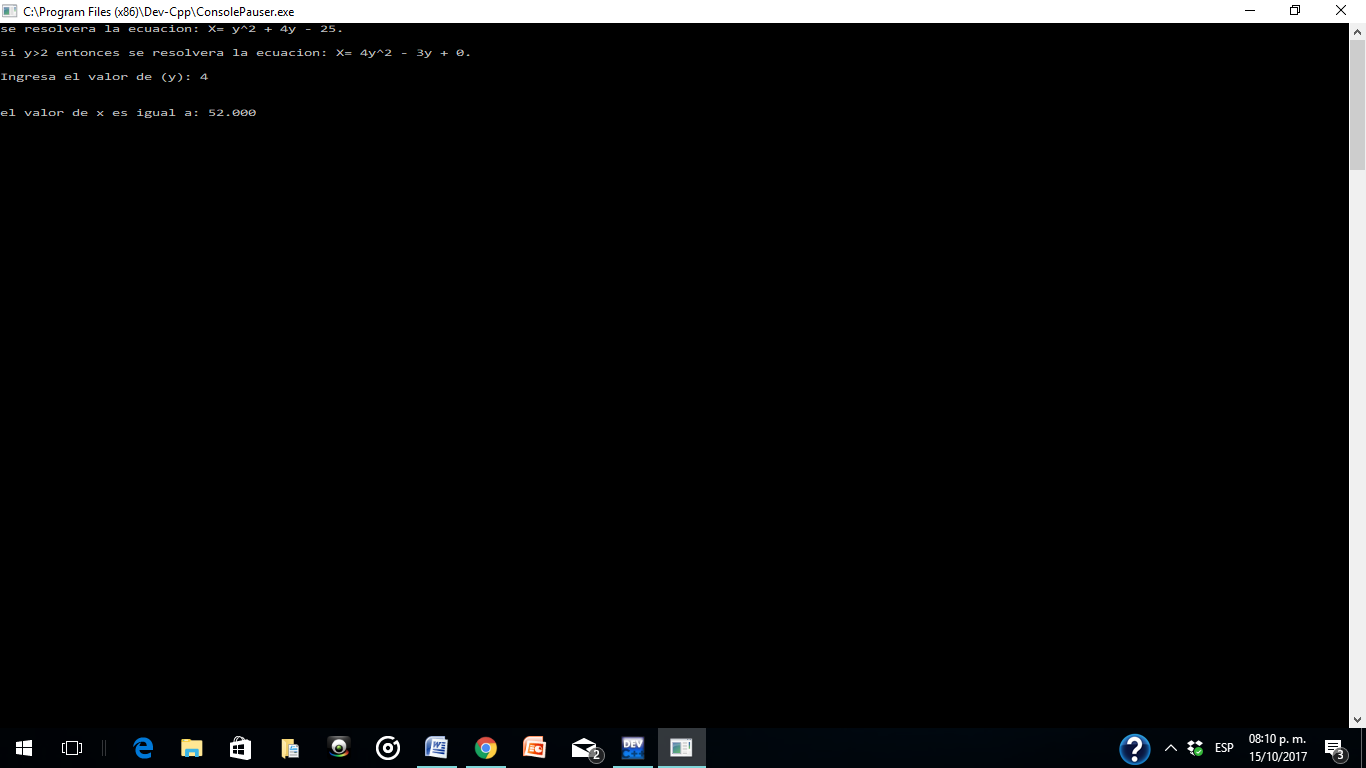






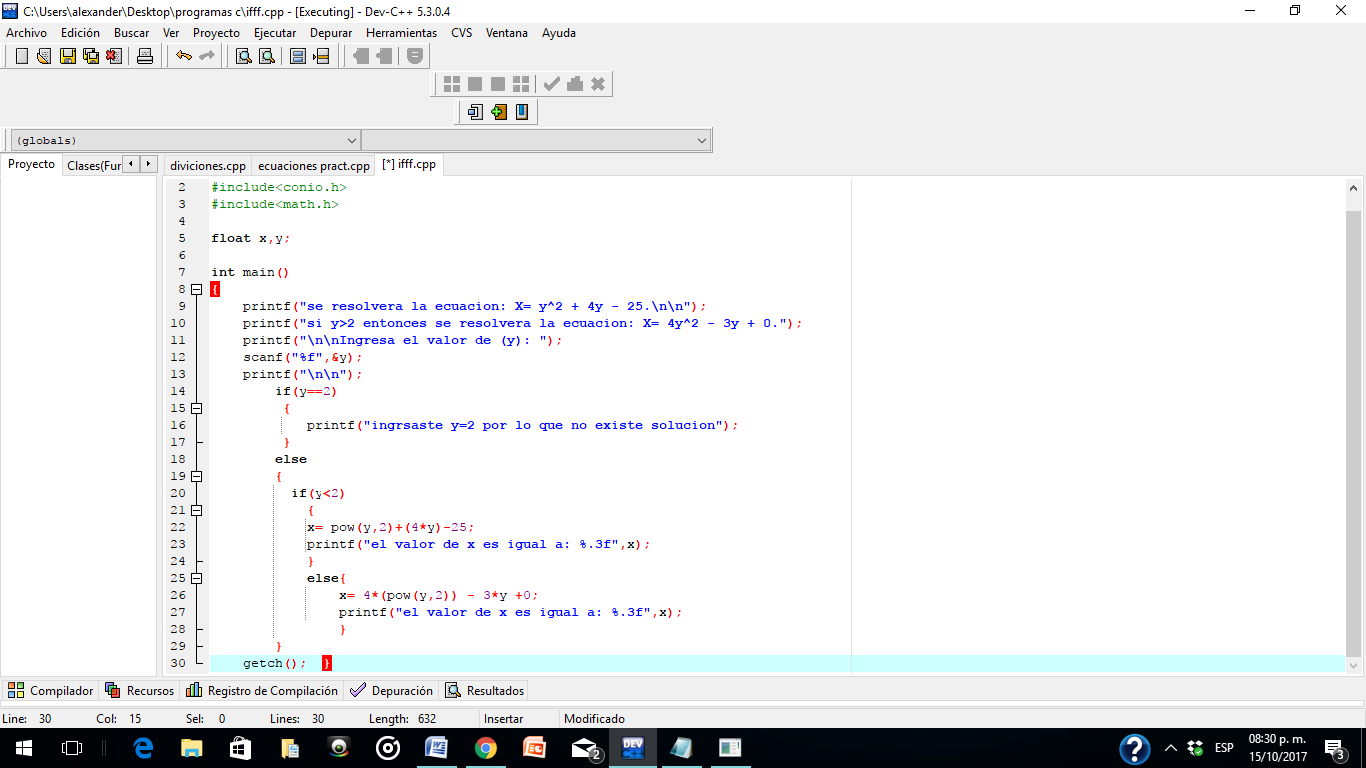
Programa numero 2.-

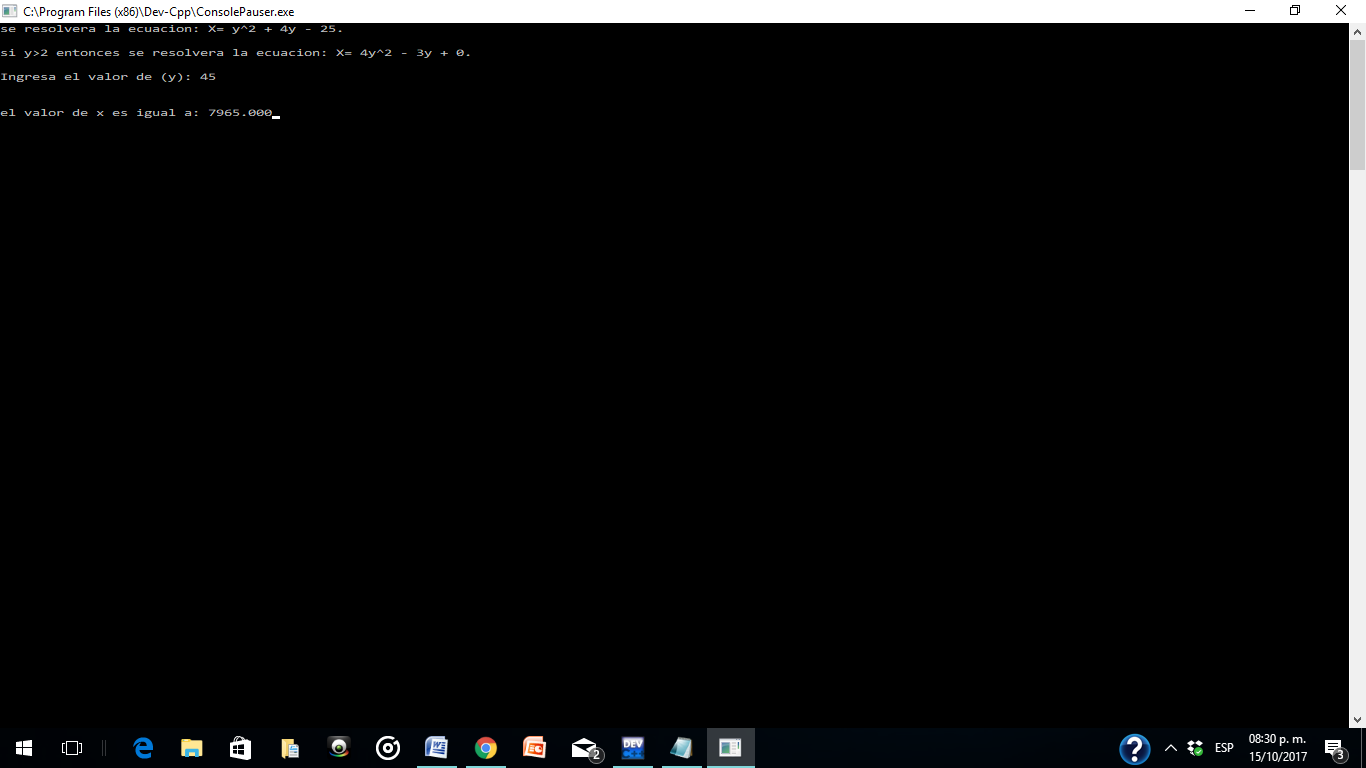
Resolución de la ecuación x= y^2+ 4y -25 +4y -25 y utilizacion del operador ternario.



Programa numero 3.-

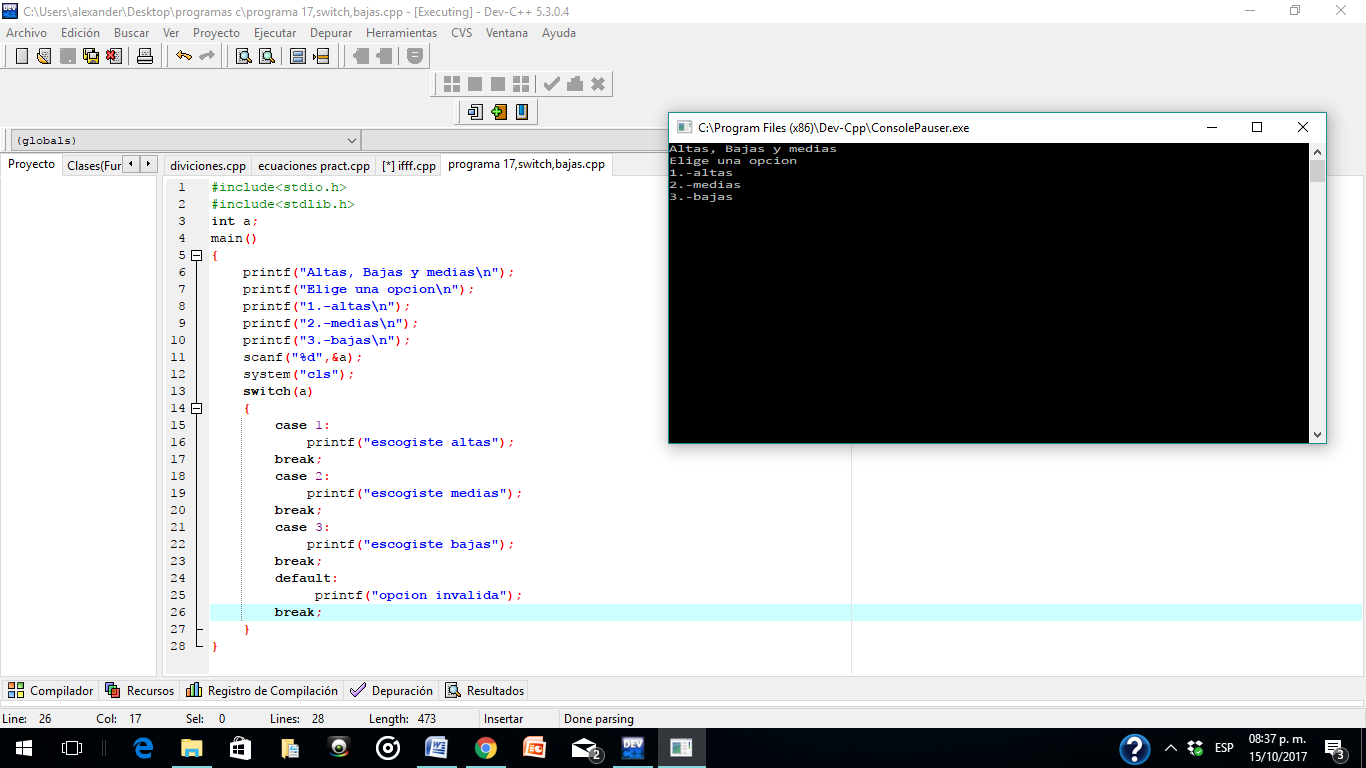
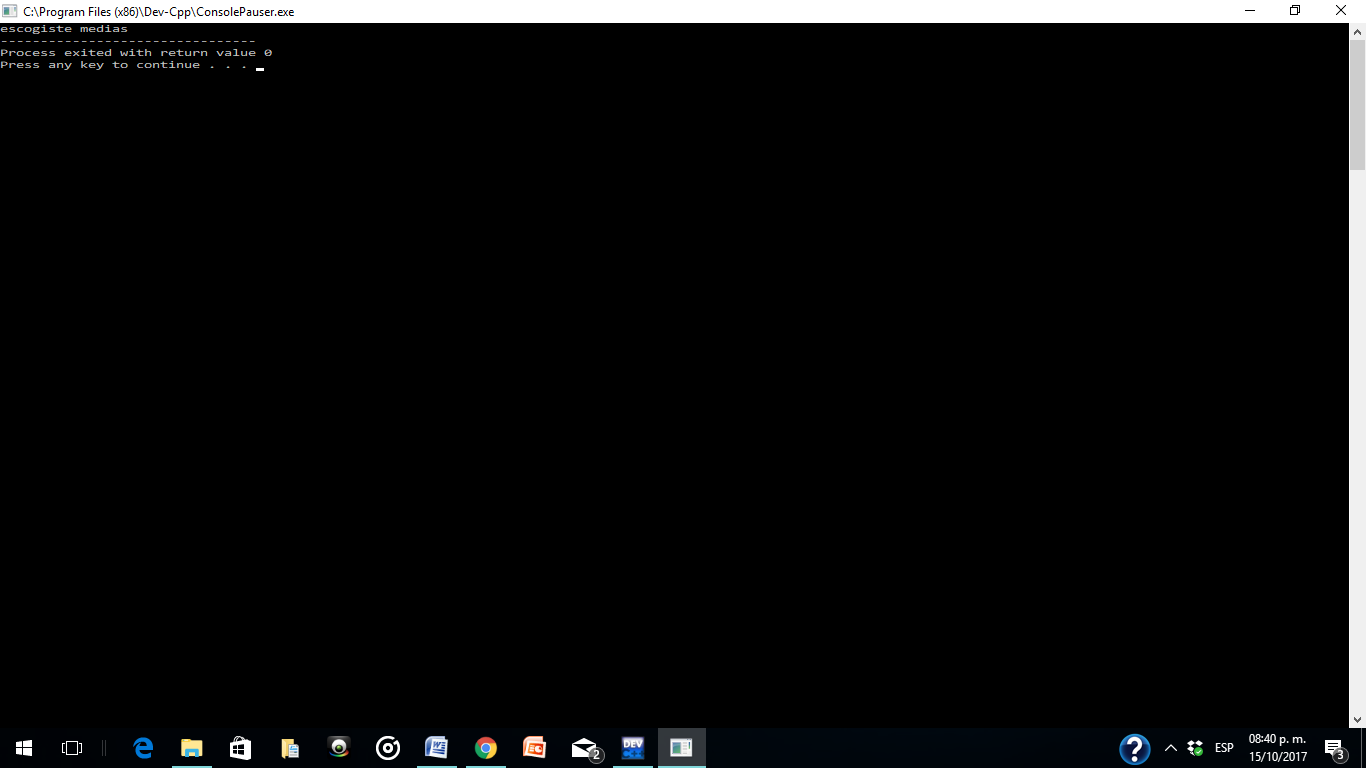
Resolución de la ecuación x= y^2+ 4y -25 +4y -25, utilizando la estructura de selección if-else.





Programa numero 4.-

Utilizacion de la estructura de selección switch.



Al concluir podemos decir que las estructuras de selección tienen una función muy importante dentro de la programacion, ya que a partir de ellas es posible evaluar una, dos o más condiciones simultáneamente lo que nos facilita el trabajo y tener una mejor presentación en nuestro código fuente. Además estas nos permiten crear un mayor número de programas ya que la utilizacion de las estructuras se pueden combinar o anidar.