|  |
| --- |
| Laboratorio de Computación  Salas A y B |
| |  |  | | --- | --- | | Profesor: | Claudia Rodríguez Espino | | Asignatura: | Fundamentos de Programación | | Grupo: | 1104 | | No de Práctica(s): | 8 | | Integrante(s): | Téllez Torres Lorena Alejandra | |  |  | |  |  | |  |  | | No. de Equipo de cómputo empleado: | 51 | | Semestre: | 1ª | | Fecha de entrega: |  | | Observaciones: |  | |  |  | |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**OBJETIVO**

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

**DESARROLLO**

Las estructuras de control de flujo en un lenguaje especifican el orden en que se realiza el procesamiento de datos.

Las estructuras de selección (o condicionales) permiten realizar una u otra acción con base en una expresión lógica. Las acciones posibles a realizar son mutuamente excluyentes, es decir, solo se puede ejecutar una a la vez dentro de toda la estructura.

Lenguaje C posee 3 estructuras de selección: la estructura if-else, la estructura switch y la estructura condicional o ternaria.

**Estructura de control selectiva if**

La estructura de control de flujo más simple es la estructura condicional if, su sintaxis es la siguiente:

*if (expresión\_lógica) {*

*// bloque de código a ejecutar*

*}*

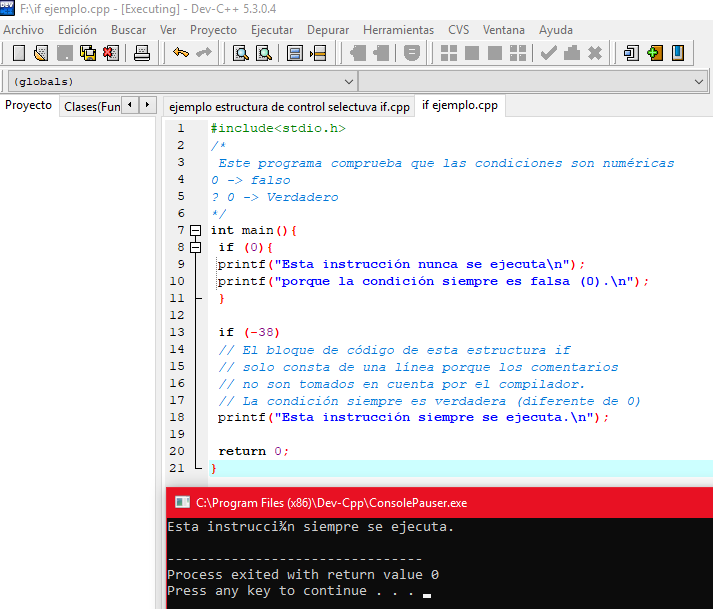
En esta estructura se evalúa la expresión lógica y, si se cumple (si la condición es

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.

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.



**Estructura de control selectiva if-else**

La sintaxis de la estructura de control de flujo if-else es la siguiente:

*if (expresión\_lógica) {*

*// bloque de código a ejecutar*

*// si la condición es verdadera*

*} else {*

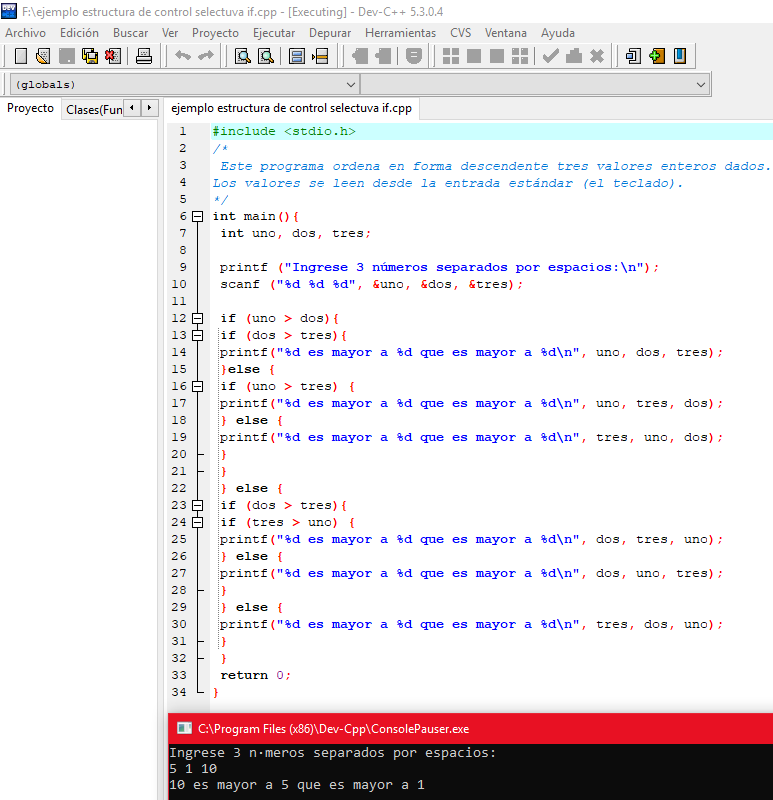
*// bloque de código a ejecutar*

*// si la condición es falsa*

*}*

Esta estructura evalúa la expresión lógica y si la condición es verdadera se ejecutan las instrucciones del bloque que se encuentra entre las primeras llaves, si la condición es falsa se ejecuta el bloque de código que está entre las llaves después de la palabra reservada 'else'. Al final de que se ejecute uno u otro código, se continúa con el flujo normal del programa.

Es posible anidar varias estructuras if-else, es decir, dentro de una estructura if-else tener una o varias estructuras if-else.



**Estructura de control selectiva switch-case**

La sintaxis de la estructura switch-case es la siguiente:

*switch (opcion\_a\_evaluar){*

*case valor1:*

*/\* Código a ejecutar\*/*

*break;*

*case valor2:*

*/\* Código a ejecutar\*/*

*break;*

*…*

*case valorN:*

*/\* Código a ejecutar\*/*

*break;*

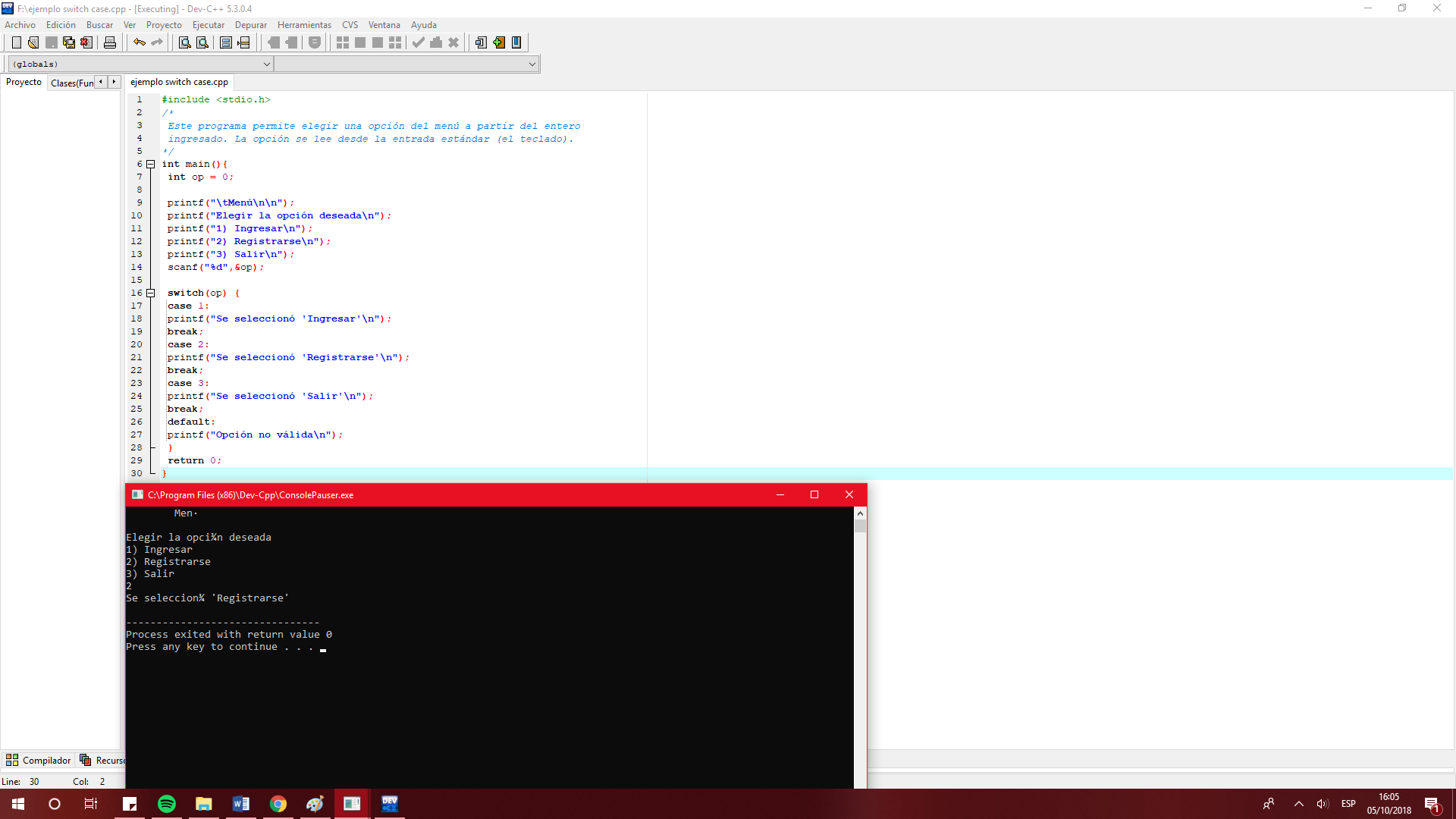
*default:*

*/\* Código a ejecutar\*/*

*}*

La estructura switch-case evalúa la variable que se encuentra entre paréntesis después de la palabra reservada switch y la compara con los valores constantes que posee cada caso (case). Los tipos de datos que puede evaluar esta estructura son enteros, caracteres y enumeraciones. Al final de cada caso se ejecuta la instrucción break, si se omite esta palabra reservada se ejecutaría el siguiente caso, es decir, se utiliza para indicar que el bloque de código a ejecutar ya terminó y poder así salir de la estructura.

Si la opción a evaluar no coincide dentro de algún caso, entonces se ejecuta el bloque por defecto (default).



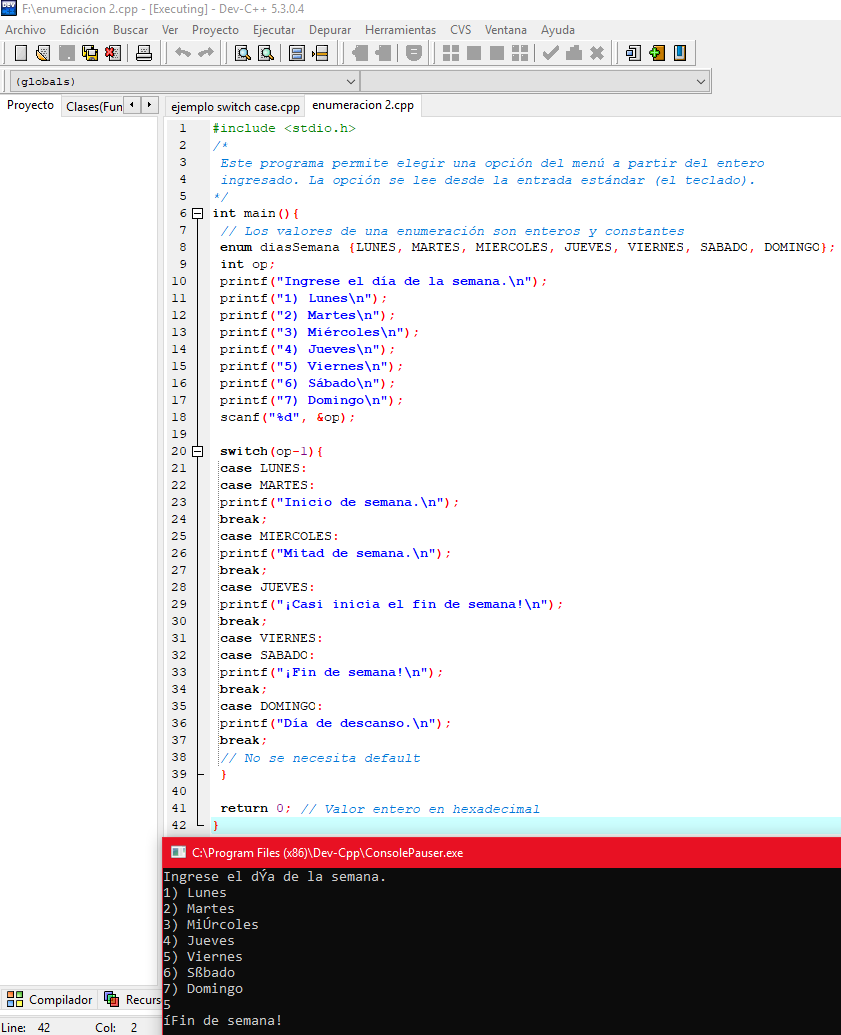
**Enumeración**

Existe otro tipo de dato constante conocido como enumeración. Una variable enumerador se puede crear de la siguiente manera:

*enum identificador {VALOR1, VALOR2, ... , VALORN};*

Para crear una enumeración se utiliza la palabra reservada *enum,* seguida de un

identificador (nombre) y, entre llaves se ingresan los nombres de los valores que puede tomar dicha enumeración, separando los valores por coma. Los valores son elementos enteros y constantes (por lo tanto, se escriben con mayúsculas).

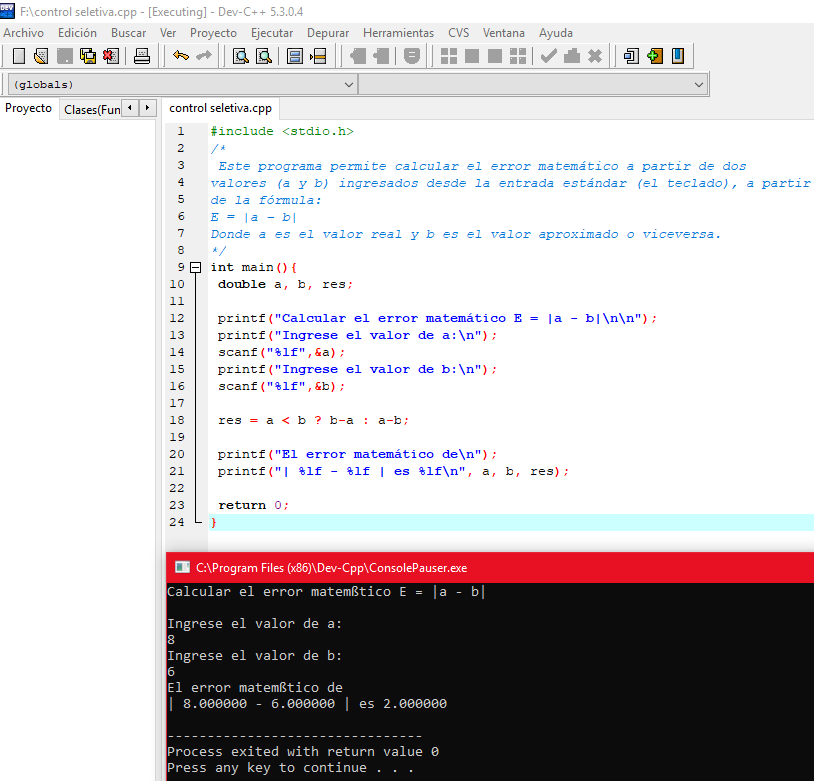


**Estructura de control selectiva condicional**

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

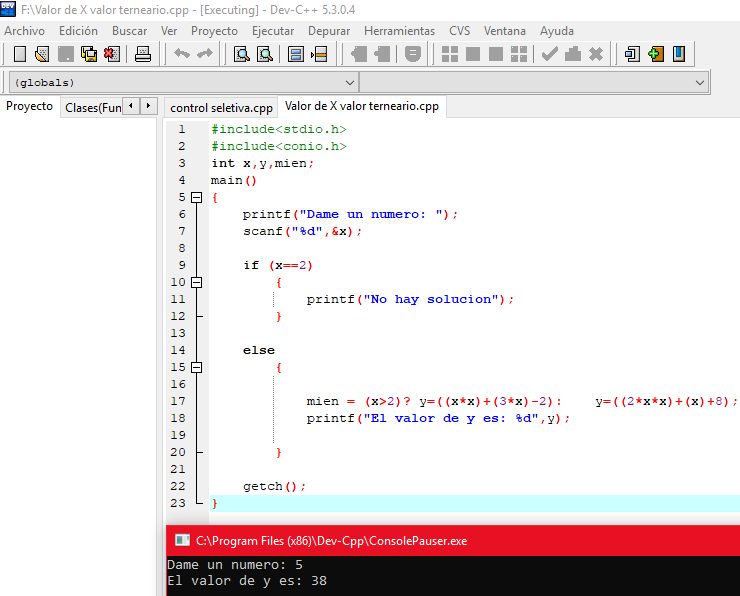
*Condición ? SiSeCumple : SiNoSeCumple*

Consta de tres partes, una condición y dos acciones a seguir con base en la expresión condicional. Si la condición se cumple (es verdadera) se ejecuta la instrucción que se encuentra después del símbolo ‘?’; si la condición no se cumple (es falsa) se ejecuta la instrucción que se encuentra después del símbolo *‘:’.*

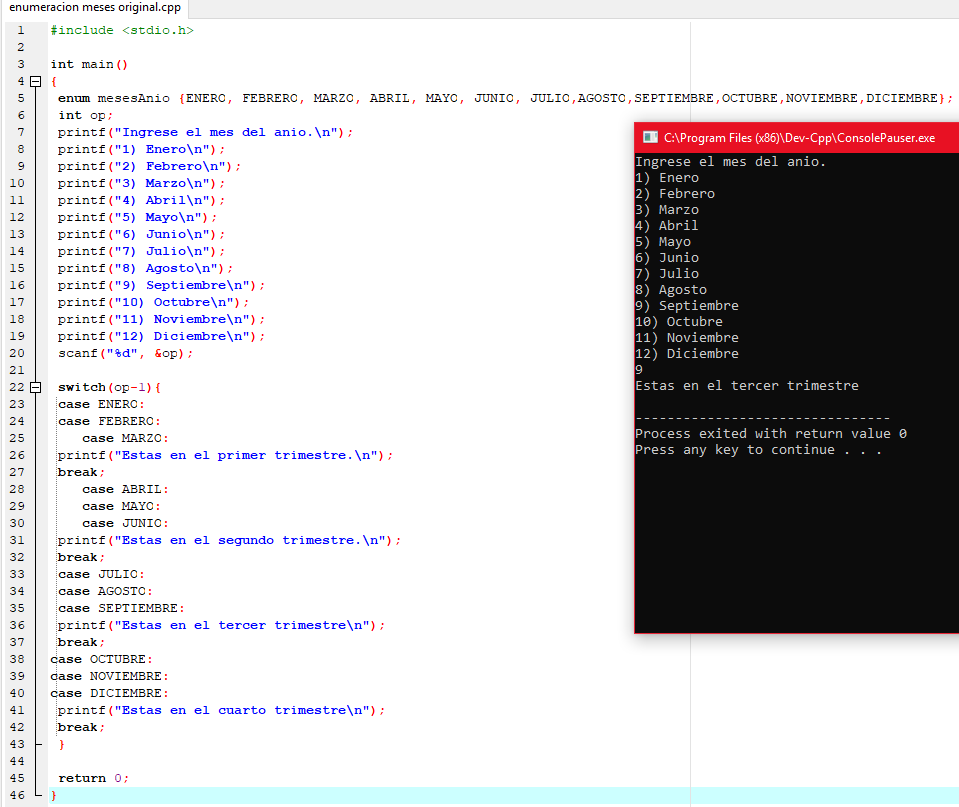


**Ejercicios:**

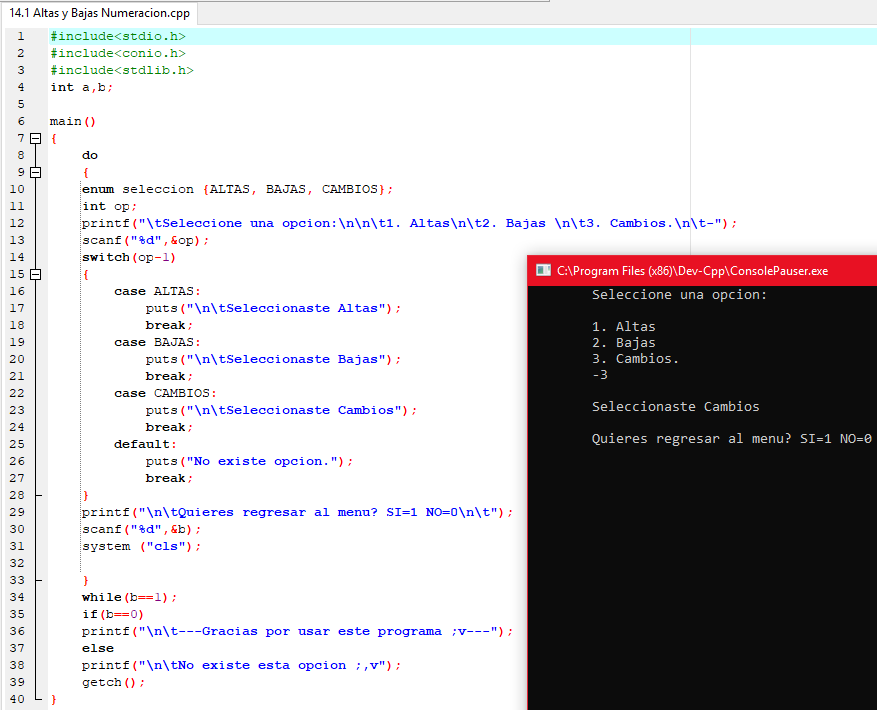
1. **Valor de X**

****

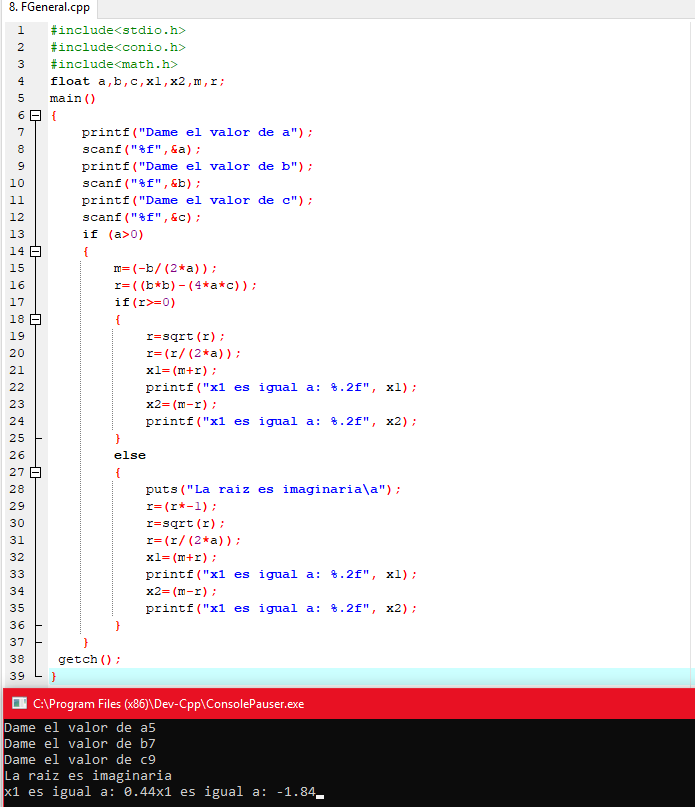
1. **Enumeración**



1. **Altas, Bajas y Cambios**

****

1. **Fórmula General**

****

**CONCLUSIONES**

Me pareció interesante la práctica ya que no conocía dos cosas que se utilizaron: la enumeración y la estructura de control selectiva condicional. Fue más fácil dado que pude agarrar programas que ya tenía y modificarlos a las condiciones que se pedían. En lo personal pienso que entre más cosas conozcas dentro del mundo de la programación mejor ingeniero en computación serás, así que me parece muy importante aprender más y más del lenguaje de C.