|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Manual de practicas** | | | **No.** | **1** |
| **Asignatura:** | **Métodos Numéricos.** | **Carrera:** | **Ing. Sistemas Computacionales** | **Duración de la práctica (Hrs)** |  |
| **Nombre :** | **Verónica Cruz Osorio** | | **Grupo :** | **4032** | |

**I. Competencia(s) específica(s):**

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

**III. Material empleado:**

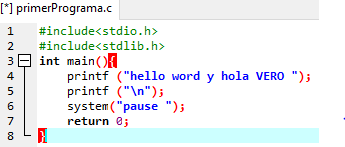
Computadora.

Dev C++

**IV. Desarrollo de la práctica:**

**Primer programa:** Lleva por nombre **hola mundo y hola vero**

Son Bibliotecas estándar para datos de entrada y salida

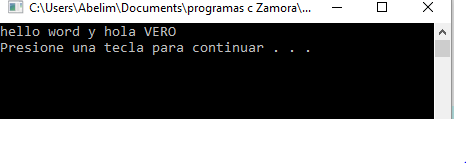


**\n** Salto de línea

Sirve para imprimir en pantalla una cadena de caracteres con formato.



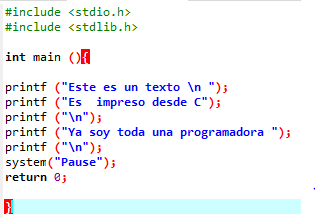
Esta opción nos permite guardar y nombrar nuestro programa.

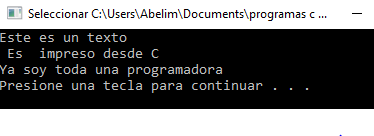


La declaración de las funciones de la biblioteca estándar, para hacer operaciones, entrada o salid.

Standard library, archivo de cabecera de la biblioteca estándar de propósito general- prototipo de funciones para gestionar la memoria dinámica, control de procesos.

**Segundo programa**



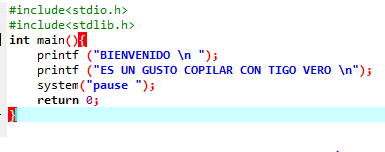


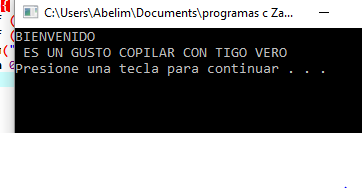
Printf nos permite la impresión de salida de datos y que se muestre en pantalla una cadena de caracteres con formato.

\n Salto de línea

**Tercer Programa: Crea un nuevo programa que imprima una bienvenida con tu nombre.**

**printf** es para imprimir en el mensaje (“Bienvenido”) y se usa **\n** un salto de línea para que no se coloque otro mensaje frete al primero y después se colocó otro para otro mensaje y es lo que va a mostrar en pantalla.



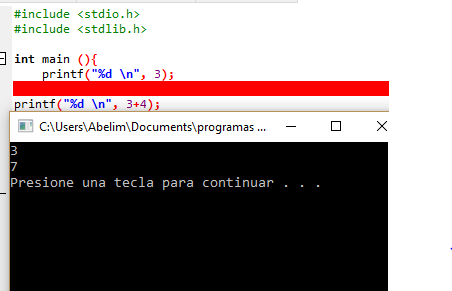


Declaración de las bibliotecas

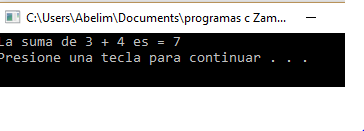
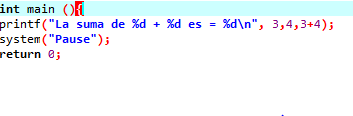
**Cuarto programa**

El programa requiere que se imprima el número **3** y en l aparte siguiente se imprima la suma de dos valores que ya estas determinados. 3+4 que nos debe de mostrar **7.**

Se utilizó **%d**  que permite que se tome el valor 3 y se pueda imprimir.



Al igual que el anterior programa se va a imprimir la **suma de 3+4,** donde **%d +%d** y se deben de colocar en el orden como se debe de sumar 3, 4 y después **3+4** para que en el momento de imprimir se muestre el mensaje que se encuentra en “” y el resultado



Se utilizó printf para imprimir los valores y %d para que se ordene en el orden que vas a desear.

Utilizamos un salto de línea par que los resultados no se muestren amontonados.

**%d+%d= 5+8,5+8** al igual con la siguiente operación.

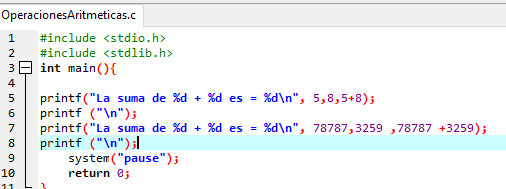
**%d+%d= 78787, 3259, 78787+3259**

**Quinto Programa**

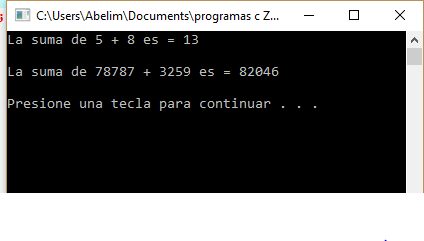
En el método principal (main) escribir el código para

Imprimir la suma de:  5+8 y  78787 + 3259

Declaración de las bibliotecas



Y en pantalla se muestra los resultado de la suma pero si se dan cuenta no me muestra %d+%d si no los numero 5+8 al igual que la otra operación.

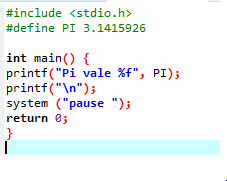
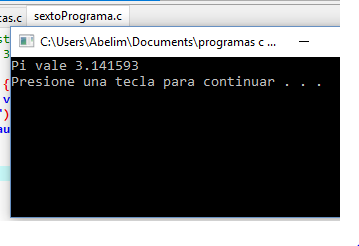


**Sexto programa**

Librerías y donde el lenguaje C nos permite definir constantes con valor numérico por medio de macros.

También en este lenguaje se utiliza lo que es una constante que es un valor que no puede ser alterado durante la ejecución de un programa. Y pi es un valor que no se puede cambia **3.1415**

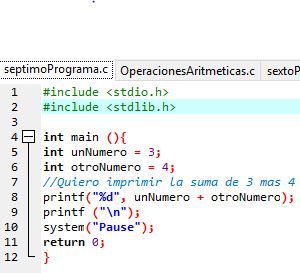
Imprime el valor de pi



Se muestra en pantalla el valor de pi,

**Séptimo programa**

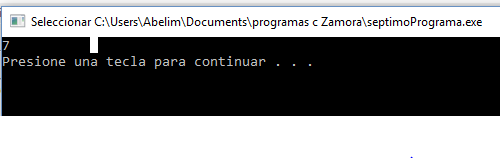
Librerías de C



Se declara las variables de tipo entero que llevan por nombre **unNumero** = 3 y **otroNumero**=4.

Donde se va a imprimir la suma de **unNumero+ otroNumero.**

Y es mostrado el resultado que es 7 en la pantalla.

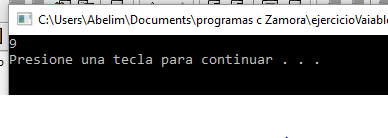
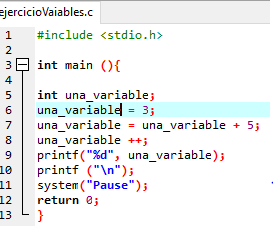


Una variable es una referencia a una sección de memoria.

Entonces se declara la variable de tipo entero con el valor de 3, después, esa variable se le suma 5 y después, al resultado se aumenta el contador 1 más.

Entonces **3+5=8+1=9,**  se manda a imprimir a en pantalla.

**Octavo Programa**



**Noveno programa**

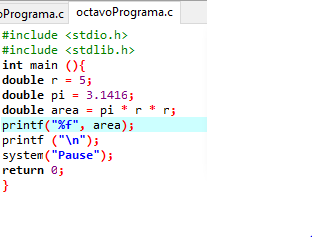
Crea un programa para obtener el área de círculo que tiene como radio 5.

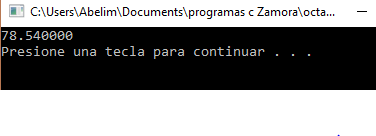
Se declara las bibliotecas.

Se declarara las variables de tipo doublé que nos permite manejar valores con punto decimal. **r=5** y el **pi** que vale 3.1416.

Y se declara otra variable que es **área**  que va a almacenar el resultado de la operación que es pi\*r\*r; y se manda a imprimir y se coloca el nombre de la variable que es **área**.

Muestra el resultado en pantalla el resultado que es 78.54



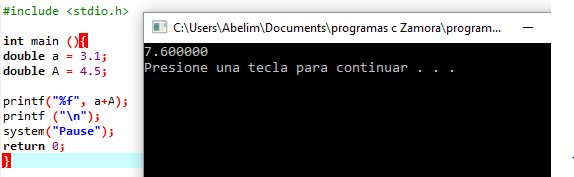


**Decimo programa**

Realiza un programa donde se cumpla la operación de la suma de dos valores con punto decimal.

Declarar la biblioteca datos E/S

El resultado se muestra en pantalla



Se declaran variables de tipo **doublé (punto decimal)** que son las que se va a guardar en la memoria **RAM** y lleva por nombre **A, a**, y que es la suma de **A+a.** Y se manda a imprimir el resultado por medio de un printf,

**Onceavo Programa**

 Ejercicio 1

Declarar 3 variables y asignarles valores enteros.

Mostrar el resultado de la multiplicación de las 3 variables

 Ejercicio 2

Declarar 2 variables dobles (reales) y asignarles valores diferentes de 0.

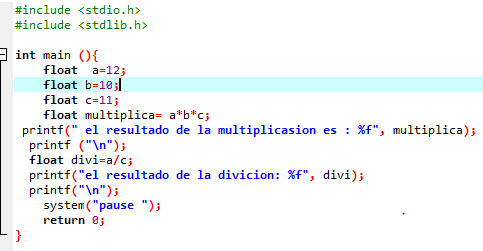
Mostrar el resultado de dividir la primera entre la

Segunda

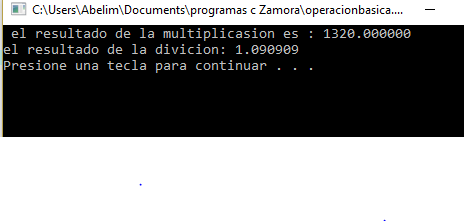
Se declara la biblioteca.

Se declaran 3 variables, de tipo **float a, b,c.**  Y se declara otra que lleva por nombre multiplica donde se va a guardar la operación de la multiplicación de los 3 valores y otra operación que es división de 2 valores, que se va a guardar en la variable **divi**.

Y se manda a imprime **pintf** la variable multiplica. Damos un salto de línea, y después se realiza la impresión del resultado de operación de división llamando a divi.

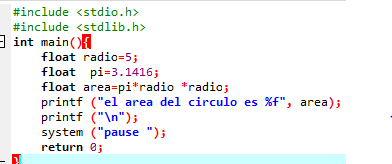


Se muestra el resultado de las operaciones en pantalla.



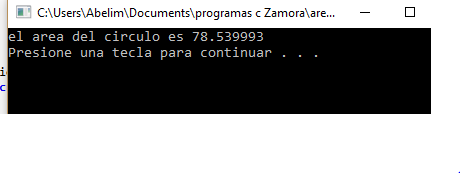
**Programa 13**.

Crea un programa que calcule el área de un círculo que tiene 10 metros de diámetro.



Se declara la biblioteca E/S

Se declara la variables de tipo float con nombre **radio=5 y pi=3.1416**. Y se declaró otra variable de nombre **área** y es donde se guarda la operación de **pi\*radio\*radio** y el resultado se almacena en la variable área y se manda a imprimir en **prinft**



Y se muestra el resultado en pantalla. Con el mensaje **el área del círculo es……**

**Programa 14**

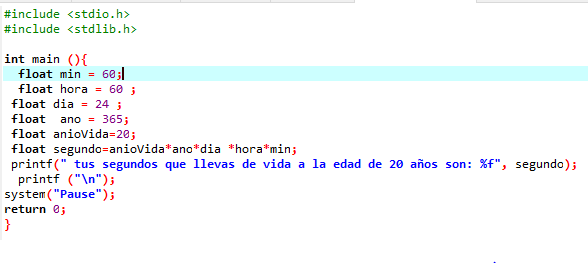
**Calcula la cantidad de segundos que has**

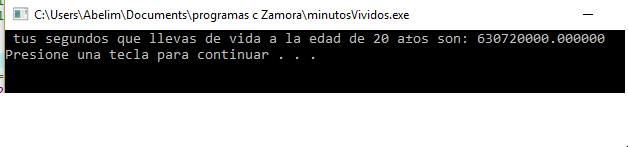
**vivido.**

Son bibliotecas.

Se declaran variables de tipo **float** por que se manejan valores pequeños por poco datos, que llevan por **nombre min = 60, hora = 60, día = 24, ano = 365, anioVida=50**;

Y se crea otra variable del mismo tipo y lleva por nombre segundo que es donde se va a realizar la operación para obtener los **segundos=anioVida\*ano\*día \*hora\*min**.

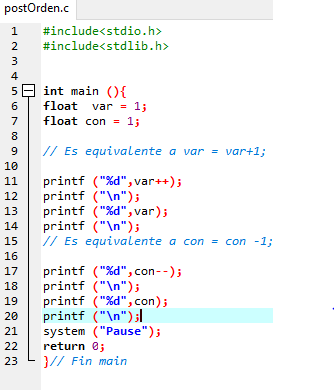




Y se manda a llamar la variable segundo para imprimirla con la ayuda de un **printf (tus segundos que llevas de vida la edad de 20 años son…….)**

Programa 15

Usa el valor actual de la variable, y después le hace un incremento/decremento

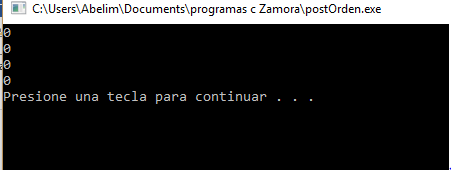


Al inicio colocamos un printf que esto nos realizar la operación dentro de ella y después imprimirlo.

Biblioteca.

Se declaran variables de tipo float con nombre **var=1 y con=1,**  printf se imprime la variable pero se le suma uno, entonces si var=1 +1=0 y se manda a imprimir el resultado.

Así pasa con la siguiente pero en vez se sumar se le resta uno entonces si var=1 -1=0.

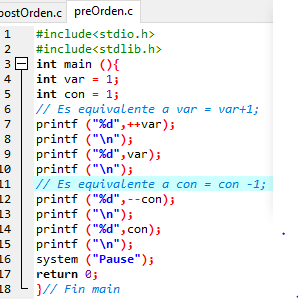


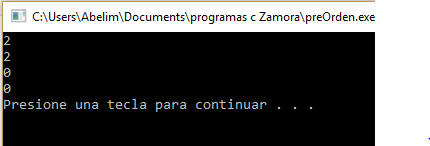
Biblioteca.

Se declaran variables de tipo float con nombre **var=1 y con=1,**  printf se imprime la variable pero se le suma uno, entonces si ++var=1 +1=2 y se manda a imprimir el resultado que es 2. Utilizando \n salto de línea.

Así pasa con la siguiente pero en vez se sumar se le resta uno entonces si var=1 -1=0 y se imprime con ayuda de un printf el resultado que es 0 en la pantalla con ayuda de printf y %d que son un especificador de conversión.

Programa 16



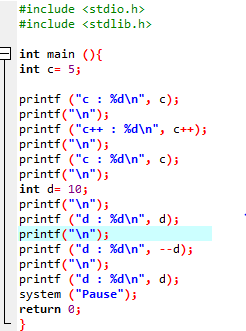


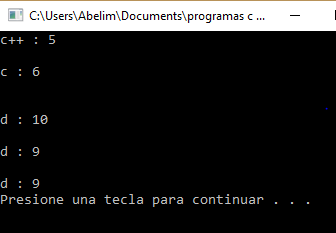
Programa 17

Se declara la variable te tipo entero y que lleva por nombre c y vale 5, entonces se manda a imprimir la variable c y se imprimir, después c se suma 1, entonces si c vale5+1=6 y se imprime el resultado que es 6.

Se declara otra variable te tipo entero y que lleva por nombre d que vale 10, entonces se manda a imprimir la variable d y se imprimir, después d se resta 1, entonces si d vale 10-1=9 y se imprime el resultado que es 9.

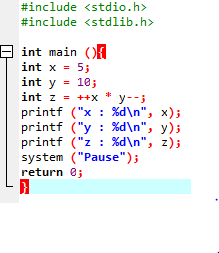
Se imprime los valores en pantalla, con ayuda de printf y %d que es un especificador de conversión.





Programa 18.

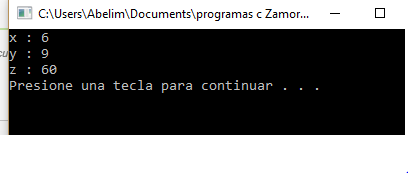
Incremento Decremento. Crea un programa para calcular el valor de **z**



Se declaran las variables de tipo entero x=5 & y=10 y se declara otra z que es la que vamos a calcular, entonces se realiza la operación donde se declaró la variable ++x\*y--; entonces ++5\*10--= se realiza lo siguiente.

Se imprime la variable x con un salto de línea, se imprime y también con un salto de línea y por ultimo z, se ejecuta la operación que tiene y da el resultado de 60. Porque 5+1=6\*10=60

Y se imprime los resultados en pantalla, para el usuario.



Se declaran las variables de tipo entero que llevan por nombre: **p=5, q=1, r=2, w=3, x=9, y=6, z**  se desconoce.

Entonces para encontrar **z=p\*r %q + w / x - y**;

Para resolver esta operación en necesario utilizar la jerarquía de operaciones del lenguaje C.

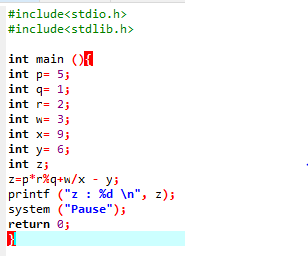
Se debe de que hay que resolver la operación 1-p\*r; 2-r%q; 3-w/x

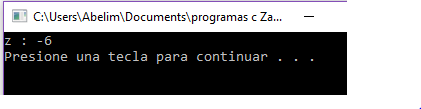
4-q+w; 5-x-y

1-**5\*2** 2-**2%1**; 3-**3/9** 4-**1+3**; 5- **9-6=** -6

Y se imprime el resultado en pantalla.

**Programa 19**





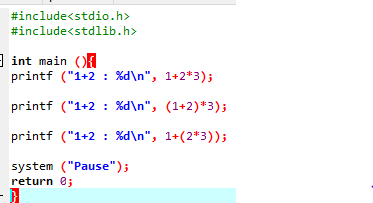
**Programa 20**

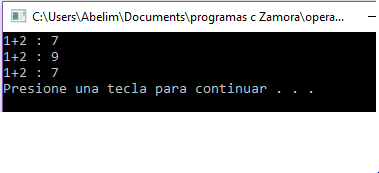
Biblioteca

Se imprime el resultado de **1+2\*3**, recordando que se soluciona por la jerarquía de operaciones, primero la multiplicación y después la **suma (2\*3=6+1=7)**

En la siguiente apartado se muestra otra operación que se va a **imprimir ((1+2)\*3)** lo primero es resolver lo que se encuentra en paréntesis que es **(1+2)=3** y después se **multiplica (3) (3)=9**

Y en la última operación es **1+ (2\*3)),** lo primero que se va a hacer es la multiplicación que se encuentra entre paréntesis, que es **(2\*3)=6** y después **1+6=7**





Se imprime los valores en pantalla 7, 9, 7

Es la impresión de valores de operadores lógico, donde && es condicional. Las dos se deben de cumplir. Si 1 es verdadero y 1 es verdadero entonces el resultado es verdadero =1;

Si 1 es verdadero y 0 es falso entonces el resultado es falso=0;

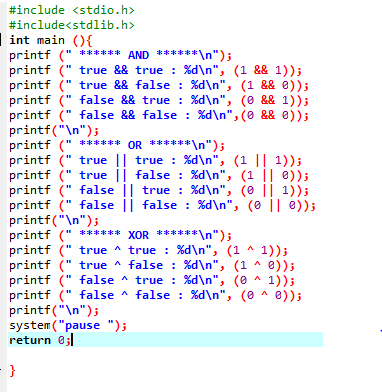
Si 0 es falso y 1 es verdadero entonces el resultado es falso=0;

Si 0 es falso y 0 es falso también entonces el resultado es falso=0;

Si 1 es verdadero y 0 es falso entonces el resultado es falso=0;

**Programa 22**

**Operadores lógicos**



Es la impresión de los valores XOR que por lo menos una de las condiciones deben de ser verdaderas.

Si 1 es verdadero y 1 es verdadero entonces el resultado es falso =0;

Si 1 es verdadero y 0 es falso entonces el resultado es verdadero=1;

Si 0 es falso y 1 es verdadero entonces el resultado es verdadero=1;

Si 0 es falso y 0 es falso también entonces el resultado es falso=0;

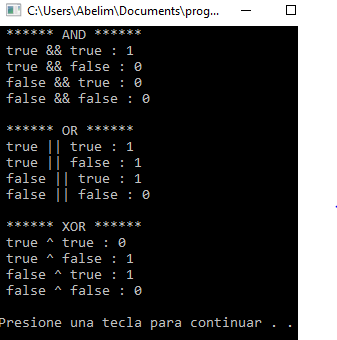
Es la impresión de los valores OR que por lo menos una de las condiciones deben de ser verdaderas.

Si 1 es verdadero y 1 es verdadero entonces el resultado es verdadero =1;

Si 1 es verdadero y 0 es falso entonces el resultado es verdadero=1;

Si 0 es falso y 1 es verdadero entonces el resultado es verdadero=1;

Si 0 es falso y 0 es falso también entonces el resultado es falso=0;

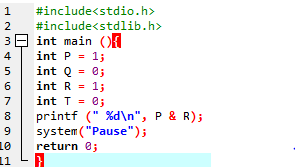


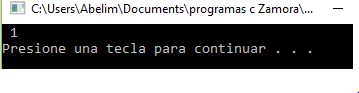
Se imprimen los valores dependiendo de la condición 1es verdadero y 0 es falso

**23 Programas operadores2**

Se declaran las variables de tipo entero y se inicializan con los valores que ya no se van a cambiar en la trayectoria de la ejecució**n P=1, Q=0, R=1, T=0,**

Manda a llamar a P&R y realiza la comparación de **P**  es true o false y al igual que **R, 0 es false y 1 es true,** entonces ambas son verdadera entonces el resultado es true





Se manda a imprimir con printf el resultado de la comparación que es true i se mostrara el **valor de 1**

**Programa 24**

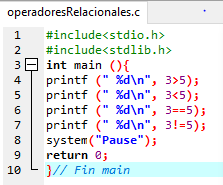
Se declaran las bibliotecas.

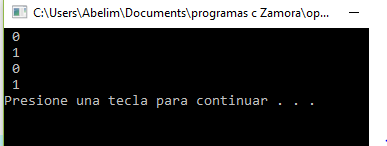
Se imprime las siguientes condiciones, donde 3 mayor que 5, no, entonces el resultado en false=0;

3 es menor de 5, si, entonces el resultado en true=1;

3 es igual que 5, no, entonces el resultado es false=0;

3 es diferente que 5, si, entonces el resultado es true=1;

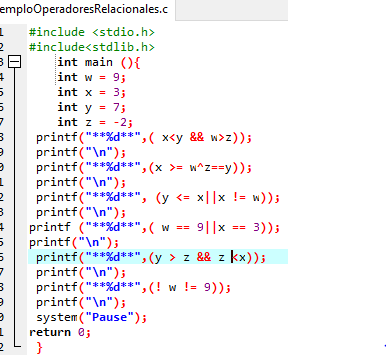




Se muestran los resultados con la ayuda de printf en pantalla, para el usuario.

**Programa 25**

De declaran las variables de tipo entero **w=9, x=3, y=7, z=2**.



En el primera operación nos dice que;

3 menor que 7 (**si)** && 9 es mayor que -2 (**si**), entonces en **true=1**; especificador de conversión, y se coloca el nombre de la variable donde esta almacenada la operación y el resultado

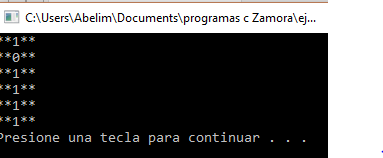
En segunda operación es 3 es mayor o igual que 9 (no**)** ^ -2 es igual que 7(**no**) entonces es **false =0**;

En tercera operación 7 es menor o igual que 3(**no**) || 3 es diferente de 9 (**si**) entonces es **true =1**;

Cuarta operación 9 es igual que 9(**si**) || 3 es igual que 3(**si**), entonces es **true=**1;

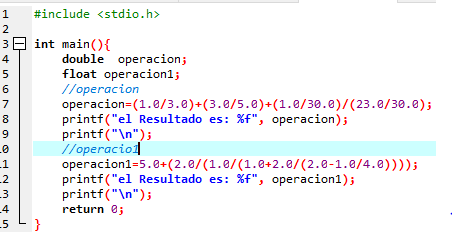
Quinta operación, 7 es mayor que -2(**si**) && -2 es menor que 3(**si**), entonces **true=1**;

Ultima 9 es diferente que 9 (**no**) es **true=1**;



Se muestra en pantalla los resultados de las 5 operaciones lógicas con ayuda de %d que sabemos que es especificador de conversión.

**Programa 26.**

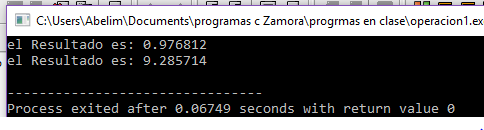


De declaran variables con los nombres operación de tipo doublé y otro de operacion9 de tipo **float.**

La variable **operación**, se inicializa y se coloca como se va a solucionar, y recordemos que la jerarquía de operaciones, primero la operación de la división, y posteriormente la suma de la operaciones, y se manda a imprimir con printf que sabemos que es un especificador de conversión, y siguiente operación y un salto de línea. Y lo mismo pasa en la operacion1 se realiza la división y posteriormente la suma de los resultados.

Su van a imprimir en pantalla.

Especificador de conversión



**V. Conclusiones:**

En el desarrollo de los programas del lenguaje de programación en C.

Maneja el uso de datos de entra y salida, no es parte del lenguaje si ni que proporciona a través de una biblioteca de funciones, permite la agrupación de instrucciones, es programación estructurada economizando las expresiones con abundancia usando los operadores, tipos de datos, codificando en alto y bajo nivel. En la realización de este pequeño manual de práctica sobre los programas realizados en clase, utilizando el lenguaje C, permitiendo desarrollar lógica, utilizar operadores lógicos false (0) y true (1).

Usa funciones matemáticas propias del lenguaje y funciones suministradas por bibliotecas para el manejo de archivos. Es un lenguaje muy flexible, permite programar con múltiples estilos, uno de los estilos más usados es el estructurado. Usa un grupo muy pequeño de palabras clave, lo cual facilita la memorización por parte del programador.