

## PRÁCTICA/LABORATORIO N° 01

### LENGUAJES DE PROGRAMACIÓN

#### Indicaciones para el envío:

- Enviar dentro de un **.zip** los archivos **.cpp** generados en la práctica.

#### Objetivos:

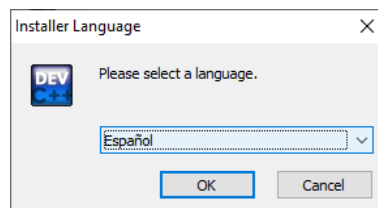
0. Descargar e instalar [Dev-C++](#).
1. Configurar Dev-C++.
2. Mi primer “Hola mundo” en C++.
3. Variables e identificadores en C++.
4. Operaciones entre variables en C++.
5. Expresiones aritméticas en C++.
6. Operaciones lógicas en C++.

#### 0. Instalación de Dev-C++

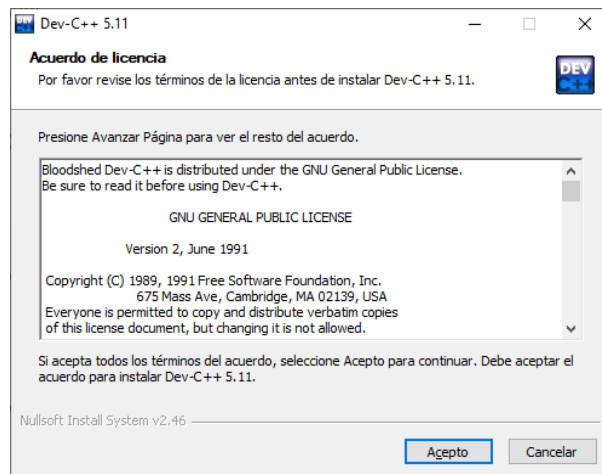
El primero paso es descargar el IDE Dev-C++ que nos permitirá programar y compilar en C++, el enlace es el siguiente:

<https://sourceforge.net/projects/orwelldvcpp/files/latest/download>

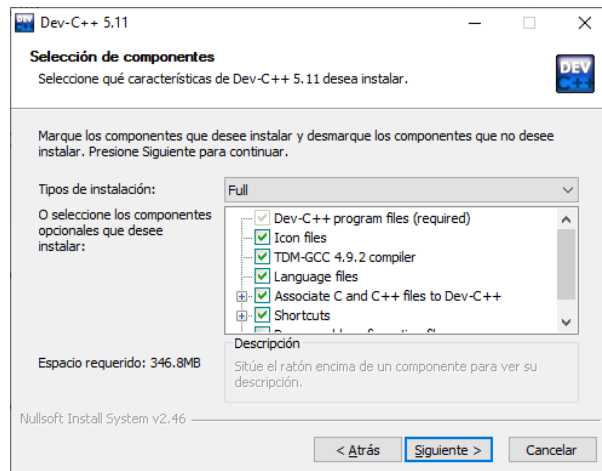
Una vez descargado, debemos ejecutar el instalador, de requerirse privilegios de administrador aparecerá un cuadro en el que debemos hacer click en **SI**. Luego, nos encontraremos con la siguiente ventana:



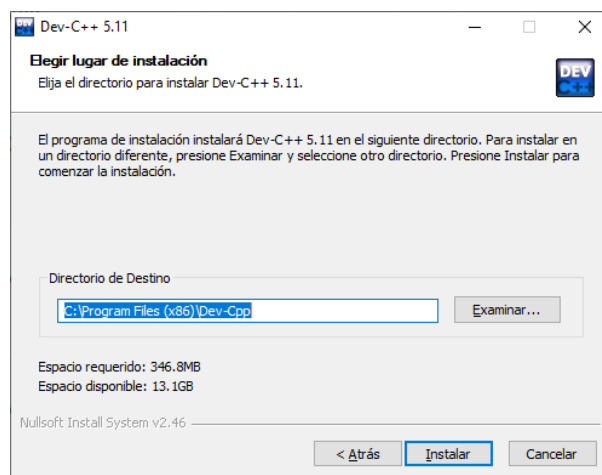
Seleccionaremos **Español** y haremos click en **OK**. Nos encontraremos con la siguiente ventana:



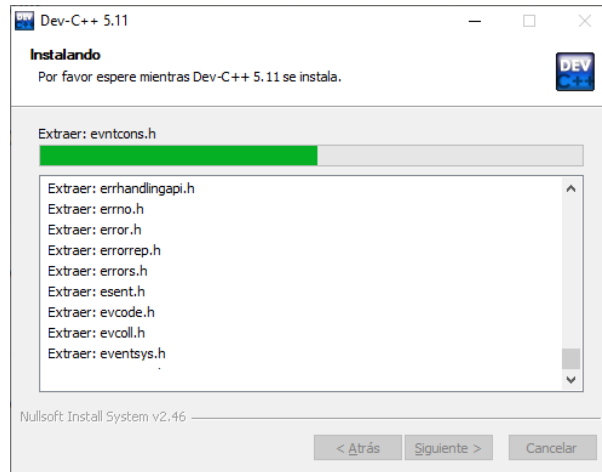
Nos mostrará el acuerdo de licencia de uso, hacer click en **Acepto**. Nos encontraremos con la siguiente ventana:



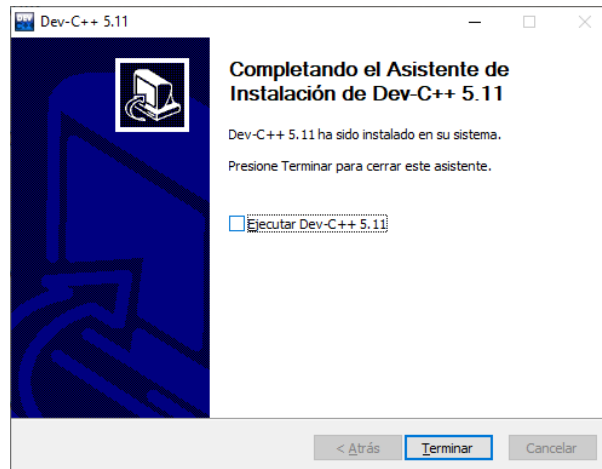
Nos muestra los componentes a instalarse, seleccionaremos **FULL** y haremos click en **Siguiente>**. Nos encontraremos con la siguiente ventana:



Nos mostrará la ruta de instalación, para la mayoría de casos solo haremos click en **Instalar**. Nos encontraremos con la siguiente ventana:



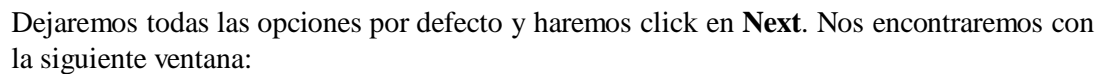
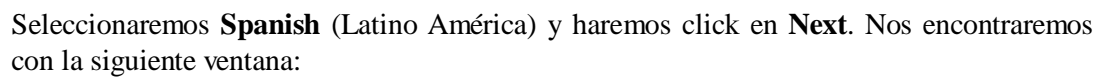
Esperaremos que la barra gris se torne completamente verde mientras se extraen e instalan los componentes necesarios. Nos encontraremos con la siguiente ventana:

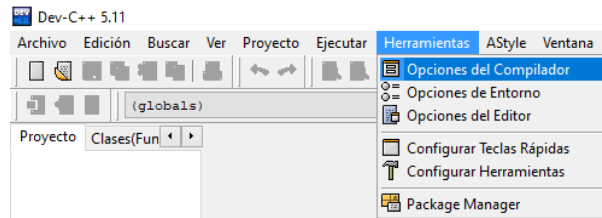


**Desactivaremos** la opción Ejecutar Dev-C++ 5.11 y haremos click en **Terminar**. Luego, veremos en el escritorio el siguiente acceso directo:

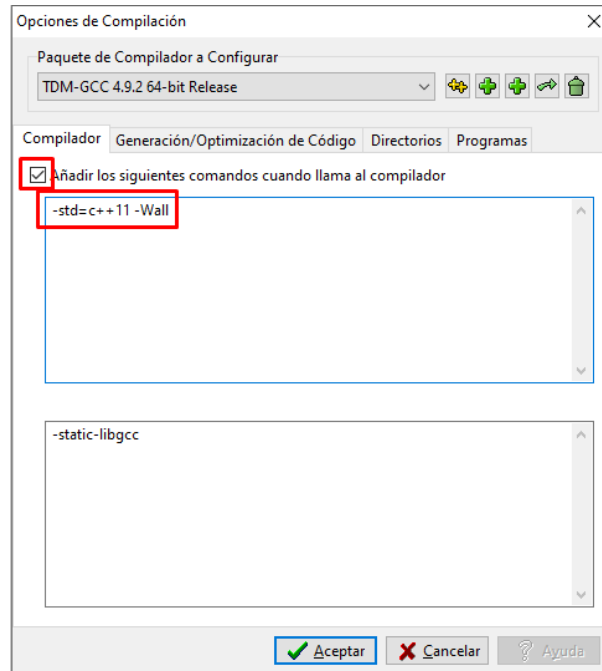


Haremos doble click en él ícono del escritorio para acceder a **Dev-C++**. Nos encontraremos con la siguiente ventana:



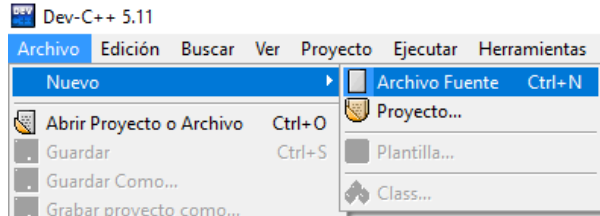


Nos encontraremos con la siguiente ventana:

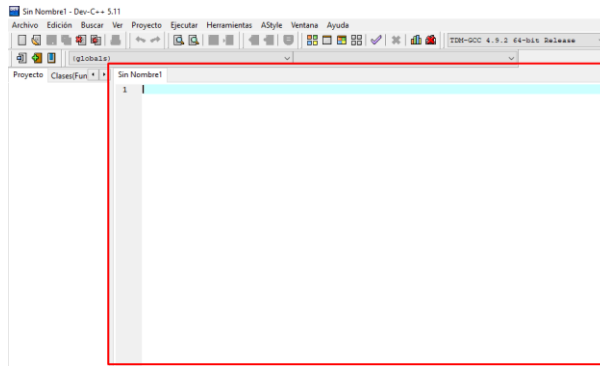


## 2. Mi primer Hola mundo en C++

A continuación, crearemos nuestro primer “**Hola mundo**” para asegurarnos de que podemos empezar a programar en C++. Iremos a **Archivo -> Nuevo -> Archivo Fuente**:



Se habilitará la sección de edición de archivos del IDE (zona del cuadro rojo), donde podemos empezar a escribir nuestro código:



Escribiremos el siguiente código:

```
[*] Sin Nombre1
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      |
8      return 0;
9  }
```

<b>Línea 1:</b> Se incluyen las librerías estándar.
<b>Línea 3:</b> Utilizaremos el espacio de nombres de la librería estándar.
<b>Línea 5:</b> Declaramos nuestra función principal <b>main</b> .

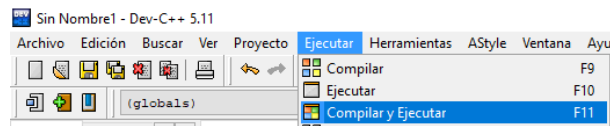
Esta es la **plantilla** que se repetirá en todos los códigos que programemos.

En la **línea 7** escribiremos un “Hola mundo”. En C++ puedes utilizar la sentencia **cout<<** para imprimir resultados en pantalla. Escribiremos **cout<<”Hola Mundo”**

**Hola mundo** debe ir entre comillas puesto que nos referimos a una **cadena de caracteres**.

```
[*] Sin Nombre1
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      cout<<"Hola mundo"
8      return 0;
9  }
```

A continuación, iremos a **Ejecutar -> Compilar y Ejecutar**.

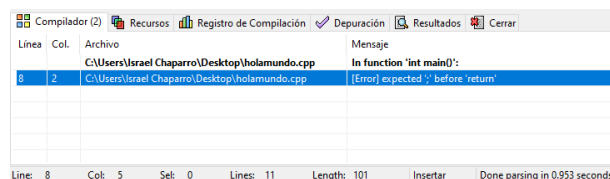


Se nos mostrará un cuadro de diálogo para elegir dónde y con qué nombre **guardar** nuestro código fuente. Notarás que el archivo que guardes tendrá la extensión **.cpp**, lo cual indica que es un código fuente de C++ (C plus plus). **GUARDA EL ARCHIVO COMO holamundo.cpp**

Como recomendación, puedes crear una carpeta y empezar a guardar allí tus programas. Luego, verás lo siguiente en el editor de código:

```
holamundo.cpp
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      cout<<"Hola mundo"
8      return 0;
9  }
```

Como puedes ver, ha aparecido una **línea color rojo**, usualmente en señal de **error**. En algunas ocasiones, los errores se encuentran justo antes de la línea marcada, podemos encontrar más información en la caja de mensajes inferior.



La caja de mensajes inferior, la primera pestaña nos muestra los mensajes del compilador.

En este caso, nos indica que en la función “**int main()**” existe un “[**Error**]” en la **línea 8** que indica “**Se esperaba un ‘;’ antes de return**”. Esto nos indica, literalmente, que debimos colocar un ‘;’ antes de escribir **return**.

En este caso, olvidamos colocar ‘;’ luego de **cout<<“Hola mundo”**

A continuación, corregimos el error:

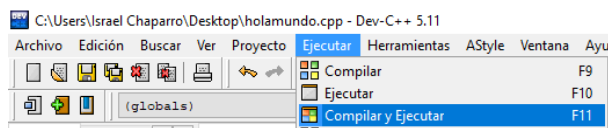
```
[*] holamundo.cpp
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      cout<<"Hola mundo";
8      return 0;
9  }
10
11
```

C++ requiere que coloquemos un ‘;’ cada vez que terminamos una **sentencia**.

Notarás del código mostrado que **algunas** líneas **requieren** un ; al final y algunas **no**. Con el tiempo llegarás a comprender por completo donde colocar un ; y dónde no.

Entender los **errores que detecta el compilador** (y que aparecen en la caja de mensajes) es una habilidad muy necesaria en un programador, si no entiendes el error puedes googlearlo, buscar en stackoverflow o preguntarle a ChatGPT. Si el error persiste consulta a tu docente o jefe de prácticas; si no estás en clase, envía un mensaje.

Ahora, intentaremos nuevamente **compilar y ejecutar el programa**.



Luego de unos segundos, nos aparecerá una ventana de la consola con el mensaje **Hola mundo** como la que mostramos a continuación:

```
C:\Users\Israel Chaparro\Desktop\holamundo.exe
Hola mundo
-----
Process exited after 0.07269 seconds with return value 0
Presione una tecla para continuar . . .
```

**¡Felicidades!**, acabas de escribir, compilar y ejecutar tu **primer programa en C++**.

No te olvides de cerrar el programa después de cada ejecución.

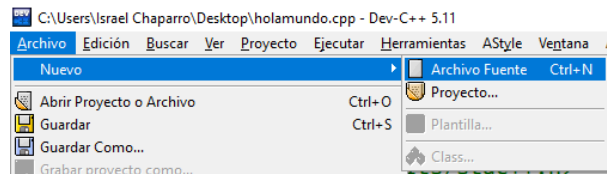
Ya conoces como **escribir** en pantalla (**cout**), pero, ¿qué hay acerca de **leer** valores desde el teclado?



### 3. Variables e identificadores en C++

Al **leer** valores del teclado, necesitamos **almacenar** la lectura en algún lugar, para eso existen las **variables** de C++. Principalmente los tipos de variable más usados son: bool, int, float, double y string.

A continuación, ve a **Archivo -> Nuevo -> Código Fuente**:



Escribe el siguiente código:

```
holamundo.cpp
1  #include<iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int a;
8      cout<<"Ingrese un valor entero: ";
9      cin>>a;
10     cout<<"El valor ingresado es: "<<a;
11     return 0;
12 }
```

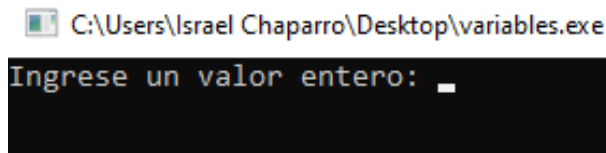
**Línea 7:** La sentencia **int a** nos permite declarar una variable de **identificador/nombre a**. Esto es, un número entero.

**Línea 9:** La sentencia **cin>>a** sirve para leer, almacenando en la variable **a**, un valor que puedes ingresar a través del teclado (o haciendo copia y pega en la terminal).

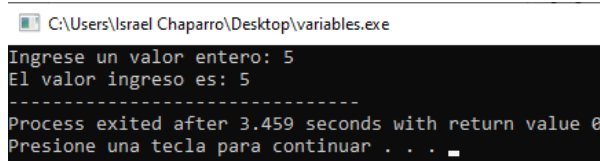
**Línea 10:** La sentencia **cout** nos permite imprimir en pantalla. Notarás que en una sola línea estamos escribiendo en pantalla un mensaje y la variable **a**. Esto es posible enlazando dos veces el operador **<<**. Podrías enlazar más operadores **<<** si deseas imprimir más textos o variables. Puedes escribir **endl** para realizar un salto de línea. Al imprimir **a**, se imprimirá el valor almacenado en esta variable.

**Compila y ejecuta** (o solo aprieta **F11** en tu teclado) como ya lo has hecho antes. Te aparecerá un cuadro de diálogo para guardar el nuevo archivo. **GUARDA EL ARCHIVO COMO variables.cpp**

Luego, se abrirá la consola y mostrará el siguiente mensaje:



Podrás ver que el cursor de la consola está tintineando, esto es que está a la espera de que ingreses un valor. Ingresa, por ejemplo, el **5** (prueba luego con otro valor, incluso un negativo).



El programa ha finalizado y se ha comportado según lo descrito en el código.

Vuelve a ejecutar el programa y responde las siguientes preguntas: ¿Qué pasa si ingreso el valor **12.5**? ¿Qué pasa si ingreso **hola**? ¿Qué pasa si ingreso **-2147483649** o **2147483648**?

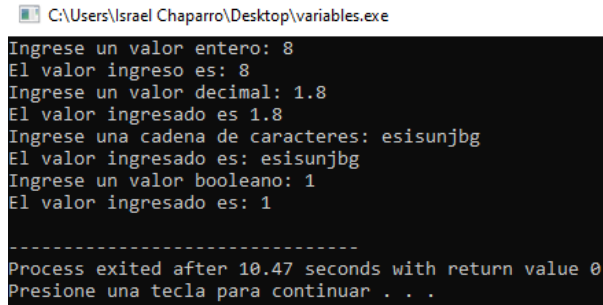
Como ya revisamos en teoría, **cada tipo de dato tiene un tipo de variable**. En C++, **int** es para **enteros**, **float** para valores con **punto decimal**, **string** para **caracteres**, **bool** para **booleanos**. Adicionalmente, algunas variables tienen límites: **int**: [-2147483646, 2147483647]. Si necesitas trabajar con un valor más alto, puedes usar **long long**.

Escribe el siguiente código:

```
[*] variables.cpp
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      int a;
8      cout<<"Ingrese un valor entero: ";
9      cin>>a;
10     cout<<"El valor ingreso es: "<<a<<endl;
11
12     float b;
13     cout<<"Ingrese un valor decimal: ";
14     cin>>b;
15     cout<<"El valor ingresado es "<<b<<endl;
16
17     string c;
18     cout<<"Ingrese una cadena de caracteres: ";
19     cin>>c;
20     cout<<"El valor ingresado es: "<<c<<endl;
21
22     bool d;
23     cout<<"Ingrese un valor booleano: ";
24     cin>>d;
25     cout<<"El valor ingresado es: "<<d<<endl;
26
27     return 0;
28 }
```

Lee y comprende el código, luego **compila y ejecuta**.

Se abrirá nuevamente la consola, luego ingresa los valores que tú desees pero que cumplan la condición señalada (entero, decimal, cadena de caracteres **SIN** espacio y **1** para verdadero o **0** para falso). A continuación, un ejemplo:



```
C:\Users\Israel Chaparro\Desktop\variables.exe
Ingrese un valor entero: 8
El valor ingreso es: 8
Ingrese un valor decimal: 1.8
El valor ingresado es 1.8
Ingrese una cadena de caracteres: esisunjbg
El valor ingresado es: esisunjbg
Ingrese un valor booleano: 1
El valor ingresado es: 1

-----
Process exited after 10.47 seconds with return value 0
Presione una tecla para continuar . . .
```

Ahora, ya sabes cómo **leer valores**, **asignarlos** a una variable, e **imprimir** el valor de una variable.

#### 4. Operaciones entre variables en C++

A continuación, ve a **Archivo -> Nuevo -> Código Fuente**:

Revisemos a continuación un ejemplo de **operaciones de variables**. Escribe el siguiente código (puedes copiar el código del programa anterior y realizar las modificaciones necesarias):

```
[*] operaciones.cpp
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      int a;
8      cout<<"Ingrese un valor entero: ";
9      cin>>a;
10     a=a*2;
11     cout<<"El valor ingresado multiplicado por 2 es: "<<a<<endl;
12
13     float b;
14     cout<<"Ingrese un valor decimal: ";
15     cin>>b;
16     b=b/3;
17     cout<<"El valor ingresado dividido entre 2 es "<<b<<endl;
18
19     float c;
20     c=a*b;
21     cout<<"La multiplicacion de a por b es: "<<c<<endl;
22
23     bool d;
24     cout<<"Ingrese un valor booleano: ";
25     cin>>d;
26     d=!d;
27     cout<<"La negacion del valor ingresado es: "<<d<<endl;
28
29     return 0;
30 }
```

<b>Línea 10:</b> Multiplicamos el valor de <b>a</b> por <b>2</b> , y almacenamos el resultado en <b>a</b> .
---

<b>Línea 16:</b> Dividimos el valor de <b>b</b> por <b>3</b> , y almacenamos el resultado en <b>b</b> .
---

<b>Línea 20:</b> Multiplicamos el valor de <b>a</b> por el valor de <b>b</b> y lo almacenamos en <b>c</b> .
---

<b>Línea 26:</b> Negamos el valor de <b>d</b> .
---

Lee y comprende el código, luego **compila y ejecuta** como ya lo has hecho antes. **GUARDA EL ARCHIVO COMO operaciones.cpp**

Se abrirá la consola y se ejecutará el programa. Ingresas los valores que tú desees que cumplan la condición señalada (entero, decimal, y **1 para verdadero o 0 para falso**). A continuación, un ejemplo de resultado de ejecución:

```
C:\Users\Israel Chaparro\Desktop\operaciones.exe
Ingrese un valor entero: 4
El valor ingresado multiplicado por 2 es: 8
Ingrese un valor decimal: 1.9
El valor ingresado dividido entre 2 es 0.633333
La multiplicacion de a por b es: 5.06667
Ingrese un valor booleano: 0
La negacion del valor ingresado es: 1
-----
Process exited after 5.16 seconds with return value 0
Presione una tecla para continuar . . .
```

## 5. Expresiones aritméticas en C++

A continuación, ve a **Archivo -> Nuevo -> Código Fuente**:

Luego, escribe el siguiente código:

```
[*] Sin Nombre4
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      int a=1;
8      cout<<(a+(12%10)*pow(5,2)/sqrt(9))+9/2<<endl;
9      cout<<(a+(12%10)*pow(5,2)/pow(9,1/2.))+9/2<<endl;
10     return 0;
11 }
```

**Línea 7:** Declaramos la variable **a** de tipo **int** (entero) y le asignamos el valor de **1**.

**Línea 8:** Calculamos una **expresión aritmética**. Cómo ya ha sido estudiado, la prioridad son primero los **paréntesis** (incluidas las funciones como **pow(5,2)** que representa  $5^2$  y **sqrt** que representa  $\sqrt{2}$ ), luego las **multiplicaciones**, **divisiones** y **módulos** y finalmente las **sumas** y **restas**.

**Línea 9:** Reemplazamos **sqrt(9)** por **pow(9,1/2.)**. La función **sqrt** solo nos permite calcular la raíz cuadrada, si queremos calcular raíces de otros radicales como cúbicas, cuarta, etc. necesitamos usar la función **pow** y pasar como segundo parámetro **1/radical**. Siendo “1” y “radical” valores **int** (enteros), el resultado será un valor **int** (**0**). Es por ello que colocamos un **.** luego de **1/radical** para señalar que el resultado será un **decimal** (**0.5**). El valor 9 elevado a 0.5 es igual a la raíz cuadrada de 9.

Realiza la **operación a mano** en una hoja, **recuerda que la división entre dos enteros (ej. 7/3) da un entero (la parte entera de la operación: 2).**

**Compila y ejecuta** el código. **GUARDA EL ARCHIVO COMO expresiones.cpp.**

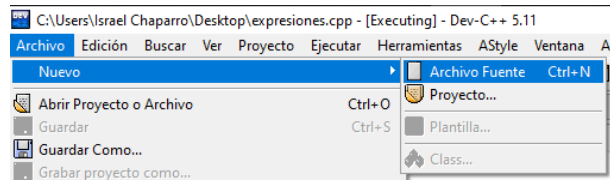
Se mostrará la consola con el siguiente resultado:

```
C:\Users\Israel Chaparro\Desktop\expresiones.exe
21.6667
21.6667
-----
Process exited after 0.07997 seconds with return value 0
Presione una tecla para continuar . . .
```

Podemos ver que se imprime dos veces el mismo valor (esto es, que **sqrt(9)** es igual a **pow(9,1/2.)**). Adicionalmente, este valor debe coincidir con la **operación a mano** que realizaste.

## 6. Operaciones lógicas en C++

A continuación, ve a **Archivo -> Nuevo -> Código Fuente**:



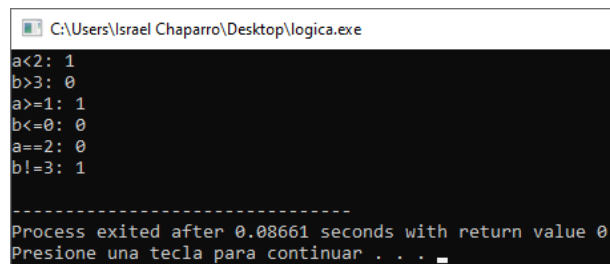
Luego, escribe el siguiente código:

```
logica.cpp
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      int a=1;
8      int b=2;
9      cout<<"a<2: " <<(a<2)<<endl;
10     cout<<"b>3: " <<(b>3)<<endl;
11     cout<<"a>=1: " <<(a>=1)<<endl;
12     cout<<"b<=0: " <<(b<=0)<<endl;
13     cout<<"a==2: " <<(a==2)<<endl;
14     cout<<"b!=3: " <<(b!=3)<<endl;
15     return 0;
16 }
```

De la línea **9** a la **14**, se ejecutan entre paréntesis **operadores relacionales** cuyos resultados son **1** (verdadero) o **0** (falso).

Realiza la **operación a mano** en una hoja. Luego **Compila** y **Ejecuta**. **GUARDA EL ARCHIVO COMO logica.cpp.**

Se mostrará la consola con el siguiente resultado:



```
C:\Users\Israel Chaparro\Desktop\logica.exe
a<2: 1
b>3: 0
a>=1: 1
b<=0: 0
a==2: 0
b!=3: 1
-----
Process exited after 0.08661 seconds with return value 0
Presione una tecla para continuar . . .
```

Los resultados mostrados deben coincidir con la **operación a mano** que realizaste.

Actualiza el código de la siguiente forma:

```
logica.cpp
1  #include<bits/stdc++.h>
2
3  using namespace std;
4
5  int main()
6  {
7      int a=1;
8      int b=2;
9      cout<<"a<2: "<<(a<2)<<endl;
10     cout<<"b>3: "<<(b>3)<<endl;
11     cout<<"a>=1: "<<(a>=1)<<endl;
12     cout<<"b<=0: "<<(b<=0)<<endl;
13     cout<<"a==2: "<<(a==2)<<endl;
14     cout<<"b!=3: "<<(b!=3)<<endl<<endl;
15
16     bool c=0,d=1;
17     cout<<"0 or 1: "<<(c or d)<<endl;
18     cout<<"0 or 0: "<<(c or c)<<endl;
19     cout<<"1 and 1: "<<(d and d)<<endl;
20     cout<<"1 and 0: "<<(d and c)<<endl;
21     cout<<"!0: "<<(!c)<<endl;
22     cout<<"!(1 and 1) or (0 or 0): "<<(!(d and d) or (c or c))<<endl;
23     return 0;
24 }
```

**Realiza las nuevas operaciones a mano** descritas en las líneas 16-22. Luego **Compila y Ejecuta**.

Se mostrará la consola con el siguiente resultado:

```
C:\Users\Israel Chaparro\Desktop\logica.exe
a<2: 1
b>3: 0
a>=1: 1
b<=0: 0
a==2: 0
b!=3: 1

0 or 1: 1
0 or 0: 0
1 and 1: 1
1 and 0: 0
!0: 1
!(1 and 1) or (0 or 0): 0

-----
Process exited after 0.08683 seconds with return value 0
Presione una tecla para continuar . . .
```

Los resultados mostrados deben coincidir con la **operación a mano** que realizaste.