

## Las partes de un programa en C++

Para comenzar, tenemos el formato general de un programa hecho en C++:

```
#include <iostream.h>
#include <conio.h>

int main()
{
    clrscr();
    cout<<"Hola Mundo";
    getch();
}
```

Aquí esta su explicación punto por punto:

#include <iostream.h>    //Librería necesaria para cout.

#include <conio.h>        //Librería necesaria para getch() y clrscr();

int main()                //Rutina o función principal

//Determina donde empieza el código a ejecutar

clrscr();                //Borra el contenido de la pantalla

cout<<" ";                //Escribe en la pantalla

getch();                //Hace una copia de la pantalla

}                        //Indica que el código terminó

## Una parte importante de todo código de programación son los comentarios:

Los comentarios, son partes del código que sirven como referencia para otros programadores que desean modificar el código fuente. Además, si un día formulas un código complejo, a los tres meses podrías no saber como funciona, pero si lo has comentado no tendrás problemas.

Los comentarios son de dos tipos:

// --> Comenta una línea

/\* \*/ --> Comenta un bloque de código, de varias líneas

Aquí hay un pequeño ejemplo:

```
#include <iostream.h> //Librería para cout y cin
#include <conio.h>      //Librería para clrscr() y getch()

int main()
{
    /*Este es un comentario
    continúa hasta que se cierre
    el slash*/
    clrscr(); //Borra la pantalla
    cout<<"Hola Mundo"; //Escribe un mensaje en la pantalla
    //Este es otro tipo de comentario, que solo comenta una línea
    getch(); //Copia la pantalla
}
```

# Variables y Constantes

## Que es una Variable:

En C++, una variable es un lugar para guardar información, es un espacio en memoria que guarda datos por un periodo determinado de tiempo. Hay dos tipos, unsigned y signed. Todas las variables son signed por default (Signed incluye negativos).

## Tipos de Variables:

**Bool** --> Verdadero o Falso  
**unsigned short int** --> 0 a 65,535  
**short int** --> -32,768 a 32,767  
**unsigned long int** --> 0 a 4,294,967,295  
**long int** --> -2,147,483,648 a 2,147,483,647  
**int** --> -32,768 a 32,767  
**unsigned int** --> 0 a 65,535  
**char** --> 256 caracteres  
**float** --> 1.2e-38 a 3.4e38  
**double** --> 2.2e-308 a 1.8e308

## Definiendo una Variable:

Para declarar una variable, primero debe escogerse su tipo, ya sea int, long, o short int. Luego, hay que darle un nombre que, sea de preferencia relacionado con la función que va a realizar. Las variables deben ser llamadas tal como han sido declaradas. Edad no es lo mismo que EDAD o que EdaD. Por ejemplo:

```
#include<iostream.h>    //Librería para cout y cin
#include<conio.h>        //Librería para clrscr() y getch()

int main()
{
    unsigned short clases; //Declaración de variables
    long total;
    int alumnos;
    total = clases * alumnos; //Asignamos a la variable Total, la multiplicación de
```

```
//clases por alumnos
cout<<"El total es: "<<total; //Imprimimos el valor de la variable Total
getch(); //Copiamos la pantalla
}
```

### **Creando más de una variable a la vez:**

Se puede declarar más de una variable a la vez, separándolas por comas:

```
#include<iostream.h> //Librería para cout y cin
#include<conio.h> //Librería para clrscr() y getch()

int main()
{
    unsigned int miEdad,tuEdad; //Declaramos dos variables en la misma línea
    long Dia,Mes,Anho; //Declaramos tres variables en la misma línea
    getch(); //Copiamos la pantalla
}
```

### **Asignando valores a las variables:**

Para asignarle valores a una variable, se utiliza el símbolo "=":

```
#include<iostream.h> //Librería para cout y cin
#include<conio.h> //Librería para clrscr() y getch()

int main()
{
    unsigned short clases = 5; //Declaración de variables
    long total;
    int alumnos = 25; //Asignamos el valor 25 a la variable Alumnos
    total = clases * alumnos; //Asignamos a la variable Total, la multiplicación de
    //las variables Clases por Alumnos
    getch();
}
```

Otro ejemplo:

```
#include<iostream.h> //Librería para cout y cin
#include<conio.h>      //Librería para clrscr() y getch()

int main()
{
    unsigned short int Ancho = 5,Largo; //Declaración de variables
    Largo = 10;

    unsigned short int Area = (Ancho * Largo); /*Declaramos la variable Area y le
    asignamos el valor de la multiplicación de la variable Ancho por la variable
    Largo*/

    cout<<"Ancho:"<<Ancho<<"\n";    // "\n" Sirve para bajar una línea.
    cout<<"Largo:"<<Largo<<"\n";    //Imprimimos los valores de las variables
    cout<<"Area:"<<Area<<"\n";
    getch();    //Copiamos la pantalla
}
```

### Usando typedef:

typedef, es un comando que te permite crear un alias para la declaración de funciones, esto es, que puedes crear un alias para unsigned short int y evitar tener que escribirlo cada vez que lo necesites.

```
#include<iostream.h> //Librería para cout y cin
#include<conio.h>      //Librería para clrscr() y getch()
typedef unsigned short int USHORT; //Declaramos un tipo de variable
//utilizando typedef

int main()
{
    USHORT Ancho = 5; //Utilizamos el tipo de variable que hemos creado para
    USHORT Largo = 2; //poder crear nuevas variables
}
```

```
USHORT Area = Ancho * Largo; //Asignamos a la variable Area, el valor de la
//multiplicación de la variable Ancho por la variable Largo
cout<<"Area: "<<Area<<endl;    //endl, cumple la función de "\n"
getch(); //Copiamos la pantalla
}
```

### Caracteres especiales:

- \* "\n" --> Nueva Línea o Salto de Carro
- \* "\t" --> Tab o espacio
- \* "\b" --> Backspace o retroceder una línea
- \* "\"" --> Comilla doble
- \* "\"" --> Comilla simple
- \* "\?" --> Signo de interrogación
- \* "\\" --> backslash

### Constantes:

Al igual que las variables, las constantes guardan información, pero a diferencia de ellas, esta información no puede ser modificada.

#### Constantes Literales:

Una constante literal, es un valor ingresado directamente en tu programa, en cualquier lugar que lo necesite.

```
int miEdad = 23;
```

miEdad es una variable de tipo int (Entero); 39 es una constante literal. No puedes asignar un valor a 39 y su valor no puede cambiarse.

#### Constantes Simbólicas:

Una variable simbólica, es una variable que está representada por un nombre, tal como una variable está representada. A diferencia de una variable, sin embargo, su valor no puede ser modificado.

```
Estudiantes = clases * estudiantesporclase;
```

clases y estudiantesporclase son variables simbólicas. En cada lugar que el programa lea estudiantes, asignará el valor de clases \* estudiantesporclase.

### Definiendo constantes con "const":

Utilizar const es muy útil, ya que se le puede dar un tipo a las variable y volverla constante.

```
const unsigned short int estudiantesporclase = 15;
```

### Constantes Enumeradas:

Las constantes enumeradas, te permiten crear nuevos tipos y luego definir dichas variables con valores que están restringidos a un grupo de posibles variables.

```
enum COLOR {ROJO,AZUL,VERDE,BLANCO,NEGRO};
```

COLOR es el nombre de la enumeración, en otras palabras, un nuevo tipo. ROJO es una variable simbólica con el valor "0", AZUL="1", etc. Se pueden inicializar con el valor que uno elija. {ROJO=100,AZUL,VERDE=500};

Ejemplo:

```
#include<iostream.h>    //Librería para cout y cin
#include<conio.h>        //Librería para clrscr() y getch()

int main()
{
enum Dias {Domingo,Lunes,Martes,Miercoles,Jueves,Viernes,Sabado};
//El enum, es una variable que contiene multiples valores, identificados por un
//índice
int opcion;
cout<<"Ingresa un día (0-6): ";
cin>>opcion; //Leemos un valor ingresado por teclado
if(opcion == Domingo || opcion == Sabado)
//Si el número que hemos ingresado es igual al valor de los índices (0 -
//Domingo) o (6 - Sabado), ejecuta una acción
```

```
cout<<"Ya estás en el fin de semana"; //Mostramos un mensaje  
else //Si el número ingresado, esta fuera del rango (0-6)  
cout<<"\nTodavía sigues en clases"; //Mostramos un mensaje  
getch(); //Copiamos la pantalla  
}
```



# Expresiones y Secuencias

## Que son Secuencias:

En C++, una secuencia controla la ejecución, evalúa una expresión, o no hace nada.

```
x = a + b;
```

A diferencia del álgebra, en la cual x sería igual a "a + b", en C++, x obtendrá el valor de la suma de a + b.

## Bloques y Secuencias compuestas:

En cualquier lugar que puedas poner una secuencia, puedes poner una secuencia compuesta. Una secuencia compuesta, ejecuta una serie de expresiones. Comienza con una { y termina con una }.

```
{  
temp = a;    //En este ejemplo, al utilizar una variable temporal,  
a = b;       //podemos intercambiar el valor de A por el de B y el  
b = temp;    //valor de B por el de A.  
}
```

## Expresiones:

Una expresión debe devolver un valor, por lo tanto 3+2 devuelve el valor 5. Todas las expresiones son secuencias.

```
3.2 //Devuelve el valor 3.2
```

```
PI //Devuelve el valor 3.14
```

```
SegundosPorMinuto //Constante entera que devuelve el valor 60
```

## Operadores:

Un operador es un símbolo que obliga al compilador a tomar una acción.

### Operador de Asignación:

El operador de asignación provoca que la variable izquierda tome el o los valores de las variables derechas.

```
x = a + b;  
z = 10 - x;
```

### Operadores Aritméticos:

Existen cinco operadores matemáticos: suma(+), resta(-), multiplicación(\*), división(/) y módulo(%).

```
x = 5 + 5; //10  
y = 10 - 8; //2  
z = 6 * 4; //24
```

### División Entera y Módulo:

La división entera, toma solamente la parte entera de la división. La división de módulo, toma solamente el residuo de la división.

```
x = 10 / 5; //2  
y = 10 % 5; //0  
z = 6 % 4; //2
```

### Incremento y Decremento:

El operador incremento (++) eleva el número en un dígito. El operador decremento (--) lo reduce en un dígito.

```
c++;  
c = c + 1;  
//Son equivalentes
```

### Prefix y Postfix:

El prefix se escribe antes del nombre de la variable, e incrementa el valor y luego lo asigna. El postfix se escribe después de la variable, y asigna el valor y luego lo incrementa.

```
int a = ++x; //Prefix  
int b = x++; //Postfix
```

### Otro ejemplo:

```
#include <iostream.h> //Librería para cout y cin  
#include <conio.h>     //Librería para clrscr() y getch()  
  
int main()  
{  
    int miEdad = 23; //Declaración de variables  
    int tuEdad = 23;  
    cout<<"Yo tengo: "<<miEdad<<" años de edad.\n";  
    cout<<"Tu tienes: "<<tuEdad<<" años de edad.\n";  
    miEdad++; //Asignamos el valor y lo incrementamos en 1  
    ++tuEdad; //Incrementamos el valor en 1 y lo asignamos  
    cout<<"Un año pasó...\n";  
    cout<<"Yo tengo: "<<miEdad<<" años de edad.\n";  
    cout<<"Tu tienes: "<<tuEdad<<" años de edad.\n";  
    cout<<"Un año pasó...\n"; //Mostramos un mensaje  
    cout<<"Yo tengo: "<<miEdad++<<" años de edad.\n";  
    cout<<"Tu tienes: "<<++tuEdad<<" años de edad.\n";  
    cout<<"Escribamoslo de nuevo.\n"; cout<<"Yo tengo: "<<miEdad<<" años de  
    edad.\n";  
    cout<<"Tu tienes: "<<tuEdad<<" años de edad.\n";  
    getch(); //Copiamos la pantalla  
}
```

## Operadores Relacionales:

Son usados para determinar cuándo dos números son iguales o si uno es mayor que el otro:

Igualdad	==	100 == 50	→ falso
Desigualdad	!=	100 != 50	→ verdadero
Mayor que	>	100 > 50	→ verdadero
Mayor igual que	>=	100 >= 50	→ verdadero
Menor que	<	100 < 50	→ falso
Menor igual que	<=	100 <= 50	→ falso

## El operador IF:

Este operador examina una condición dada y ejecuta una acción determinada:

```
#include <iostream.h>    //Librería para cout y cin
#include <conio.h>        //Librería para clrscr() y getch()
int main()
{
    int miNum=5,tuNum;    //Declaración de variables
    cout<<"Ingresa un número:"; //Mostramos un mensaje en la pantalla
    cin>>tuNum;          //Leemos el valor ingresado por teclado
    if(tuNum > miNum)
    //Comparamos si el valor de la variable tuNum es mayor que el valor de miNum
        cout<<"\nGanaste, tu número es mayor que el mio";
    getch(); //Copiamos la pantalla
}
```

## Capítulo 7 - Secuencias de Control

### ¿Qué son las secuencias de control?

Las secuencias de control, son métodos o funciones, que nos permiten ejecutar una instrucción más de una vez, o por un número ilimitado de veces o hasta que se cumpla una condición dada, y que generalmente es matemática.

Para mi punto de vista, este es uno de los puntos más importantes, ya que casi todos o todos los lenguajes de programación los utilizan, aunque no por eso, significa que no sean fáciles de aprender y utilizar.

### While o Mientras

While o Mientras(En pseudo-código), permite ejecutar una instrucción o un conjunto de instrucciones mientras la condición dada sea verdadera.

```
#include <iostream.h> //Librería para cout y cin
#include <conio.h>      //Librería para clrscr() y getch()

int main()
{
    int counter = 0; //Declaramos una variable, para usar como contador

    while(counter < 5) //Mientras counter sea menor a 5
        // ejecuta la instrucción.
    {
        counter++; //Aumenta a counter en uno.
        cout<<"Hola\n"; //Imprimimos un mensaje
    }
    getch(); //Copia la pantalla
}
```

```

#include <iostream.h>    //Librería para cout y cin
#include <conio.h>        //Librería para clrscr() y getch()

int main()
{
    clrscr(); //Para borrar la pantalla.
    int num; //Declaramos una variable de tipo entero
    cout<<"Ingresa un número del 1 al 10: ";
    cin>>num; //Leemos el valor ingresado por teclado
    while(num>0 && num<11) //Mientras num sea mayor a 0 y menor que 11
    {
        cout<<"\nTu número es num: "<<num; //Imprimimos el mensaje
        num++; //Aumentamos a nuestro contador en 1
    }
    getch();
}

```

### Do...While o Hacer...Mientras

Do...While o Hacer...Mientras(En pseudo-código). Aunque es similar al While, el Do...While, tiene la ventaja de que ejecuta las instrucciones antes de verificar si la condición se cumple, por lo cual, siempre se ejecutará por lo menos una vez.

```

#include <iostream.h>    //Librería para cout y cin
#include <conio.h>        //Librería para clrscr() y getch()

int main()
{
    clrscr(); //Para borrar la pantalla.
    int num; //Declaramos una variable de tipo entero

    cout<<"Ingresa un número del 1 al 10: ";
    cin>>num; //Leemos el valor ingresado por el teclado
}

```

```

do //Hacerlo una vez
{
    cout<<"\nTu numero es num: "<<num;
    num++; //Aumentamos nuestro contador en 1
}while(num > 0 && num < 11); //Y verificar si num es mayor a 0
                                //y menor que 11, para seguir ejecutándolo
getch(); //Copia la pantalla
}

```

### For o Para

El For o Para(En pseudo-código), se utiliza para asignar un valor, verificar una condición y aumentar la variable, todo en una sola línea.

```

#include <iostream.h> //Librería para cout y cin
#include <conio.h>     //Librería para clrscr() y getch()

int main()
{
    clrscr(); //Para borrar la pantalla.
    int num;
    for(num = 0; num < 6; num++) //Para num = 0, mientras que num sea
//menor que 6, aumentamos num en 1
        cout<<"\nEl número es: "<<num; //Imprimimos el valor de num
    getch(); //Copia la pantalla
}

```

### Switch o Casos

El Switch o Casos(En pseudo-código), permite escoger entre una amplia gama de posibilidades, sin tener que ejecutar millones de líneas de if, else, etc.

Es muy útil a la hora de generar menues, que piden al usuario, escoger una opción.

```
#include <iostream.h>    //Librería para cout y cin
#include <conio.h>        //Librería para clrscr() y getch()

int main()
{
    clrscr();    //Limpia la pantalla
    int opcion; //Declaramos una variable de tipo entero
    do //Hacer
    {
        cout<<"1)\n";
        cout<<"2)\n";
        cout<<"3)\n";
        cout<<"Escoja una opción: ";
        cin>>opcion; //Leemos el valor ingresado por teclado
    }while(opcion<0 || opcion>4); //Ejecutamos el bloque mientras la
//opción sea menor que 0 o mayor que 4
    switch(opcion) //Verifica el valor de opción
    {
        case 1: //Si es uno
            cout<<"Eligio la opción 1";
            break; //El break sirve para que no ejecute
            //las demas opciones.
        case 2: //Si es dos
            cout<<"Eligio la opción 2";
            break;
        case 3:
            cout<<"Eligio la opción 3";
            break;
        default:
            cout<<"Esa no es una opción valida";
            break;
    }
    getch();} //Copia la pantalla
```