

Lenguaje de programación C++

Introducción

Temario

Sintaxis, Estructura de un programa, Ejecución, variables, flujo de función y funciones.

Clases y objetos. Métodos. Constructores y destructores.

Manejo de memoria. Punteros

Sintaxis

```
#include<iostream>

using namespace std;

int main(){
    cout << "Hola mundo!!";
    return 0;
}
```

C++ es como dice su nombre es un C aumentado.

Su sintaxis es similar, las funciones se delimitan con llaves { }

```
#include<iostream>
```

Es una directiva al compilador, incluye las libs con las funciones de entrada y salida estándar.

Luego vemos el main que debe ser una función única en el programa ya que será la buscada al ejecutarse. al igual que en C, retorna un entero y puede recibir argumentos por línea de comando

```
int main(int argc, char* argv[])
```

Sintaxis

Los comentarios también son como en C:

// para comentar en una línea

/* */ para comentar todo un bloque

```
int x = 0; // este es un comentario de línea
```

```
/* esto comenta un bloque
```

```
completo */
```

Sintaxis

```
#include<iostream>
```

```
using namespace std;
```

```
int main(){
```

```
    cout << "Hola mundo!!";
```

```
    return 0;
```

cout es la función que invoca la salida estándar (**s**tandard **c**haracter **o**utput)

<< es el operador de inserción de flujo.

inserta lo que se encuentra a la derecha en la salida estándar

Otra forma de escritura es como vemos en este ejemplo.

En vez de generalizar el uso del std al principio del archivo, lo indicamos en cada función que pertenece a este namespace.

Veremos mas adelante lo que es un namespace.

```
#include<iostream>
```

```
int main(){
```

```
    std::cout << "Hola mundo!!";
```

```
    return 0;
```

```
}
```

Compilación y Ejecución

La compilación y ejecución son tan simple como las de un programa de C.

```
$> g++ holaMundo.cpp
```

esto, si no hay errores, dejará un archivo **a.out** que podrá ser ejecutado.

Se pueden usar los comandos que ya conocen:

-o NombreArchivo (da el nombre del ejecutable)

-Wall para la los warnings (explicita los warnings como errores para poder corregirlos)

Dejaremos publicado un documento de gcc para complementar la lectura

Variables

Similar al C, C++ es fuertemente tipado. Las variables primitivas son similares, pero se agregan algunas diferencias, por ejemplo, existe el tipo booleano. También se distingue mayúsculas de minúsculas.

Por ejemplo podemos usar dos modos para declarar e inicializar:

```
int a = 0; // siempre se recomienda inicializar al declarar.
```

```
int a(0); // escritura que inicializa como si el entero fuera un objeto.
```

Variables

Resumiendo tenemos las siguientes variables:

char, int, float, double, bool y void

Además se pueden anteponer definiciones como ***long int, unsigned short int, signed int.***

Los conceptos son iguales a los vistos con anterioridad para C.

El rango del valor entero irá variando según los modificadores, por ejemplo

el rango del unsigned va de 0 and 65535, y el signed entre -32768 y 32767.

Flujo de control

if, else, while, do - while, switch son idénticos al uso que ya conocen, no agregaremos ni profundizaremos sobre este tema.

Ocurre lo mismo con la escritura del ***for***.

Funciones

Como en C, una función tiene que estar declarada antes de ser usada, o por lo menos el prototipo debe estar en el inicio del archivo o en .h que se incluye.

Admite que los retornos, los nombres, los parámetros sean diferentes. Un poco mas adelante, veremos conceptos como sobrecarga de funciones o funciones polimórficas, esto último permite lo siguiente por ejemplo:

int maximo (int, int);

float maximo (float, float);