Iostream - input output stream

string

cout <<

cin>>

salto de linea ----- endl

pasos de un compilador

1. sintaxis

2. codigo intermedio

3. codigo ejecutable / link

4. Archivo ejecutable

para las librerias

- hay un namespace y adentro hay objetos

- el namespace es iostream es STD

\* para meter el namespace en la librería se usa, using namespace std;

--------------- Clase 2.---------------------

getline lee toda la linea, el cin solo hasta un espacio

//ejemplo

// Example program

#include <iostream>

#include <string>

int main()

{

std::string name;

std::cout << "What is your name? ";

getline (std::cin, name);

std::cout << "Hello, " << name << "!\n";

}

//ffin

Ejemplo uso de for y numeros y explicacion memeoria buffer

/////////////////////

// Example program

#include <iostream>

#include <string>

using namespace std;

int main()

{

int array[5];

cout << "escriba el valor de cinco enteros:"<< endl;

for(int i=0; i<5; i++){

cout<< "Ingrese el entero "<< i <<" : ";

cin >> array[i];

}

cout << "Los enteros leidos son: ";

for(int i; i<5;i++){

cout << array[i]<< " ";

}

return 0;

}

/////////////////

Enteros

int,long,short,byte

Otros los considera como enteros

char,bool

Editero de texto descargar: \*Vim

//ejemplo de char y bool como enteero

// Example program

#include <iostream>

#include <string>

using namespace std;

int main()

{

//Casteo estático

int num = 'a';

char letra =70;

cout << "Num: "<< num<< endl

<< "Char: " << letra << endl; //toma valores de la tabla ascii

bool condition = 0; // si es distinto de 0 es verdadera,

// el 0 es el mismo falso,

// en bool

if(condition ) // compara pones ==,

//si solo se pone un "=" lo toma como asigancion

cout << "Verdadero";

else

cout<<"Falso";

return 0;

}

////////////////////////////////////////////

Método: { procedimiento, función]

//Uso de funciones

// Example program

#include <iostream>

#include <string>

using namespace std;

//documentar siempre \*

int suma(int,int);

//Suma de dos enteros que recibe de parametro

//LOs metodos solo se ponene arriba del main \*funcion

//A menos que usae prototipo

int main()

{

cout << "El resultado es : "<< suma (10,45) << endl;

return 0;

}

int suma(int a, int b){

return a+b;

}//Fin metodo especificamnete funcion

//

///////Fin clase 2///////

//Clase 3/////////////////////7

Apuntadores

Cuando la variable esta instanciada tiene una direccion en memoria

tiene valor en tiempo de ejecucion

→Direcciones en memoria

Apuntador es un tipo de variable tiene:

nombre

dirección

\*\* El valor de la variable es una dirección

ArrayList <int> lista \*\*\*espacio en memoria llamado lisra

lista = new ArayList<int>; \*\*\*++ crea otra direccion

el primer espacio almacena la direccion de la ++Direccion instanciada

\*\*\* si no hay include no se pude utilizar la palabra NULL pero puede asignar cero al apuntador

////////////ejemplo/////////////  
  
// Example program

#include <iostream>

#include <string>

using namespace std;

int main()

{

//Declarar apuntador

int\* intPointer = NULL;

//instanciar un arreglo de 5 ints

intPointer = new int[5];

cout <<

"la direccion de la variable es: "

<<&intPointer<<endl

<<"El contenido de la varibale es: "

<<intPointer<<endl;

//liberar memoria

delete[] intPointer; //Se elimina por el espacio de memoria

/\*Si se instancia un arreglo se elimina un arreglo\*/

//Buena practica

//Asignar null

intPointer=0;

//Liberar algo liberado da error

return 0;

}

///////////////////////////

//////////////////////Clase 4///////////////////////////777

VIM

#include <iostream>

using namespace std; //Busar como cambiar cantidad de espacio de tab, gedit

//Se guarda con :w,,, insert, se sale con q

//para compilar se usa g++ helloWolrd.cpp -o ejemplo1

//para ejecutar ./ejemplo1

int main(){

return 0;

}

/////////////////////Fin clase 4////////////////////////////777

//Clase 5////////////////////////////