Structured programing

Class :First Class المرحلة الاولى

Lecture : fourth المحاضرة الرابعه

Lecturer : Maha Ali Hussain

م. مها علي حسين

□□ □ □□ □□ / □□ □□ □□□ □ □□□ □□ □

2021

Not:

- The variable list may consist of one or more identifier names separated by commas. Some **valid declarations are shown here**:

Example:

```
int    i, j, k;
char   c, ch;
float  f, salary;
double d;
```

The line **int i, j, k;** both declares and defines the variables i, j and k; which instructs the compiler to create variables named i, j and k of type int.

- Variables can be initialized (assigned an initial value) in their declaration. The initializer consists of an **equal sign** followed by a **constant expression** as follows:

Example:

```
int    d = 3, f = 5;  // definition and initializing d and f.
float  A1 = 2.2;      // definition and initializes 2.2
char   x = 'x';      // the variable x has the value 'x'.
```

4.1 C++ Program Structure

Let us look at a simple code that would print the words Hello World.

```
#include <iostream>
using namespace std;
int main()      // main() is where program execution begins.
{
C++ statements;
……..
…..
return 0;
}
```

1. The C++ language defines several headers, which contain information that is either necessary or useful to your program. For this program, the header **<iostream>** is needed for output string in the screen.
2. **int main()** : is the main function where program execution begins.
3. // : is a single-line comment available in C++. Single-line comments begin with // and stop at the end of the line.
4. **cout << " : This is my first C++ program.";** causes the message "This is my first C++ program" to be displayed on the screen.
5. << : it is the send operator
6. **return 0**: terminates main() function and causes it to return the value 0 to the calling process.
7. **;** : semicolon , its used as terminator for every C++ statement.

❖ The **OUTOUT** for this program is :

```
Hello World
```

4.2 Standard Output (cout)

cout: the standard output of a program is the screen, and the C++ stream object defined to access it is cout. The < < **operator** is overloaded to output data items of built-in types integer, float، double, strings and pointer values.

Example:

```
cout << "Output sentence";    // prints Output sentence on screen
cout << 120;                  // prints number 120 on screen
cout << x;                    // prints the content of x on screen
```

4.3 Standard input (cin)

cin: is the input stream object, its read the input value from keyboard.
>> : it is the operator use to get from operator.
endl : is used to add a new-line at the end of the line.

- You can also use cin to request more than one datum input from the user:

cin >> a >> b;

is equivalent to:

cin >> a;
cin >> b;

- In both cases the user must give two data, one for variable a and another one for variable b that may be separated by any valid blank separator: a space, a tab character or a newline.

Example:

```
#include <iostream>
using namespace std;

int main()
{
char name;
cout << "Please enter your name: ";
cin >> name;
cout << "Your name is: " << name << endl;
return 0;
}
```

```
#include <iostream>
using namespace std;

int main()
{
 cout << "This is a sentence,";
cout << "This is another sentence.";
return 0;
```

following the other **without any line break between them**.

This is a sentence,This is another sentence.

Example:

```
#include <iostream>
using namespace std;


int main ()
{
int i;
cout << "Please enter an integer value: ";
cin >> i;
cout << "The value you entered is  " << i;
cout << " and its double is " << i*2 << endl;
return 0;
}
```

❖ The **OUTPUT** for this program: will be shown on the screen:

Please enter an integer value: 702
The value you entered is 702 and its double is 1404.