TP 3: Variables and constants

1. Declaring variables

The basic types used in the C++ language are: the "***int''***(for integer), the***''float''***(for real),***''bool''*****(**to represent a Boolean variable) and the ***''char''*** (for characters).

Syntax:

- To declare a variable in C++, we use the following syntax:

Type *Name-variable1, variable-name2, ... ;*

Example :

int A, C;
char Ch;
string First_name;
float tx1, x2;
bool B;

- At the same time as declaring a variable, it is possible to assign it an initial value.

Example :

int a=34; or int a (34);
float x=10.5; or float x (10.5);

1.1.The integer type (int):

- The C++ language has several basic types to designate an integer.
- ✓ **int**: contains a normal-sized integer, positive or negative.
- ✓ **short int**: contains a small integer, positive or negative.
- ✓ **long int**: contains a large integer (32 bits), positive or negative.
- ✓ **long long int**: contains a larger integer (64 bits), positive or negative.
- ✓ **unsigned int**: contains a normal-sized integer, positive or zero.
- ✓ **unsigned short int**: contains a small integer, positive or zero.
- ✓ **unsigned long int**: contains a large integer (32 bits), positive or zero.
- ✓ **unsigned long long int**: contains a larger integer (64 bits), positive or zero.

1.2. The real type

- To represent a real, there are 3 basic types:
- ✓ **float** (simple precision) number in 32 bits (one bit for the sign)
- ✓ **double** (double precision) ; number in 64 bits (one bit for the sign)
- ✓ **long double** (extended precision)

1.3. The character type (char)

- to represent a character, this character can be a number, example '**3**', or a letter, example '**s**', or a special character, example '**?**'.

1.4. The boolean type (bool)

- This type is formed from two values : **true** and **false**.

Noticed :

Character strings (***string***), is not an elementary (simple) type.
To use it, you must place the file header:***# include <string>***

2. Display (Writing) on screen

Displaying the value of an expression on the screen

Syntax**:**

cost << *expression***;**

- More generally we can write:

Cost << *expression1*<< *expression2*<<…. **;**

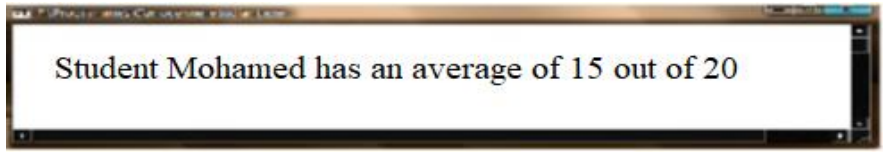
Examples:

- Several variables can be displayed at the same time as follows:

cout << ***A***<< ***x1***<< ***Ch***<< ***name;***

➢we can also display a message at the same time as a variable as follows:

cout<< "Student" <<name<< "has an average of " <<avg<< " out of 20";



Mohamed: this is the value of the variable **name**
15: this is the value of the variable **avg**

3. Reading a variable

Syntax:

cin>> *Variable-name***;**

This instruction reads (from the keyboard) a value and assigns it to the variable.

- More generally we can write:

cin >> var1>>. . . >>varn;

Examples:

cin >>A ;

cin >> x1 >> ch >> name;

Practical work 1:

Write 3 programs: the first reads and displays a real, the second reads and Displays a character and the third reads and displays a string.

Noticed :

Function ***getchar();*** interrupts program execution until the user presses a key on the keyboard.
Delete the function ***getchar();*** and run the program again?

4. Constants

Constants do not change values during program execution unlike variables.
To define a constant in the C++ language we use the keyword ***const*** or the key word ***#define.***

Syntax:
ConstType *Name-constant = value;*
Examples:
const int A=5;
const float Pi=3.14;
const char Ch='a';
Noticed :
- We cannot read a constant with the cin>> function and assign a new value to it since the value of the constant does not change during the execution of the program.
- Constants are displayed in the same way as variables.

5. Comments:

You can add comments in the source code of a program using the symbol /* */or the symbol //:

- We start with slash / followed by a star * and end with a star * followed by a slash: /* several lines */
- We start with a double slash for a comment that spans a single line : // Single line

TP 2:
Write a C++ program that asks the user for a student's last name, first name, group, algorithmic grade, and algebra grade and then displays the average. Use comments and messages.
The execution of the program is as follows: