short int --> 2 bytes (16 bits)

int --> 4 bytes (32 bits)

long long int --> 8 bytes (64 bits)

float --> 4 bytes (32 bits)

double --> 8 bytes (64 bits)

char --> 1 bytes (8 bits)

Literals are the integers, floating point numbers or expressions which are not stored in any variable.

For Example:

cout << 45;

In this example 45 is a literal as it is not stored in any variable.

If the literal represents an integer value ie (45 , 78 , 99) the default datatype of the literal will be int.

If the literal represents an floating point value ie (45.8 , 78.54 , 99.374) the default datatype of the literal will be double.

Constant Identifier 🡪 Can’t be changed.

Non - Constant Identifier 🡪 Can be changed.

L Value 🡪 can be placed on left side of the assignment operator (=)

R Value 🡪 can’t be placed on left side of the assignment operator (=)

***Formula for random number using rand():***

lowest number + rand() % (highest number – lowest number).

***Type Casting:***

We can change the data type of the variables explicitly for example the variable has a data type of integer we can convert in into float by type casting.

Int x = 30;

(float)x 🡪 It converts the data type into float but the original data type of x will remain int. It will only change its data type only at this line or for this expression.

***Conversion:***

034 --> for displaying octal number system.

or

cout << oct << 45; 🡪 For this use #include<iomanip>

0x347 --> for displaying hexa decimal number system.

0b00101 --> for displaying binary number system.

or

cout << bin << 45; 🡪 For this use #include<iomanip>

***strcpy;***

// strcppy is used to change the array even after it is declared. It has two arguments (first argument is the variable(array) in which we want to change data or redeclare it, the second argument is actual data that will be changed.)

#define \_CRT\_SECURE\_NO\_WARNINGS have to use it in order to use strcpy ie strcpy displays warning it will ignore those warnings.

***strcmp(s, t);***

This is used to compare two strings. It has two arguments (first will contain a variable, second will also contain a variable) and then compare both. It will generate 0 when both are equal. It will generate negative value (in most compilers -1) if the second value is larger and will return positive value (in most compilers 1) when first value is larger.

***Buffer clearing methods using loop...***

This program is used to clear the buffer until it is not good (or we have the desired data.)

int a;

cin >> a;

while (!cin.good())

{

cin.clear();

cin.ignore();

cin >> a;

}

***For Loop:***

This is the counter clock loop it will be used when we know in advance how many times the loop will be generated.

for (int i = 0; i < 10; i++)

{

cout << num << " \* " << i << " = " << num \* i << endl;

}

***Do while Loop:***

It will be used when we have to run the loop one time irrespective of the condition and for next iterations it checks the condition.

do

{

cout << "Hello " << endl;

cin >> choice;

} while (choice == 'y');