C++ Character Set

a-z, A-Z, 0-9, and underscore (_)

C++ is case sensitive language, meaning that the variable first_value, First_Value or FIRST_VALUE will be treated as different .

Identifier and Keywords:

Identifiers in C++ are used to give names to various elements in a program like variable, constants, functions and arrays etc. One must give the name to an identifier as per its usage.

Keywords are identifiers that are reserved words in C++ and have specific purpose. These reserved cannot be used by the user as user defined identifiers. C++ has 34 keywords.

Escape Sequence in C++

An escape sequence is a sequence of characters that does not represent itself when used inside a character or string literal, but is translated into another character or a sequence of characters that may be difficult or impossible to represent directly.

In C/C++, all escape sequences consist of two or more characters, the first of which is the backslash, \; the remaining characters determine the interpretation of the escape sequence. The escape sequence are given below:

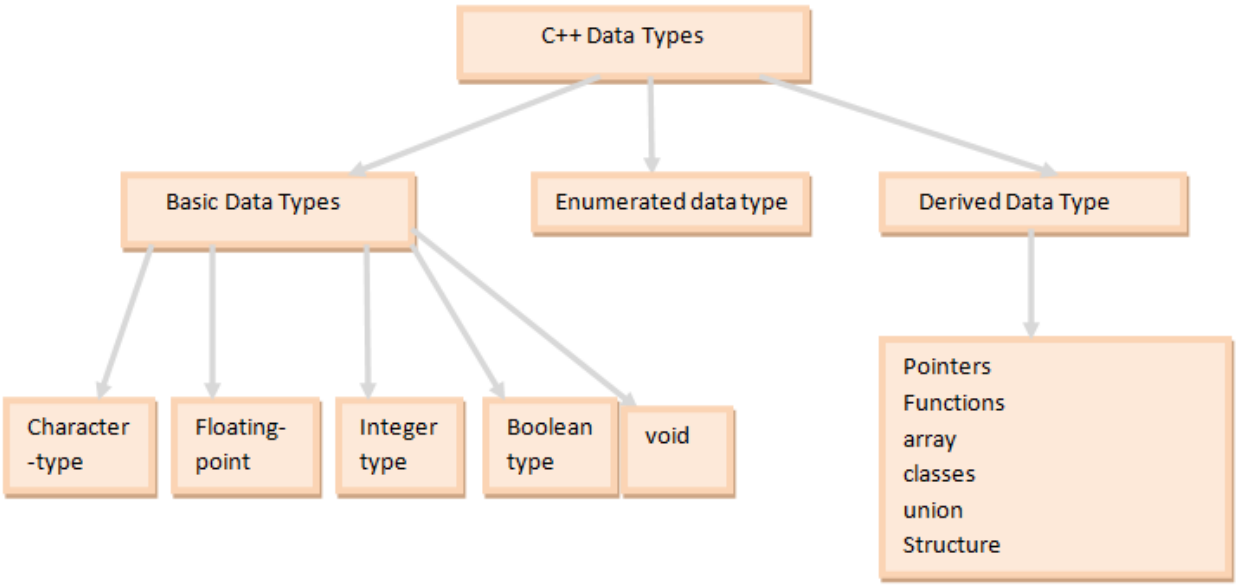
Escape sequence	Character	Action
\a	Bell/beep	Sound a beep
\b	backspace	Moves cursor one back space
\f	formfeed	Next line
\n	newline	Next line
\t	tab	Horizontal tab
\r	return	Return
\ \	backslash	For comment statement
\’	Single quote	
\”	Double quote	
‘\xdd’	hexadecimal	Base 16
‘\o’	Null character	
‘\?’	Question mark	
‘\xhhh’	Hexa decimal value such as \x1b	
‘\ooo’	Octal value such as \ o33	

The escape can be used in cout statement within the “ “ as

cout<<”\n enter a number”; //This statement will print enter a number on new line

Data Types in C++

Data type in a programming language is used to assign a type of value to a variable from the range of the values supported by that type. For example a data type char can be used to assign any one of the character set defined in C++ to a variable. They are of the following types:



Data types in C++			
Data Type	**Memory (ByteS)**	**Minimum Value**	**Maximum Value**
Bool	1	Logical Value T/F	Logical Value T/F
Char	1	-128	127
Unsigned Char	1	0	255
Short	2	-32768	32767
Unsigned Short	2	0	65535
Int	2	-32768	32767
unsigned int	2	0	65535
Long	4	-2147483648	2147483647
unsigned long	4	0	4294967295
Float	4	10^{-38}	10^{38}
Double	8	10^{-308}	10^{308}
long double	10	10^{-4932}	10^{4932}

Token in C++

Keywords: have their meaning predefined in C++. They cannot be used by programmer for any other purpose.

Identifiers: Are the name of the user defined variables, arrays, functions etc

Variables:

Constants: Are whose value cannot change during the execution of the program.

Constants in C++

Constants in c++ are the data elements which do not change throughout the execution of the program. Constants can be of following type:

Integer constant : 7, 3, 10, 345 etc

Character constant : 'A', 'Z' , 'x' …… etc

Float constant : 0.2, 0.34, 2.35……

String constant : "RAM", "XYZ"……

Expression Vs. Statement

Expression	Statement
X=2+3	X=2+3;
Not terminated by semicolon (;)	Always terminated by semicolon (;)

Operators in C++

An operator in C++ is used to perform an operation on the operand(s) and part of an expression. For example c= a+b; here there are two operators; one the "+" arithmetica addition operates which evaluates the sum of a and b, and the other is "=" equivality operator which is used to assign the value to the left hand side operand. Operators in C++ are of the following types.

1. Assignment Operator
2. Mathematical Operators
3. Relational Operators
4. Logical Operators
5. Bitwise Operators
6. Shift Operators
7. Unary Operators
8. Ternary Operator
9. Comma Operator

Arithmetic Operators		+, -, *, /, %
Assignment Operators		=, +=, -=, *=, %=, /=, >>=, <<=, &=, |=, ^=
Relational Operators		>, >=, <, <=, !=, ==
logical operators	AND (&&), OR (||), NOT(!) Used to combine two operands, for AND validity of both statements will be considered, for OR(||) validity of 2nd statement is considered if 1st statement is invalid	These operators are mostly used in loops (especially while loop) and in Decision making.
Bit wise logical operators	and, or, xor, not	&, |, ^, ~
Shift Operator	Right Shift, left Shift, unsigned right shift	>>, <<, >>>>
Unary Operator	increment, decrement, Address of, dereference, unary -, bitwise NOT, logical NOT, **new**, **delete**	++, --, &, *, -, &, ~, !
Ternary		If-else ? : Example int a = 10; a > 5 ? cout << "true" : cout << "false"
Comma Operator	Used to separate : variable names and expressions	int a,b,c; // variables declaration using comma operator

		a=b++, c++; // a = c++ will be done.

Operator Precedence

Unary Operator	Sizeof(type)	++	-				L to R	Highest
Arithmetic Multiply, divide and remainder	*	/	%				L to R	
Arithmetic Add, Sub	+	-					L to R	
Relational Operator	<	<=	>	>=			L to R	
Equality	==	!=					L to R	
Logical and	&&	||					L to R	
Conditional	?:						L to R	
Assignment Operator	%=	/+=	*=	- =	+ =	=	L to R	Lowest

Z = -10/5 – 20 + 8

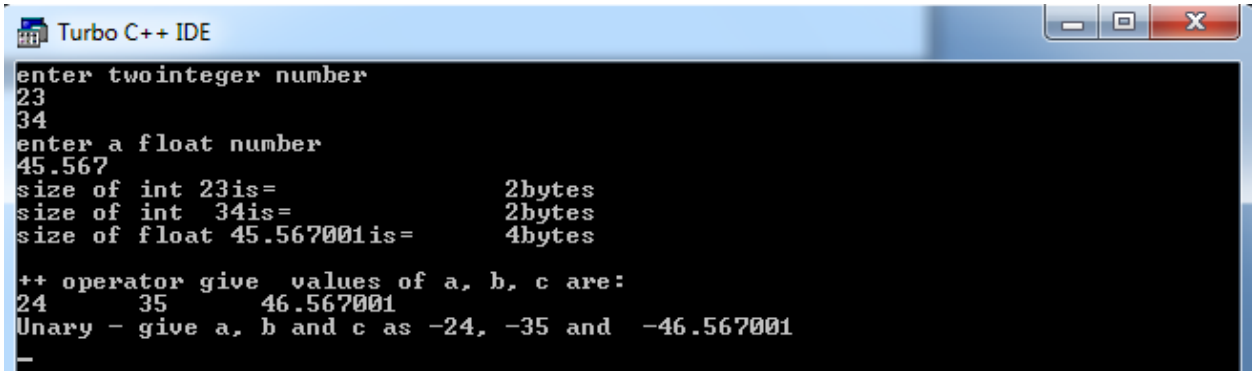
= -2 -22 +8 = -16

Use of Operator

(a) Unary Operators : Sizeof(type), ++ and –

```
#include<iostream.h>
#include<conio.h>
void main()
{
   clrscr();
   int a,b;
   float c;
   cout<<"enter twointeger number"<<endl;
   cin>>a>>b;
   cout<<"enter a float number"<<endl;
```

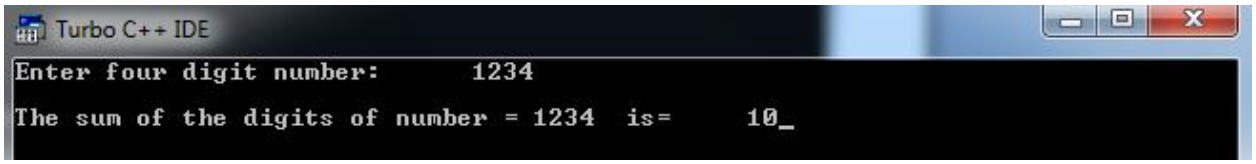
```
    cin>>c;
    cout<<"size of int "<<a<<"is=\t\t"<<sizeof(a)<<"bytes"<<endl;
    cout<<"size of int  "<<b<<"is=\t\t"<<sizeof(b)<<"bytes"<<endl;
    cout<<"size of float "<<c<<"is=\t"<<sizeof(c)<<"bytes"<<endl;
    a++;
    b++;
    c++;
    cout<<endl<<"++ operator give  values of a, b, c are:"<<endl<<a<<"\t"<<b<<"\t"<<c<<endl;
    cout<<"Unary - give a, b and c as "<<-a<<",\t"<<-b<<" and\t "<<-c<<endl;
    getch();
}
```



(b) Arithmetic Multiply(*), Divide(/), Mod(%), Add(+) and Sub(-) Operator

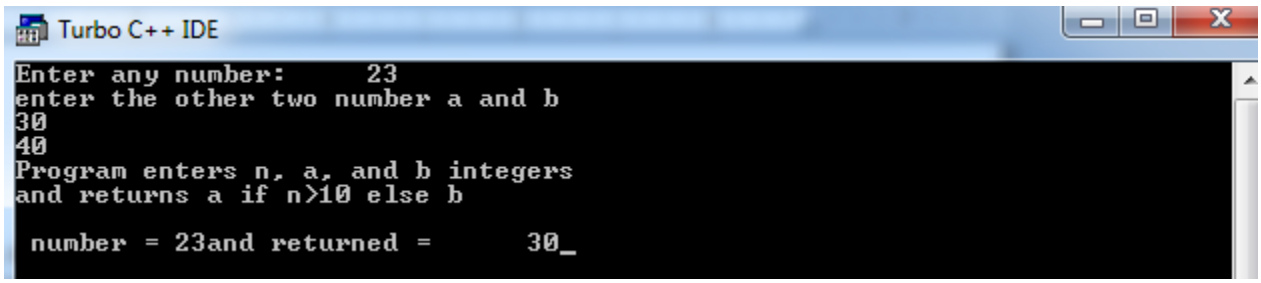
```
#include<iostream.h>
#include<conio.h>
void main()
{
        clrscr();
        int temp,a,b,c,d,n,sum;
        cout<<"Enter four digit number:\t";
        cin>>n;
        temp=n;
        a=n%10;
        n=n/10;
        b=n%10;
        n=n/10;
        n=n/10;
        d=n%10;
```

```
        sum=d+c+b+a;
        cout<<"\nThe sum of the digits of number = "<<temp<<"  is=\t"<<sum;
        getch();
}
```



(c) Ternary Operator: A ternary operator is a conditional operator. If a particular condition is met then the first value after the ternary operator is returned other wise the second value after the “,” is returned.

```
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
void main()
{
    clrscr();
    int a,b,n,c;
    cout <"\nProgram enters n, a, and b integers\n";
    cout<<"and returns a if n>10 else b"<<endl;
    cout<<"Enter any number:\t";
    cin>>n>>a>>b;
    c= n > 10 ? a : b;
    cout<<"\n number = "<<n<<"and returned =\t"<<c;
    getch();
}
```



Bit wise logical operators (AND(&), OR(|), XOR(^), NOT(~))

```
#include<iostream.h>
#include<conio.h>
#include<iomanip.h>
void main()
{
        clrscr();
        int a,b,c,d,e,f;
        cout<<"enter two numbers\n";
        cin>>a>>b;
        c= a & b;
        d= a | b;
        e=a ^ b;
        f= ~a;
        cout<<"\n bitwise and(&) of"<<a <<" and "<<b<<" is = "<<c<<endl;
        cout<<"\n bitwise or (|) of "<<a <<" and "<<b<<" is = "<<d<<endl;
        cout<<"\n bitwise xor (^) of"<<a <<" and "<<b<< " is = "<<e<<endl;
        cout<<"\n bitwise NOT (~) of"<<a << " is = "<<f<<endl;
        getch();

}
```