

>>	Shift Right	A>>2;//shift right by 2 places
&=	Bitwise AND assign	A &= B;// A=A&B
=	Bitwise OR assign	A = B; // A=A B
^=	Bitwise XOR assign	A ^= B; // A=A^B
<<=	Bitwise shift left assign	A <<= 2;//Shift left by 2 places and assign to A
>>=	Bitwise shift right assign	A >>= 1;//Shift right by 1 place and assign to A

Example

<pre>#include<iostream.h> void main() { int A=42, B=12, C=24, D; D = A^B; C <<= 1; A <<=2; B >>=2 ; cout<<"A="<<A<<endl; cout<<"B="<<B<<endl; cout<<"C="<<C<<endl; cout<<"D="<<D; } </pre> <p>The expected outputs are given below :</p> <p>A = 168 B = 3 C = 48 D = 38</p>	<pre>#include<iostream.h> void main() { int A =20, D , E, F; int B = 18, C = 30; D = C^B; E = A &B ; F = C A ; cout <<"E="<<E<<endl; cout <<"F="<<F<<endl; cout <<"D="<<D<<endl; } </pre> <p>The expected outputs are given below :</p> <p>E = 16 F = 30 D = 12</p>
---	---

7. Increment and Decrement Operators

They are used for increasing and decreasing a variable by unity. These operators may be placed before or after the variable name as shown in the following table.

Operator	Pre or post	Description
++k	Pre-increment	First increase the value of k by one then evaluate the current statement by taking incremented value.

k++	Post-increment	First use the current value of k to evaluate the current statements then increase k by unity.
--k	Pre-decrement	First decrease the value of k by unity then evaluate the statement.
k--	Post-decrement	First use the current value of k to evaluate the current statements then decrease k by unity.

Pre-increment	Post-increment
<pre>int a= 10, b= 11, c; c=a+ ++b; //c=10+12=22, b=12</pre>	<pre>int a= 10, b= 11, c; c=a+ b++; //c=10+11=21 , b=12</pre>

The following program illustrates the increment and decrement operators.

```
#include <iostream.h>
void main()
{
    int a = 6, p = 4, r=3, n =5, A,B,C,K ;
    A = 6 * ++n ;
    cout << "A = " << A << "\t n = " << n << endl;
    K = 5 * a-- ;
    cout << "K = " << K << "\t a = " << a << endl;
    B = r++ * r++ ;
    cout << "B = " << B << "\t r = " << r << endl;
    C = p-- * p-- ;
    cout << "C = " << C << "\t p = " << p << endl;
}
```

The expected output is given below:

```
A = 36    n = 6
K = 30    a = 5
B = 9     r = 5
C = 16    p = 2
```

H.W.

Evaluate the following expressions for $m = 6, n = 2, a = 0, b = 0, c = 0, d = 0,$ and $e = 0$

- (i) $a += 4 + ++m * n ;$
- (ii) $b * = 3 + --m * m ;$
- (iii) $c += 2 + m * ++m ;$
- (iv) $d * = 2 * + m * m -- ;$
- (v) $e -- = 2 * ++m / m -- ;$

PRECEDENCE OF OPERATORS

Precedence is an important aspect of operators. A list of operators and their precedence are given in the following table:

Operator	Description
()	Highest precedence given to parentheses. In case of several pairs of parentheses without nesting, they are evaluated from left to right.
++ and --	Increment and decrement operators have the precedence over the other arithmetic operators. In case of several such operators, they are evaluated from left to right.
*, /, %	Multiplication, division and remainder operators are evaluated after increment and decrement operators. In case of several such operators, they are evaluated from left to right.
+, -	Plus and minus are evaluated last. If there are several such operators then they are evaluated from left to right.

Example

Evaluate the value stored in the variable 'result' in the following expressions using precedence of operators if a=5, b=20, c=10, d=4, e=7;

- (a) result = a * b + c - d % e;
- (b) result = a * ++b + c - d % e;
- (c) result = ++a * b + (c - d) % --e;
- (d) result = a * (b + c) - ++d % e;

Solution

- (a) result = 5 * 20 + 10 - 4 % 7 = 110
- (b) result = 5 * ++20 + -- 10 - 4 % 7 = 114 , b = 21, c = 9
- (c) result = 5++ *20 + (10 - 4) % --7 = 101 , a=6 , e=6
- (d) result = ++5 * (20 + 10) - ++4 % 7 = 180 , a=6 , d=5

H.W.

Evaluate the following expressions if y=2:

- a) Z = 5 *y + 3*y*(10*y + 5/2) ;
- b) Z = 7* y % 2 + 2*(3 +(y % 3 + 2)) ;
- c) Z = 7 % y *(y + 6*7) -5;
- d) Z = 2* ++y + 3 * --y;

The sizeof() operator

This operator accepts one parameter, which can be either a type or a variable itself and returns the size in bytes of that type or object :

```
int x;
int y=sizeof(x);
a = sizeof (char);
```

Most of the mathematical functions are declared in the `<math.h>` header file, as shown in the table below.

Function	Description	Example
<i>sin(x)</i>	sine of x (x in radians)	sin(2) returns 0.909297
<i>cos(x)</i>	cosine of x (x in radians)	cos (2) returns -0.416147
<i>tan(x)</i>	tangent of x (x in radians)	tan(2) returns -2.18504
<i>asin(x)</i>	inverse sine of x (x in radians)	asin(0.2) returns 0.201358
<i>acos(x)</i>	inverse cosine of x (x in radians)	acos(0.2) returns 1.36944
<i>atan(x)</i>	inverse tangent of x (x in radians)	atan(0.2) returns 0.197396
<i>exp(x)</i>	exponential of x (base e)	exp(2) returns 7.38906
<i>fabs(x)</i>	absolute value of x	fabs(-2) returns 2.0
<i>log(x)</i>	natural logarithm of x (base e)[Ln(x)]	log(2) returns 0.693147
<i>log10(x)</i>	common logarithm of x (base 10)	Log10(2) returns 0.30103
<i>sqrt(x)</i>	square root of x	sqrt(2) returns 1.41421
<i>pow(x,p)</i>	x to the power p	pow(2,3) returns 8.0
<i>ceil(x)</i>	ceiling of x (rounds up)	ceil(3.141593) returns 4.0
<i>floor(x)</i>	floor of x (rounds down)	floor(3.141593) returns 3.0

Ex: Write a program to evaluate the area and circumference of a circle.

```
#include<iostream.h>
#define PI 3.14159
void main( )
{
    float radius, area, circum;
    cout<<"enter the radius of the circle"<<endl;
    cin>>radius;
```

```
    area=PI*radius*radius;
    circum=2*PI*radius;
    cout<<"area="<<area<<endl;
    cout<<"circumference="<<circum;
}

```

Ex: 6Ω , 3Ω resistors are connected in series across a 36v source, write a program to find the total current and the voltage of each resistor.

```
#include<iostream.h>
void main( )
{
    float R1, R2, V,I,V1,V2;
    cin>> R1>> R2>> V;
    I=V/(R1+R2);
    V1=I*R1;
    V2=I*R2;
    cout<<"I="<<I<<endl;
    cout<<"V1="<<V1<<"  V2="<<V2;
}

```

H.W

- 1- Write a C++ program to read the grades of four subjects (M1, M2, M3, M4), then calculate and print the average.
- 2- Write a simple program to print your name, class.
- 3- Write C++ program to calculate sum of two numbers.