

Department Of Computer Application
Dr. Rakesh Ranjan

BCA – Part III
C++ PROGRAMMING LANGUAGE

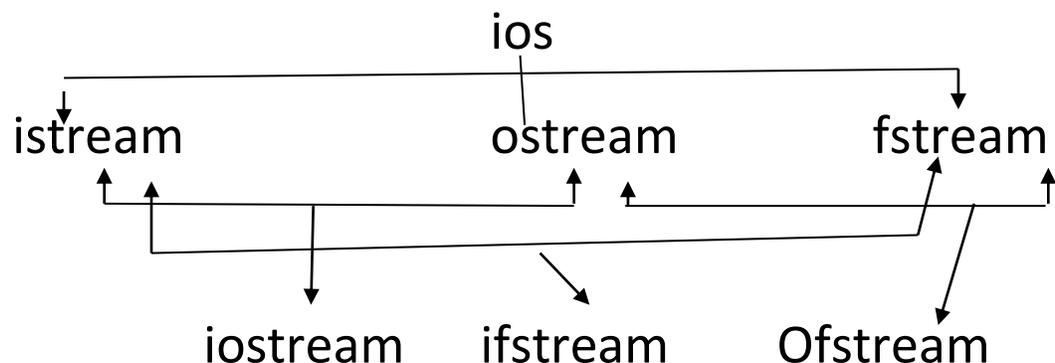
Input/ Output statement in c++ →

Statement -> single line instruction in program terminated by semicolon is known as statement

Compound statement → multiple line instruction bonded within { } curly brass is known as compound statement .

In C++ , “ios” is the base class which is derive in different classes to provide input/output functionality.

The class hierarchy may be shown as :-



iostream is the base class that provide functionality of input /output in c++ and we can add in our program through header file iostream.h

input class → I

n iostream.h we have a class “cin” that provide input function .

cin >> → it is shift operator overloaded function that is used to read all type of basic data type from keyboard and store in variables.

```
int a;
```

```
float b ;
```

```
double c
```

```
char d;
```

```
cin>> a>>b>>c>>d;
```

cin.get()-> this function is used to read a single character from keyboard.

```
cin.get(c );
```

we may read multiple character as

```
char p,q,r,s;
```

```
cin.get(p).get(q).get( r ).get(s);
```

cin.getline()-> it read multiple character as string up to given size .

```
char name[50];
```

```
cin.getline(name,50);
```

output in c++→

“cout” is the output class defined in iostream.h that provide functionality to display output on screen.

Cout<< → shift operator overloaded function to display all type of data on the screen .

Syntax -> cout<<” message”<< var <<”message ...”<< var;

```
Int a;
```

```
Float b;
```

Cout<< "enter any two data one in integer and other in float";

Cin>> a>> b;

Cout<<" value of A = "<< a<<" value of B = "<<b;

Some method of cout class →

Write() → to display one character on screen

Cout.write(c).write(d);

Cout.writeline() -> to display complete string on the screen

Cout.writeline(string , size of string);

Q-> WAP in c++, to read two data from keyboard and print its addition and subtraction. Finally display result on the screen.

```
# include <iosream.h>
```

```
void main()
```

```
{ int f1=0, f2=0, f3=0;
```

```
cout<<" Enter any two integer data →";
```

```
cin>> f1>>f2;
```

```
f3=f1+f2;
```

```
cout<<" Addition of given two data➔"<< f3;
```

```
f3=f1-f2;
```

```
cout<<" Subtraction of given two data➔"<<f3; return  
; }
```

Note : 1. All the reserve word / keyword are in small case . c++ is case sensitive programming language.
(Small letter and capital letter are different ➔ a or A
➔ different)

2.. In c++ we can declare variable any where in the program before their use.

HW-> 1 . WAP to calculate Simple Interest of given P,r,t and print;

2.. WAP to convert Fahrenheit temperature in celcius

3. WAP to read distance in Kilometer and Print in Meter, Centimeter and millimeter.

Meter= kilometer * 1000;

Cm = meter * 100

Mm= cm * 10

Escape sequence character →

it is special character followed by \ symbol , used in output statement to control the data display .

followings are escape sequence character used in cout<< statement.

\n → for new line

\t→ tab space (8 space ahead)

\b→ back space

\a→ bell character

\f→ formfeed (one character back without erasing the character

\\--> one \ printing

\” → double quote printing

\’ → single quote printing.

Eq-> cout<<” \n Enter the data \t\t in integer format\n other data \t\t in float”;

endl → manipulator operator used for new line

```
Cout<< endl<<"enter any data"<<endl;
```

C++ Additional operator →

:: -> scope resolution operator -> it is used to represent scope of data.

```
# include <iostream.h>
int a=30;
Void main()
{ int a=40;

    Cout<< a;

    Cout<< ::a;

}
```

::* → pointer to member

->* → pointer to member

New → operator ,used to define dynamic variable in program

Delete \rightarrow operator, used to delete the dynamic allocated variable

. \ast \rightarrow pointer to member operator

Precedence rule \rightarrow

the order of execution of operator in expression is known as precedence rule. The operator execute according to their precedence and associativity (direction of execution of operator).

Precedence	Operator	Description	Associativity
1	::	Scope resolution	Left-to-right
2	a++ a-- type() type{} a() a[] . ->	Suffix/postfix increment and decrement Functional cast Function call Subscript Member access	
3	++a --a +a -a ! ~ (type) *a &a sizeof new new[] delete delete[]	Prefix increment and decrement Unary plus and minus Logical NOT and bitwise NOT C-style cast Indirection (dereference) Address-of Size-of ^[note 1] Dynamic memory allocation Dynamic memory deallocation	Right-to-left
4	. \ast -> \ast	Pointer-to-member	Left-to-right
5	a*b a/b a%b	Multiplication, division, and remainder	
6	a+b a-b	Addition and subtraction	
7	<< >>	Bitwise left shift and right shift	
8	<=>	Three-way comparison operator (since C++20)	
9	< <= > >=	For relational operators < and ≤ respectively For relational operators > and ≥ respectively	
10	== !=	For relational operators = and ≠ respectively	
11	&	Bitwise AND	
12	^	Bitwise XOR (exclusive or)	
13		Bitwise OR (inclusive or)	
14	&&	Logical AND	
15		Logical OR	
16	a?b:c throw = += -= *= /= %= <<= >>= &= ^= =	Ternary conditional ^[note 2] throw operator Direct assignment (provided by default for C++ classes) Compound assignment by sum and difference Compound assignment by product, quotient, and remainder Compound assignment by bitwise left shift and right shift Compound assignment by bitwise AND, XOR, and OR	Right-to-left
17	,	Comma	Left-to-right

If the same precedence operator come together then operator execute according to their associativity

$$\underline{A*b} + \underline{c/(e\%f)*k-d+e}$$

Q. WAP to read a year and check their leap year condition by ternary operator .

Q. WAP to read a number and print whether it is odd or even.

Q. WAP to calculate area and perimeter of a rectangle and print which one is greater. Length and width is given.

Q. WAP to read four digit number and print its reverse. 3456 → 6543

Q. you have an amount value and print total count of 2000,500,200,100,50,20,10,5,2,1 rupees in it .

2345 → 2000 → 1. 200→1, 100-→1 , 50 → 0, 20 → 2, 10→0 , 5→1

1. ((year % 400 == 0) || (year % 100 != 0 && year % 4 != 0)) ? cout << "Given year is leap year " : cout << "Given year is not leap year";

2. Reverse → num → 4 digit → d1,d2,d2,d4,res

Cin >> num; 5678

D1 = num % 10; 8

Num = num / 10; 567

D2 = num % 10; 7

Num = num / 10 56

D3 = num % 10; 6

Num = num / 10; 5

D4 = num; 5

Res = d1 * 1000 + d2 * 100 + d3 * 10 + d4;

→ 8765

(Num % 2 == 0) ? cout << "given number is even" : cout << "given number is odd";

3. Amount

Tt = amount / 2000;

Amount = amount % 2000;

Fh = amount / 500;

Amount = amount % 500;

Th = amount / 200;

Amount=amount % 200;

Oh= amount /100;

Amount=amount%100;

Fifty= amount/50;

Amount=amount%50;

Tw=amount/20;

Ten=amount/10;

Amount=amount%10;

Five=amount/5

Amount=amount%5;

Two=amount/2;

Amount=amount%2;

One=amount;

Q. find out area and perimeter of square and print which one is greater.

Q. Find out sum of square of digit of the given 4 digit number

Q. print whether the given number is divisible by 7 and 3.

X