

[C Quiz](#)[C++ Tutorials](#)[Blog Home](#)[Data Science](#)

AD
[Data Science Tutorials](#)

[Categories](#)[Machine Learning Tutorials](#)[C Tutorials](#)[Free Courses](#)[Big Data](#)[Beginner Tutorials](#)[Python Tutorials](#)[Data Science Tutorials](#)[Bioinformatics Tutorials](#)[Apache Spark Tutorials](#)[SQL & NoSQLs](#)[Relational Tutorials](#)[JavaScript Tutorials](#)[AngularJS Tutorials](#)[ReactJS Tutorials](#)[Java Script Tutorials](#)[MongoDB Tutorials](#)[SAS Tutorials](#)[SAP HANA Tutorials](#)[AI Tutorials](#)[C Tutorials](#)[C Interview Questions](#)

mathematical operations. Operators in C and C++, are tools or symbols that are used to perform

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

Logical, Conditional and Bitwise Operations.

AD

There are many sub-operators present in each type of Operator in C/C++. Let's discuss one by one with their examples.

This is just the beginning, at the end of this article we will test your knowledge with an interactive quiz

Types of Operators in C and C++

There are 6 types of Operators in C/C++



C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials

+

operator.

C++ Interview Questions

+

AD

1. Arithmetic Operators

It includes basic arithmetic operations like addition subtraction, multiplication, division, modulus operations, increment, and decrement.

AD

The Arithmetic Operators in C and C++ include:

1. **+** (**Addition**) – This operator is used to add two operands.
2. **-** (**Subtraction**) – Subtract two operands.
3. ***** (**Multiplication**) – Multiply two operands.
4. **/** (**Division**) – Divide two operands and gives the quotient as the answer.
5. **%** (**Modulus operation**) – Find the remains of two integers and gives the remainder after the division.
6. **++** (**Increment**) – Used to increment an operand.

C Tutorials

+

C Interview Questions

+

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials

+

operand.

C++ Interview Questions

+

AD

Example of Arithmetic Operators in C

AD

```
1. #include<stdio.h>
2. int main()
3. {
4.     int a=10, b=7;
5.
6.     printf("Welcome to DataFlair
7. tutorials!\n\n");
8.     printf("The Addition of %d and %d is:
9. %d\n",a,b,a+b);
10.    printf("The Subtraction of %d and %d is:
11. %d\n",a,b,a-b);
12.    printf("The Multiplication of %d and %d
13. is: %d\n",a,b,a*b);
14.    printf("The Division of %d and %d is:
15. %d\n",a,b,a/b);
16.    printf("The Modulus operation of %d and %d
17. is: %d\n",a,b,a%b);
18.    printf("The Incremented value ++a is:
19. %d\n",++a);
20.    printf("The Decrementated value --b is:
21. %d\n",--b);
22.    return 0;
23. }
```

Code on Screen-

C Tutorials

+

C Interview Questions

+

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

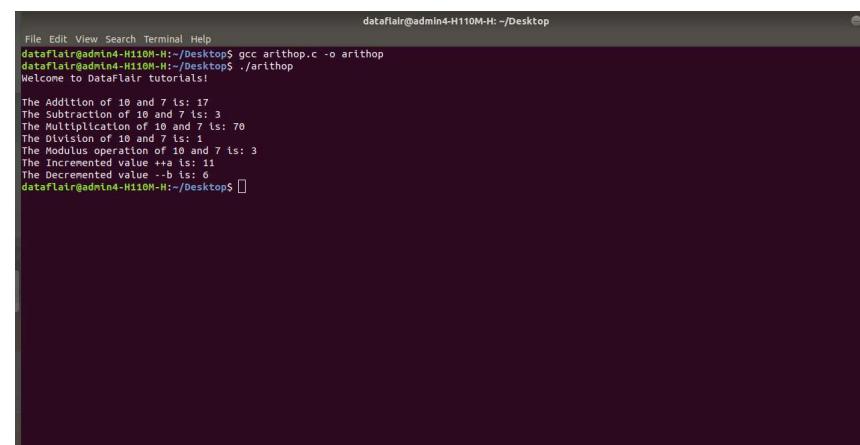
AD

```
int a=10, b=7;
printf("Welcome to DataFlair tutorials!\n\n");
printf("The Addition of %d and %d is: %d\n",a,b,a+b);
printf("The Subtraction of %d and %d is: %d\n",a,b,a-b);
printf("The Multiplication of %d and %d is: %d\n",a,b,a*b);
printf("The Division of %d and %d is: %d\n",a,b,a/b);
printf("The Modulus operation of %d and %d is: %d\n",a,b,a%b);
printf("The Incremented value ++a is: %d\n",++a);
printf("The Decrementated value --b is: %d\n",--b);
return 0;
}
```

AD

The output will be-

AD



```
dataflair@admin4-H110M-H: ~/Desktop
File Edit View Search Terminal Help
dataflair@admin4-H110M-H:~/Desktop$ gcc arithop.c -o arithop
dataflair@admin4-H110M-H:~/Desktop$ ./arithop
Welcome to DataFlair tutorials!

The Addition of 10 and 7 is: 17
The Subtraction of 10 and 7 is: 3
The Multiplication of 10 and 7 is: 70
The Division of 10 and 7 is: 1
The Modulus operation of 10 and 7 is: 3
The Incremented value ++a is: 11
The Decrementated value --b is: 6
dataflair@admin4-H110M-H:~/Desktop$
```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials

+

C++

C++ Interview Questions

+

AD

AD

Here is a code in C++ which illustrates all the basic arithmetic operators:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main()
5. {
6.     cout<<"Welcome to DataFlair
7. tutorials!\\n\\n"\\n<<endl<<endl;
8.     int a = 10, b = 7;
9.     cout<<"The Addition of "<< a << " and "
10.    b << " are: " << a + b << endl;
11.    cout<<"The Subtraction of "<< a << " and "
12.    << b << " are: " << a - b << endl;
13.    cout<<"The Multiplication of "<< a <<
14.    and " << b << " are: " << a * b << endl;
15.    cout<<"The Division of "<< a << " and "
16.    << b << " are: " << a / b << endl;
```

Code-

C Tutorials

+

C Interview Questions

+

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
int main()
{
cout<<"Welcome to DataFlair tutorials!"<<endl<<endl;
int a = 10, b = 7;
cout<<"The Addition of "<< a << " and " << b << " are: "<< a + b << endl;
cout<<"The Subtraction of "<< a << " and " << b << " are: "<< a - b << endl;
cout<<"The Multiplication of "<< a << " and " << b << " are: "<< a * b << endl;
cout<<"The Division of "<< a << " and " << b << " are: "<< a / b << endl;
cout<<"The Modulus operation between "<< a << " and " << b << " is: "<< a % b << endl;
cout<<"The Incremented value -+a is: "<< ++a << endl;
cout<<"The Decrementated value -a is: "<< --a << endl;
return 0;
}
```

AD

Output-

AD

```
dataflair@asus-System-Product-Name: ~/Desktop
File Edit View Search Terminal Help
dataflair@asus-System-Product-Name:~/Desktop$ g++ arithop.cpp -o arithop
dataflair@asus-System-Product-Name:~/Desktop$ ./arithop
Welcome to DataFlair tutorials!

The Addition of 10 and 7 are: 17
The Subtraction of 10 and 7 are: 3
The Multiplication of 10 and 7 are: 70
The Division of 10 and 7 are: 1
The Modulus operation between 10 and 7 is: 3
The Incremented value -+a is: 11
The Decrementated value -a is: 10
dataflair@asus-System-Product-Name:~/Desktop$
```

Table for Arithmetic Operators in C and C++

C Tutorials +

C Interview Questions +

Operator	Operand	Operation	Elucidation
----------	---------	-----------	-------------

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

[C++ Tutorials](#) +

[C++ Interview Questions](#) +

AD

-	a, b	a - b	Subtraction
*	a, b	a * b	Multiplication
/	a, b	a / b	Division
			Modulus operator – to find the remainder when two integral digits are divided
++	a	a ++	Increment
--	a	a --	Decrement

2. Relational Operators

It is used to compare two numbers by checking whether they are equal or not, less than, less than or equal to, greater than, greater than or equal to.

- == (Equal to)**– This operator is used to check if both operands are equal.
- != (Not equal to)**– Can check if both operands are not equal.

[C Tutorials](#) +

[C Interview Questions](#) +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

~~is greater than the second.~~

4. < (**Less than**)– Can check if the first operand is lesser than the second.
5. >= (**Greater than equal to**)– Check if the first operand is greater than or equal to the second.
6. <= (**Less than equal to**)– Check if the first operand is lesser than or equal to the second

If the relational statement is satisfied (it is true), the program will return the value 1, otherwise, if the relational statement is not satisfied (it is false), the program will return the value 0.

Example of Relational Operators in C-

```

1. #include <stdio.h>
2. int main()
3. {
4.     int a=10, b=10, c=20;
5.
6.     printf("Welcome to DataFlair
tutorials!\n\n");
7.
8.     printf("For %d == %d : The output is: %d
\n", a, b, a == b); // condition is true
9.     printf("For %d == %d : The output is: %d
\n", a, c, a == c); // condition is false
10.

```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```

13.     \n", a, b, a != b); // condition is false
14.     printf("For %d > %d : The output is: %d
15.     \n", a, b, a > b); // condition is false
16.     printf("For %d > %d : The output is: %d
17.     \n", a, c, a > c); // condition is false
18.     printf("For %d < %d : The output is: %d
19.     \n", a, b, a < b); // condition is false
20.     printf("For %d >= %d : The output is: %d
21.     \n", a, b, a >= b); // condition is true
22.     printf("For %d >= %d : The output is: %d
23.     \n", a, c, a >= c); // condition is false
24.     printf("For %d <= %d : The output is: %d
25.     \n", a, b, a <= b); // condition is true
26.     printf("For %d <= %d : The output is: %d
27.     \n", a, c, a <= c); // condition is true
28.     return 0;
29. }
```

AD

Code on Screen-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```

int a=10, b=10, c=20;
printf("Welcome to DataFlair tutorials!\n\n");

printf("For %d == %d : The output is: %d \n", a, b, a == b); // condition is true
printf("For %d == %d : The output is: %d \n", a, c, a == c); // condition is false

printf("For %d != %d : The output is: %d \n", a, b, a != c); // condition is true
printf("For %d != %d : The output is: %d \n", a, b, a != b); // condition is false

printf("For %d > %d : The output is: %d \n", a, b, a > b); // condition is false
printf("For %d > %d : The output is: %d \n", a, c, a > c); // condition is false

printf("For %d < %d : The output is: %d \n", a, b, a < b); // condition is true
printf("For %d < %d : The output is: %d \n", a, c, a < c); // condition is true

printf("For %d >= %d : The output is: %d \n", a, b, a >= b); // condition is true
printf("For %d >= %d : The output is: %d \n", a, c, a >= c); // condition is false

printf("For %d <= %d : The output is: %d \n", a, b, a <= b); // condition is true
printf("For %d <= %d : The output is: %d \n", a, c, a <= c); // condition is true
return 0;
}

```

AD

The output will be-

```

dataflair@admin4-H110M-H:~/Desktop
dataflair@admin4-H110M-H:~/Desktop$ gcc relationop.c -o relationop
dataflair@admin4-H110M-H:~/Desktop$ ./relationop
Welcome to DataFlair tutorials!

For 10 == 10 : The output ls: 1
For 10 == 20 : The output ls: 0
For 10 != 20 : The output ls: 1
For 10 != 10 : The output ls: 0
For 10 > 10 : The output ls: 0
For 10 > 20 : The output ls: 0
For 10 < 10 : The output ls: 0
For 10 < 20 : The output ls: 1
For 10 <= 10 : The output ls: 1
For 10 <= 20 : The output ls: 1
For 10 != 10 : The output ls: 0
For 10 <= 20 : The output ls: 1
dataflair@admin4-H110M-H:~/Desktop$ 

```

Example of Relational Operators in C++

Here is a code in C++ which illustrates all the basic relational operators:

C Tutorials +

C Interview Questions +

```

1. #include <iostream>
2. using namespace std;

```

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

C Tutorials +

C Interview Questions +

```
5. cout<<"\n";
6.
7. cout<<"Welcome to DataFlair tutorials!" AD
<<endl<<endl;
8.
9. int a = 10, b = 10, c = 20;
10.
11. cout<<"For " << a << " == " << b << " The
    output is: " << (a == b) << endl; //
condition is true
12. cout<<"For " << a << " == " << c << " The
    output is: " << (a == c) << endl; //
condition is false
13.
14. cout<<"For " << a << " != " << c << " The
    output is: " << (a != c) << endl; //
condition is true
15. cout<<"For " << a << " != " << b << " The
    output is: " << (a != b) << endl; //
condition is false
16.
17. cout<<"For " << a << " > " << b << " The
    output is: " << (a > b) << endl; //
condition is false
18. cout<<"For " << a << " > " << c << " The
    output is: " << (a > c) << endl; //
condition is false
19.
20. cout<<"For " << a << " < " << b << " The
    output is: " << (a < b) << endl; //
condition is false
21. cout<<"For " << a << " < " << c << " The
    output is: " << (a < c) << endl; //
condition is true
22.
23. cout<<"For " << a << " >= " << b << " The
    output is: " << (a >= b) << endl; //
condition is true
24. cout<<"For " << a << " >= " << c << " The
    output is: " << (a >= c) << endl; //
```

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

26. cout << "For " << a << " <= " << b << endl; //
output is: " << (a <= b) << endl; //
condition is true AD
27. cout << "For " << a << " <= " << c << " Th
output is: " << (a <= c) << endl; //
condition is true
28.
29. **return** 0;
30. }

Code-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Table for Relational Operators in C and C++

Operator	Operand	Operation	Elucidation
----------	---------	-----------	-------------

C Tutorials +

==

a, b

(a==b)

Used to check if both operands are

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

$!=$ a, b $(a \neq b)$ Used to check both operands

AD

not equal

$>$ a, b $(a > b)$ Used to check the first opera
is greater than second

$<$ a, b $(a < b)$ Used to check the first opera
is lesser than the second

\geq a, b $(a \geq b)$ Used to check if the first operand
is greater than or equal to the second

\leq a, b $(a \leq b)$ Used to check if the first operand
is lesser than or equal to the second

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

It refers to the boolean values which can be expressed AD as:

- Binary logical operations, which involves two variables: AND and OR
- Unary logical operation: NOT

Logical Operators in C/C++ Includes –

1. **&& (AND)** – It is used to check if both the operands are true.
2. **|| (OR)** – These operators are used to check if at least one of the operand is true.
3. **! (NOT)** – Used to check if the operand is false

If the logical statement is satisfied (it is true), then the program will return the value 1, otherwise, if the relational statement is not satisfied (it is false), the program will return the value 0.

AD

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

Example of Logical Operators in C Programming-

```

1. #include <stdio.h>
2. int main()
3. {
4.     int a = 10, b = 10, c = 20, answer;
5.
6.     printf("Welcome to DataFlair
7.         tutorials!\n\n");
8.
9.     answer = (a == b) && (c > b);
10.    printf("For (%d == %d) && (%d != %d), the
11.           output is: %d \n",a,b,b,c,answer);
12.    //condition is true
13.
14.    answer = (a == b) && (c < b) && (c>0);
15.    printf("For (%d == %d) && (%d <= %d), the
16.           output is: %d \n",a,b,b,c,answer);
17.    //condition is false
18.
19.    answer = (a == b) || (b > c);
20.    printf("For (%d == %d) || (%d < %d), the
21.           output is: %d \n",a,b,c,b,answer); /-
22.           //condition is true
23.
24.    answer = (a != b) || (a <= b) || (a>c);
25.    printf("For (%d != %d) || (%d < %d), the
26.           output is: %d \n",a,b,c,b,answer);
27.    //condition is true
28.
29.    answer = !(a == b);
30.    printf("For !(%d == %d), the output is: %d
31.           \n",a,b,answer); //condition
32.           is false

```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
24.     printf( "For : (%d == %d), the output is: %s\n", a, b, answer); //condition  
is true  
25.     return 0;  
26. }
```

AD

Code on Screen-

Follow DataFlair on WhatsApp & Stay updated with latest technology trends.

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Example of Logical Operators in C++

Here is a code in C++ which illustrates all the basic logical operators:

C Tutorials +

C Interview Questions +

```
1. #include <iostream>
2. using namespace std;
3.
4. int main()
```

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```

7.
8. cout<<"Welcome to DataFlair tutorials!"  

    <<endl<<endl; AD
9. answer = (a == b) && (c > b);
10. cout<<"For (" << a << " == " << b << ")  

    (" << b << " != " << c << "), the output is  

    " << answer << endl; //condition is true
11.
12. answer = (a == b) && (c < b) && (c>0);
13. cout<<"For (" << a << " == " << b << ")  

    (" << b << " <= " << c << "), the output is  

    " << answer << endl; //condition is false
14.
15. answer = (a == b) || (b > c);
16. cout<<"For (" << a << " == " << b << ")  

    (" << c << " < " << b << "), the output is  

    " << answer << endl; //condition is true
17.
18. answer = (a != b) || (a <= b) || (a>c);
19. cout<<"For (" << a << " != " << b << ") &&  

    (" << c << " < " << b << "), the output is:  

    " << answer << endl; //condition is true
20.
21. answer = !(a == b);
22. cout<<"For !( " << a << " == " << b << " ),  

    the output is: " << answer << endl;  

    //condition is false
23.
24. answer = !(a != b);
25. cout<<"For !( " << a << " != " << b << " ),  

    the output is: " << answer << endl;  

    //condition is true
26.
27. return 0;
28. }
```

C Tutorials +

C Interview Questions +

Code-

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Output-

AD

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Table for Logical Operators in C and C++

Operator	Operand	Operation	Elucidation
&&	a, b	(a && b)	AND: Used to check if both the operands are true

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

||

a, b

(a || b)

if at least one
the operand : AD
true

!

a

!a

NOT: Used to
check if the
operand is fal

4. Assignment Operators

It is used to assign a particular value to a variable. We will discuss it in detail in the later section with its shorthand notations.

1. **= (Assignment)**- Used to assign a value from right side operand to left side operand.
2. **+= (Addition Assignment)**- To store the sum of both the operands to the left side operand.
3. **-= (Subtraction Assignment)** – To store the difference of both the operands to the left side operand.
4. ***= (Multiplication Assignment)** – To store the product of both the operands to the left side operand.

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

division of both the operands to the left side

AD

operand.

6. **%= (Remainder Assignment)** – To store the remainder of both the operands to the left side operand.

Example of Assignment Operators in C

```
1. #include<stdio.h>
2. int main()
3. {
4.     printf("Welcome to Dataflair
5. tutorials!\n\n");
6.     int number = 10, result;
7.     result = number;
8.
9.     printf("result = %d \n", result);
10.
11.    result += number; //Same as result =
12.        result + a
13.    printf("result = %d \n", result);
14.
15.    result -= number; //Same as result =
16.        result - a
17.    printf("result = %d \n", result);
18.
19.    result *= number; //Same as result =
20.        result * a
21.    printf("result = %d \n", result);
22.
23.    result /= number; //Same as result =
24.        result / a
```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
23.     result = number; // same as result =  
24.     result % a  
25.     printf("result = %d \n", result);          AD  
26. }
```

Code on Screen-

C Tutorials +

C Interview Questions +

The output will be-

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Example of Assignment Operators in C++

Here is a code in C++ which illustrates all the basic assignment operators:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main()
5. {
```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
8. int number = 10, result;
9. result = number; AD
10.
11. cout<<"result = "<< result<<endl;
12.
13. result += number; //Same as result =
   result + a
14. cout<<"result = "<< result<<endl;
15.
16. result -= number; //Same as result =
   result - a
17. cout<<"result = "<< result<<endl;
18.
19. result *= number; //Same as result =
   result * a
20. cout<<"result = "<< result<<endl;
21.
22. result /= number; //Same as result =
   result / a
23. cout<<"result = "<< result<<endl;
24.
25. result %= number; //Same as result =
   result % a
26. cout<<"result = "<< result<<endl;
27. return 0;
28. }
```

Code-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Output-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Table for Assignment Operators in C and C++

Operator	Operand	Operation	Elucidation
=	a, b	a=b	Used to assign a value from right side operand to left side operand

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

[C++ Tutorials](#)



[+=](#)

a, b

a+=b

of a+b is stored in

a

AD

AD

[-=](#)

a, b

a-=b

of a-b is stored in

a

[*=](#)

a, b

a*=b

of a*b is stored in

a

[/=](#)

a, b

a/b

a=a/b: The value

a

of a/b is stored in

[%=](#)

a, b

a%b

a=a %b: The value

a

of a%b is stored in

5. Bitwise Operators

It is based on the principle of performing operations bit by bit which is based on boolean algebra. It increases the processing speed and hence the efficiency of the program.

[C Tutorials](#)



[C Interview Questions](#)



Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

1. **& (Bitwise AND)** – Converts the value of both the operands into binary form and performs AN operation bit by bit.
2. **| (Bitwise OR)** – Converts the value of both the operands into binary form and performs OR operation bit by bit.
3. **^ (Bitwise exclusive OR)** – Converts the value of both the operands into binary form and performs EXCLUSIVE OR operation bit by bit.
4. **\sim (One's complement operator)**: Converts the operand into its complementary form.
5. **$<<$ – Left shift**
6. **$>>$ – Right shift**

Key takeaway: *Bitwise operators are not applicable in the case of float and double data type in C.*

In order to clearly understand bitwise operators, let us see the truth table for various bitwise operations and understand how it is associated with boolean algebra.

Since there are 2 variables, namely, a and b, there are 2² combinations for values a and b can take simultaneously.

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

FOR THE RESULT TO BE 1.

AD

OR – At least one operand should have boolean value 1 for the result to be 1.

XOR (EXCLUSIVE OR) – Either the first operand should have boolean value 1 or the second operand should have boolean value 1. Both cannot have the boolean value 1 simultaneously.

One Complement: iF

a	b	a & b	a b	a ^ b	$\sim a$
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

The left and right shift operators are responsible for shifting the binary values by some specific number of places.

C Tutorials +

C Interview Questions +

Left shift: It specifies the value of the left operand to be shifted to the left by the number of bits specified by

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

Right shift: It specifies the value of the left operand be shifted to the right by the number of bits specified by its right operand.

AD

Let us take an example each of performing bitwise AND, OR, EXCLUSIVE OR and ONE'S COMPLEMENT operation.

Consider two operands, a and b with values:

a = 26 and b=14

Therefore, a & b is computed as follows:

1. Find the binary equivalent of a and b:
2. Perform boolean AND/OR/EXCLUSIVE OR operation bit by bit
3. Convert the answer into its corresponding decimal form.

- **Bitwise AND**

a = 26 = 1 1 0 1 0

b = 14 = 0 1 1 1 0

a & b = 0 1 0 1 0 which is equal to 10

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

$a = 26 = 11010$

$b = 14 = 01110$

AD

$a \mid b = 11110$ which is equal to 30

- **Bitwise XOR**

$a = 26 = 11010$

$b = 14 = 01110$

$a \mid b = 10100$ which is equal to 20

- **Bitwise One's Complement**

$a = 26 = 11010$

Reversing its bits, we get 00101 which is equal to 5 but this is not the correct answer! The correct answer is: -(a+1) which is -27 which is in accordance with two's complement.

Example of Bitwise Operators in C

C Tutorials +

C Interview Questions +

```

1. #include <stdio.h>
2. int main()
3. {
4.

```

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
7. int a = 26, b = 14;
8. printf(" Bitwise AND operation %d & %d : AD
%d\n",a,b,a&b);
9. printf(" Bitwise OR operation %d | %d :
%d\n",a,b,a|b);
10. printf(" Bitwise XOR operation %d ^ %d :
%d\n",a,b,a^b);
11. printf(" Bitwise ONE'S COMPLEMENT ~ %d
operation : %d\n",a,~a);
12. return 0;
13. }
```

Code on Screen-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

The Output is –

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Example of Bitwise Operators in C++

Here is a code in C++ which illustrates all the basic bitwise operators:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main()
5. {
6.
```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
9. int a = 26, b = 14;
10.
11. cout<<" Bitwise AND operation" << a << "
12. " << b << " : " << (a&b) << endl;
13. cout<<" Bitwise OR operation" << a << "
14. " << b << " : " << (a|b) << endl;
15. cout<<" Bitwise XOR operation" << a << "
16. " << b << " : " << (a^b) << endl;
17. cout<<" Bitwise ONE'S COMPLEMENT ~" << a
18. " operation :" << (~a) << endl;
19. return 0;
20. }
```

AD

Code-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! **Enroll Now**

C++ Tutorials +

C++ Interview Questions +

AD

AD

Output-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Table for Bitwise Operators in C and C++

Operator	Operand	Operation	Elucidation
&	a, b	(a & b)	Bitwise AND: Converts the value of both the operands into binary form and

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

|

a, b

(a | b)

Bitwise OR: AD

Converts the value of both the operands into binary form and performs OR operation bit by bit.

^

a, b

(a ^ b)

Bitwise exclusive OR: Converts the value of both the operands into binary form and performs EXCLUSIVE OR operation bit by bit.

~

a

(~ a)

One's complement operator: Converts the operand into its complementary form.

C Tutorials +

C Interview Questions +

<<

a

a<<

Left shift

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

6. Miscellaneous Operators

Apart from the above-discussed operators, there are certain operators which fall under this category which include sizeof and ternary (conditional) operators.

Here is a table which illustrates the use of these operators:

1. **sizeof** – It returns the memory occupied by the particular data type of the operand
2. **& (Pointer)** – It refers to the address (memory location) in which the operand is stored.
3. *** (Pointer)** – It is a pointer operator
4. **? (Condition)** – It is an alternative for if-else condition

Example of Miscellaneous Operators in C

C Tutorials +

C Interview Questions +

```
1. #include<stdio.h>
2. int main()
3. {
4. }
```

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
7. // Use of * and & operator
8.
9. int number = 10, *pointer;
10. pointer=&number; //Here the pointer stor
the memory address of variable number
11. printf("The value of the number is:
%d\n",*pointer);
12.
13. // Use of ?: operator
14. int expression1 = 10, expression2 = 20,
expression3;
15. expression3 = ( expression1 > expression
) ? expression1 : expression2;
16. printf("The Output of the ternary
statement is: %d", expression3);
17. return 0;
18. }
```

AD

Code on Screen-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

The output is-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Example of Miscellaneous Operators in C++

Here is a code in C++ which illustrates some of the basic misc operators:

```
1. #include <iostream>
2. using namespace std;
3.
4. int main()
5. {
```

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

```
8.  
9. // Use of * and & operator AD  
10.  
11. int number = 10, *pointer;  
12. pointer=&number; //Here the pointer stor  
the memory address of variable number  
13. cout<<"The value of the number is: "  
<<*pointer<<endl;  
14.  
15. // Use of ?: operator  
16. int expression1 = 10, expression2 = 20,  
expression3;  
17. expression3 = ( expression1 > expression  
) ? expression1 : expression2;  
18. cout<<"The Output of the ternary stateme  
is: "<< expression3 << endl;  
19. return 0;  
20. }
```

Code-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! **Enroll Now**

C++ Tutorials +

C++ Interview Questions +

AD

AD

Output-

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

A table for Misc Operators in C and C++

Operator	Operand	Operation	Elucidation
sizeof	a	sizeof(a)	It returns the memory occupied by the particular

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

&

a

& a

operand
It refers to the
address (memory
location) in which
the operand is
stored.

AD

*

a

* a

It is a pointer

:

a,b

a? b:
statement

It is an alterna-
tive statement
for if-else
condition

Quiz on Operators in C and C++

1 2 3 4 5 6 7 8 9 10 11

12 13 14 15

Current Review / Skip Answered
Correct Incorrect

C Tutorials +

C Interview Questions +

Review Question

Quiz Summary

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

Predict the output of the following C/C++ Program AD

```
#include  
using namespace std;  
int main()  
{  
    int x, y, z;  
    x=6;y=5;z=4;  
    y=++x+z++;  
    z=y++*++x;  
    x=-y+z-;  
    cout<<x<<endl<<y<<endl<<z;  
}
```

99 11 87

80 10 69

100 12 87

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

Skip question

Check

AD

Summary

Operators are the basic foundation of the C/C++ Programming language. Now, you can perform any operation of mathematical, logical, relational, with other condition. We learned each operator in C and C++ with their examples. As a beginner, you should know each operator, and how, why, when to use it.

Technologies are changing, do you want to stay updated. If yes, follow us on [WhatsApp](#)

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

AD

Tags: Arithmetic Operators with example

Assignment Operators with example Bitwise Operators with example

C Operators C Operators with Example Logical Operators with example

Relational Operators with example

2 RESPONSES

 **Comments** 2  **Pingbacks** 0

harsh shukla ⌂ [August 21, 2019 at 5:11 pm](#)

best information ,
please keep it up like that

[Reply](#)

naykokohlein

C Tutorials +

C Interview Questions +

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

C++ Tutorials +

C++ Interview Questions +

AD

[Reply](#)

AD

LEAVE A REPLY

Comment *

Name *

Email *

[Post Comment](#)



FREE Education – Knowledge is a right, not a privilege.
DataFlair, the leading Ed-tech company, offers industry-grade free certification courses on

Trending Data Science Courses

- Free Python Course
- Free ML Course [Hindi]
- Free NumPy Course [Hindi]
- Free SciPy Course [Hindi]
- Free TensorFlow Course [Hindi]

Trending Data Science Tutorials

- Data Science Tutorials
- Machine Learning Tutorials
- Python Tutorials
- TensorFlow Tutorials
- Pandas Tutorials
- NumPy Tutorials

Trending Programming Tutorials

- C Tutorials
- C++ Tutorials
- Data Structure Tutorials
- Java Tutorials
- JSP Tutorials
- Spring Tutorials

Join millions thriving with our FREE certification courses. Experience success now!

[Enroll Now](#)

Contact Us

DataFlair Web Services Pvt Ltd,
140, Sector – D,
Sudama Nagar, Indore, 452009
Madhya Pradesh, India

720 University Avenue,
Suite 120, Los Gatos,
CA 95032, USA

Email: info@data-flair.training



About DataFlair

- Home
- About us
- Contact us
- Success Stories
- Write For Us
- Terms and Conditions
- Privacy Policy
- Disclaimer

[Hindi]

- Free Pandas Course [Hindi]
- Free OpenCV Course [Hindi]
- Free Python Project Course [Hindi]
- Free Python Course [Hindi]
- Free R Programming Course
- Free SQL Course
- Free Tableau Course

Trending Big Data Courses

- Free Big Data - Hadoop Course
- Free Spark - Scala Course
- Free Apache Flink Course
- Free Apache Kafka Course
- Free Hadoop + Spark Course
- Free AWS Course

Trending Courses

- Free C Course [Hindi]
- Free C++ Course

PyTorch Tutorials

- Data Mining Tutorials
- SQL Tutorials
- Cassandra Tutorials
- MongoDB Tutorials
- IoT Tutorials
- R Tutorials
- SAS Tutorials
- AI Tutorials
- Tableau Tutorials
- PowerBI Tutorials
- QlikView Tutorials
- Qlik Sense Tutorials
- SAP HANA Tutorials
- Big Data Tutorials
- Hadoop Tutorials
- Spark Tutorials
- PySpark Tutorials
- Flink Tutorials
- Kafka Tutorials
- MongoDB Tutorials
- Cloud Tutorials
- AWS Tutorials
- Microsoft Azure Tutorials

Trending Projects

- Python Projects

HTML Tutorials

- JavaScript Tutorials
- Node Js Tutorials
- Angular Tutorials
- Angular JS Tutorials
- SAP ABAP Tutorials
- Selenium Tutorials
- Android Tutorials

Trending Tutorials

- Linux Tutorials
- Blockchain Tutorials
- Salesforce Tutorials
- Docker Tutorials
- Cyber Security Tutorials
- Operating System Tutorials
- computer network tutorial
- Computer Basics Tutorials
- MS Word Tutorials
- PowerPoint Tutorials
- MS Excel Tutorials

Join millions thriving with our FREE certification courses. Experience success now! [Enroll Now](#)

- Free DSA in C++ Course
[Hindi]
- Free Web Dev Course
[Hindi]
- Free Java Course
- Free Scala Course
- Free Angular Course
- Free Java Course [Hindi]
- Free Adv Java Course
[Hindi]
- Free Hibernate Course
[Hindi]
- Free Spring Course [Hindi]
- Free JavaScript Course
[Hindi]
- AI Projects
- Computer Vision Projects
- IoT Projects
- Django Projects
- Java Projects
- Android Projects



DataFlair © 2023. All Rights Reserved.

