



(<https://www.educba.com/software-development/>)



(<https://www.educba.com/operators-precedence-in-c/>)



(<https://www.educba.com/control-statements-in-c/>)

Left Shift Operator in C

```
#include <stdio.h>
#include <conio.h>main()
{int a = 0;
int b = 0;
int result1 = 0, result2 = 0;
printf("\n Enter the number : ");
// accepting the value of first operand
from the userscanf("%d", &a);
result1 = a << 0;result2 = a << 34;
printf("The result of left shift operation is : ");
printf("\n %d << 0 = %d", a, result1);
printf("\n %d << 34 = %d", a, result2);}
```

Enter the number : 40
The result of left shift operation is :
40 << 0 = 40
40 << 34 = 160

www.educba.com

Introduction to Left Shift Operator in C

Left shift operator is a bitwise shift operator in C which operates on bits. It is a binary operator which means it requires two operands to work on. Following are some important points regarding Left shift operator in C:





(<https://www.educba.com/software-development/>)

second operand.

Syntax:

Start Your Free Software Development Course

Web development, programming languages, Software testing & others

The syntax for left shift operator in C is as follows:

```
variable_name<<number_of_positions
```

In the above statement, there are two values; the first one is an integer variable on which we want to apply left shift operator. The name of this variable can be any name given by the user. The second value is a number which specifies the number of positions a user wants to shift the bits to the left.

How Left Shift Operator Works in C?

Left shift operator requires two operands to work on. Both the operands of the left shift operator should be of integral type. It shifts the bits of the first operand to the left by the number of positions specified by the second operand. Simultaneously, the empty spaces created by the bits shifted to the left are then filled with zeroes. Including the sign bit, the bits which are shifted off to the end are then discarded.

Following are some cases when the result of left shift operation will be undefined:

- If the value of first operand is negative, then the result of left shift operation will be undefined.





(<https://www.educba.com/software-development/>)

Thus, left shift operation will be performed when both the operands are positive.



🔗 Popular Course in this category



C Programming Training (3 Courses, 5 Project)

3 Online Courses | 5 Hands-on Projects | 34+ Hours | Verifiable Certificate of Completion | Lifetime Access

★★★★★ 4.5 (8,604 ratings)

Course Price

\$79 ~~\$399~~

[View Course](#)

(<https://www.educba.com/software-development/courses/c-programming-course/?btnz=edu-blg-inline-banner1>)

Related Courses

C++ Training (4 Courses, 5 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/c-course/?btnz=edu-blg-inline-banner1>)

Java Training (40 Courses, 29 Projects, 4 Quizzes) (<https://www.educba.com/software-development/courses/java-course/?btnz=edu-blg-inline-banner1>)

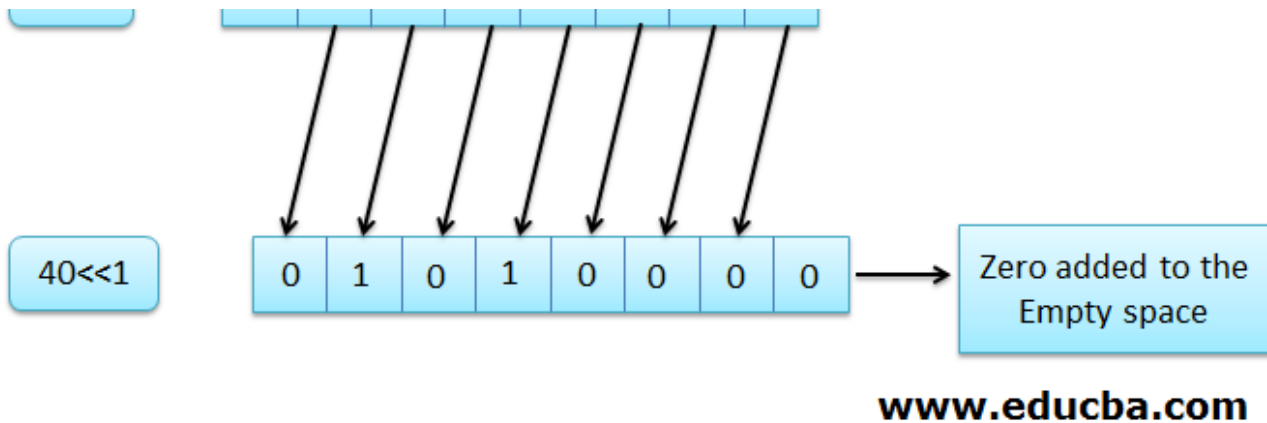
On the other hand, if the value of second operand is zero then left shift operation will not be performed. Such as the result of $40 \ll 0$ will be equal to 40 itself.



Let us now understand the working of left shift operator with the help of an example. In this



(<https://www.educba.com/software-development/>)



In the above example, we can see that on performing left shift operation on a binary value all its bits have been shifted to the left and the empty space created on the right side is filled with zero.

Thus, the value of $40 \ll 1$ is 01010000. The decimal equivalent of this binary value is 80.

Note: Whenever we will perform left shift operation on an integer 'a' with an integer 'b'. The result of the operation will always be equal of multiplying 'a' with 2^b . In the above case $a = 40$ and $b = 1$. Thus, the result of $a \ll b$ i.e. $40 \ll 1$ is ' $40 * 2^1$ ' which will be 80.

Examples to Implement Left Shift Operator in C

Below are the examples of Left Shift Operator in C:

Example #1

Example showing left shift operation performed on two positive operands.

Code:





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
int a = 40;
int b = 0;
printf("\n Enter number of positions for the bits to shift to the left : ");
// accepting the value of second operand from the user
scanf("%d", &b);
printf("The result of left shift operation is : ");
// Binary value of 40 is 101000
printf("\n %d << %d = %d", a, b, a<<b);
}
```

Output:

```
Enter number of positions for the bits to shift to the left : 1
The result of left shift operation is :
40 << 1 = 80
```

Example #2

Example showing a scenario when the value of second operand is negative.

Code:

```
#include<stdio.h>
#include<conio.h>
main()
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

```
// Binary value of 40 is 101000
printf("\n 40 << -1 = %d", result);
}
```

Output:

```
The result of left shift operation is :
40 << -1 = 20
```

Along with this there is a warning in the program for the line highlighted in yellow.

Below is the Warning:

```
C26452: Arithmetic overflow: Left shift count is negative or greater than or equal to the operand size which is undefined behavior (io.3).
```

We got this warning because our second operand is negative.

Example #3

Example showing scenario when the value of first operand is negative.

Code:

```
#include<stdio.h>
#include<conio.h>
main()
{
    int result = 0;
    result = -40 << 1;
```





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

Output:

```
The result of left shift operation is :  
-40 << 1 = -80
```

Along with this there is a warning in the program for the line highlighted in yellow.

Below is the Warning:

```
C26453: Arithmetic overflow: Left shift of a negative signed number is undefined behavior (io.4).
```

We got this warning because our first operand is negative.

Example #4

Example showing scenarios when number of positions to be shifted is zero and greater than the size of integer.

Code:

```
#include <stdio.h>  
#include <conio.h>  
main()  
{  
    int a = 0;  
    int b = 0;  
    int result1 = 0, result2 = 0;  
    printf("\n Enter the number : ");
```





(<https://www.educba.com/software-development/>)

```
printf("The result of left shift operation is : ", result1);
printf("\n %d << 0 = %d", a, result1);
printf("\n %d << 34 = %d", a, result2);
}
```

Output:

```
Enter the number : 40
The result of left shift operation is :
40 << 0 = 40
40 << 34 = 160
```

Along with this, there is a warning in the program for line highlighted in yellow.

Below is the Warning:

C26452: Arithmetic overflow: Left shift count is negative or greater than or equal to the operand size which is undefined behavior (io.3).

We got this warning because the size of the operand is greater than the size of an integer.

Conclusion

- The left shift operator is a logical bitwise operator.
- It is a binary operator that operates on two positive integral operands.
- It shifts the bits to the left by the number of positions specified by its second operand.
- Empty spaces created in the right are filled with zeroes.

Recommended Articles



This is a guide to Left Shift Operator in C. Here we discuss the Introduction of Left Shift Operator in C and how it works along with different Examples and its Code Implementation.



[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

3. [Introduction to Reverse String in C \(https://www.educba.com/reverse-string-in-c/\)](https://www.educba.com/reverse-string-in-c/)
4. [Reverse String in PHP | Loops \(https://www.educba.com/reverse-string-in-php/\)](https://www.educba.com/reverse-string-in-php/)

C PROGRAMMING TRAINING (3 COURSES, 5 PROJECT)

- ☒ 3 Online Courses
- ☒ 5 Hands-on Projects
- ☒ 34+ Hours
- ☒ Verifiable Certificate of Completion
- ☒ Lifetime Access

Learn More

<https://www.educba.com/software-development/courses/c-programming-course/?btnz=educba-blg-inline-banner3>

About Us

Blog (<https://www.educba.com/blog/?source=footer>)





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

source=footer)

Contact Us (<https://www.educba.com/contact-us/?source=footer>)

Verifiable Certificate (<https://www.educba.com/software-development/verifiable-certificate/?source=footer>)

Reviews (<https://www.educba.com/software-development/reviews/?source=footer>)

Terms and Conditions (<https://www.educba.com/terms-and-conditions/?source=footer>)

Privacy Policy (<https://www.educba.com/privacy-policy/?source=footer>)

Apps

iPhone & iPad (<https://itunes.apple.com/in/app/educba-learning-app/id1341654580?mt=8>)

Android (<https://play.google.com/store/apps/details?id=com.educba.www>)

Resources

Free Courses (<https://www.educba.com/software-development/free-courses/?source=footer>)

Java Tutorials (<https://www.educba.com/software-development/software-development-tutorials/java-tutorial/?source=footer>)

Python Tutorials (<https://www.educba.com/software-development/software-development-tutorials/python-tutorial/?source=footer>)

All Tutorials (<https://www.educba.com/software-development/software-development-tutorials/?source=footer>)

Certification Courses





[\(https://www.educba.com/software-development/\)](https://www.educba.com/software-development/)

[development/courses/python-certification-course/?source=footer\)](https://www.educba.com/software-development/courses/python-certification-course/?source=footer)

Java Course (<https://www.educba.com/software-development/courses/java-course/?source=footer>)

Become a Selenium Automation Tester (<https://www.educba.com/software-development/courses/selenium-training-certification/?source=footer>)

Become an IoT Developer (<https://www.educba.com/software-development/courses/iot-course/?source=footer>)

ASP.NET Course (<https://www.educba.com/software-development/courses/asp-net-course/?source=footer>)

VB.NET Course (<https://www.educba.com/software-development/courses/vb-net-course/?source=footer>)

PHP Course (<https://www.educba.com/software-development/courses/php-course/?source=footer>)

© 2022 - EDUCBA. ALL RIGHTS RESERVED. THE CERTIFICATION NAMES ARE THE TRADEMARKS OF THEIR RESPECTIVE OWNERS.

