# Bitwise operator

1. The **& (bitwise AND)** in C or C++ takes two numbers as operands and does AND on every bit of two numbers. The result of AND is 1 only if both bits are 1.
2. The **| (bitwise OR)** in C or C++ takes two numbers as operands and does OR on every bit of two numbers. The result of OR is 1 if any of the two bits is 1.
3. The **^ (bitwise XOR)** in C or C++ takes two numbers as operands and does XOR on every bit of two numbers. The result of XOR is 1 if the two bits are different.
4. The **<< (left shift)** in C or C++ takes two numbers, left shifts the bits of the first operand, the second operand decides the number of places to shift.
5. #include <stdio.h>
7. // Function to return the only odd
8. // occurring element
9. int findOdd(int arr[], int n)
10. {
11. int res = 0, i;
12. for (i = 0; i < n; i++)
13. res ^= arr[i];
14. return res;
15. }

## Ternary operator

if(Expression1)

{

variable = Expression2;

}

else

{

variable = Expression3;

}

The **ternary operator** is an **operator** that exists in some programming languages, which takes three operands rather than the typical one or two that most **operators** use. It provides a way to shorten a simple if else block.