

# BINARY OPERATORS OVERLOADING IN C++

[http://www.tutorialspoint.com/cplusplus/binary\\_operators\\_overloading.htm](http://www.tutorialspoint.com/cplusplus/binary_operators_overloading.htm)

Copyright © tutorialspoint.com

The unary operators take two arguments and following are the examples of Binary operators. You use binary operators very frequently like addition + operator, subtraction – operator and division / operator.

Following example explains how addition + operator can be overloaded. Similar way, you can overload subtraction – and division / operators.

```
#include <iostream>
using namespace std;

class Box
{
    double length;    // Length of a box
    double breadth;   // Breadth of a box
    double height;    // Height of a box
public:

    double getVolume(void)
    {
        return length * breadth * height;
    }
    void setLength( double len )
    {
        length = len;
    }

    void setBreadth( double bre )
    {
        breadth = bre;
    }

    void setHeight( double hei )
    {
        height = hei;
    }
    // Overload + operator to add two Box objects.
    Box operator+(const Box& b)
    {
        Box box;
        box.length = this->length + b.length;
        box.breadth = this->breadth + b.breadth;
        box.height = this->height + b.height;
        return box;
    }
};

// Main function for the program
int main( )
{
    Box Box1;           // Declare Box1 of type Box
    Box Box2;           // Declare Box2 of type Box
    Box Box3;           // Declare Box3 of type Box
    double volume = 0.0; // Store the volume of a box here

    // box 1 specification
    Box1.setLength(6.0);
    Box1.setBreadth(7.0);
    Box1.setHeight(5.0);

    // box 2 specification
    Box2.setLength(12.0);
    Box2.setBreadth(13.0);
    Box2.setHeight(10.0);
```

```
// volume of box 1
volume = Box1.getVolume();
cout << "Volume of Box1 : " << volume <<endl;

// volume of box 2
volume = Box2.getVolume();
cout << "Volume of Box2 : " << volume <<endl;

// Add two object as follows:
Box3 = Box1 + Box2;

// volume of box 3
volume = Box3.getVolume();
cout << "Volume of Box3 : " << volume <<endl;

return 0;
}
```

When the above code is compiled and executed, it produces the following result:

```
Volume of Box1 : 210
Volume of Box2 : 1560
Volume of Box3 : 5400
```

Loading [Mathjax]/jax/output/HTML-CSS/jax.js