

OVERLOADING INCREMENT ++ AND DECREMENT --

http://www.tutorialspoint.com/cplusplus/increment_decrement_operators_overloading.htm

Copyright © tutorialspoint.com

The increment ++ and decrement -- operators are two important unary operators available in C++.

Following example explain how increment ++ operator can be overloaded for prefix as well as postfix usage. Similar way, you can overload operator --.

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

class Time
{
private:
    int hours;           // 0 to 23
    int minutes;         // 0 to 59
public:
    // required constructors
    Time(){
        hours = 0;
        minutes = 0;
    }
    Time(int h, int m){
        hours = h;
        minutes = m;
    }
    // method to display time
    void displayTime()
    {
        cout << "H: " << hours << " M:" << minutes << endl;
    }
    // overloaded prefix ++ operator
    Time operator++ ()
    {
        ++minutes;           // increment this object
        if(minutes >= 60)
        {
            ++hours;
            minutes -= 60;
        }
        return Time(hours, minutes);
    }
    // overloaded postfix ++ operator
    Time operator++( int )
    {
        // save the original value
        Time T(hours, minutes);
        // increment this object
        ++minutes;
        if(minutes >= 60)
        {
            ++hours;
            minutes -= 60;
        }
        // return old original value
        return T;
    }
};

int main()
{
    Time T1(11, 59), T2(10, 40);

    ++T1;                    // increment T1
    T1.displayTime();        // display T1
```

```
++T1;           // increment T1 again
T1.displayTime(); // display T1

T2++;           // increment T2
T2.displayTime(); // display T2
T2++;           // increment T2 again
T2.displayTime(); // display T2
return 0;
}
```

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

```
H: 12 M:0
H: 12 M:1
H: 10 M:41
H: 10 M:42
```

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