

Declarations

```
class MyClass {
private:
    int a;
    string str;

public:
    // constructor
    MyClass(int a, string str);
    MyClass();
    void print ();
    friend bool operator< (const MyClass &c1, const MyClass &c2);
    friend bool operator> (const MyClass &c1, const MyClass &c2);
    friend bool operator== (const MyClass &c1, const MyClass &c2);
    friend ostream& operator<< (ostream& strm, const MyClass& m);

};
```

Overloaded methods

```
bool operator< (const MyClass &c1, const MyClass &c2) {
    if (c1.a < c2.a) {
        return true;
    }
    else {
        return false;
    }
}

bool operator> (const MyClass &c1, const MyClass &c2) {
    if (c1.a > c2.a) {
        return true;
    }
    else {
        return false;
    }
}

bool operator== (const MyClass &c1, const MyClass &c2) {
    if (c1.a == c2.a) {
        return true;
    }
    else {
        return false;
    }
}

ostream& operator<< (ostream& strm, const MyClass &m) {
    strm << "[" << m.a << " ";
    return strm;
}
```

Sample Output

```
$ g++ act11.cpp
$ ./a.out
class1: [18]   class2: [12]
class1 < class2: 0
class1 > class2: 1
class1 == class2: 0
```