```
/* The following code example is taken from the book
 * "The C++ Standard Library - A Tutorial and Reference, 2nd Edition"
 * by Nicolai M. Josuttis, Addison-Wesley, 2012
 *
 * (C) Copyright Nicolai M. Josuttis 2012.
 * Permission to copy, use, modify, sell and distribute this software
 * is granted provided this copyright notice appears in all copies.
 * This software is provided "as is" without express or implied
 * warranty, and with no claim as to its suitability for any purpose.
 */
#include <iostream>
#include <set>
#include "print.hpp"
using namespace std;

// type for runtime sorting criterion
class RuntimeCmp {
  public:
    enum cmp_mode {normal, reverse};
  private:
    cmp_mode mode;
  public:
    // constructor for sorting criterion
    // - default criterion uses value normal
    RuntimeCmp (cmp_mode m=normal) : mode(m) {
    }
    // comparison of elements
    // - member function for any element type
    template <typename T>
    bool operator() (const T& t1, const T& t2) const {
        return mode==normal ?  t1<t2
                            :  t2<t1;
    }
    // comparison of sorting criteria
    bool operator== (const RuntimeCmp& rc) const {
        return mode == rc.mode;
    }
};

// type of a set that uses this sorting criterion
typedef set<int,RuntimeCmp> IntSet;

int main()
{
    // create, fill, and print set with normal element order
    // - uses default sorting criterion
    IntSet coll1 = { 4, 7, 5, 1, 6, 2, 5 };
    PRINT_ELEMENTS (coll1, "coll1: ");

    // create sorting criterion with reverse element order
    RuntimeCmp reverse_order(RuntimeCmp::reverse);

    // create, fill, and print set with reverse element order
    IntSet coll2(reverse_order);
    coll2 = { 4, 7, 5, 1, 6, 2, 5 };
    PRINT_ELEMENTS (coll2, "coll2: ");
```

```
    // assign elements AND sorting criterion
    coll1 = coll2;
    coll1.insert(3);
    PRINT_ELEMENTS (coll1, "coll1: ");

    // just to make sure...
    if (coll1.value_comp() == coll2.value_comp()) {
        cout << "coll1 and coll2 have the same sorting criterion"
             << endl;
    }
    else {
        cout << "coll1 and coll2 have a different sorting criterion"
             << endl;
    }
}
```