

```

/* The following code example is taken from the book
 * "The C++ Standard Library – A Tutorial and Reference"
 * by Nicolai M. Josuttis, Addison-Wesley, 1999
 *
 * (C) Copyright Nicolai M. Josuttis 1999.
 * 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 <list>
#include <deque>
#include <algorithm>
#include "countptr.hpp"
using namespace std;

void printCountedPtr (CountedPtr<int> elem)
{
    cout << *elem << ' ';
}

int main()
{
    // array of integers (to share in different containers)
    static int values[] = { 3, 5, 9, 1, 6, 4 };

    // two different collections
    typedef CountedPtr<int> IntPtr;
    deque<IntPtr> coll1;
    list<IntPtr> coll2;

    /* insert shared objects into the collections
     * - same order in coll1
     * - reverse order in coll2
     */
    for (int i=0; i<sizeof(values)/sizeof(values[0]); ++i) {
        IntPtr ptr(new int(values[i]));
        coll1.push_back(ptr);
        coll2.push_front(ptr);
    }

    // print contents of both collections
    for_each (coll1.begin(), coll1.end(),
              printCountedPtr);
    cout << endl;
    for_each (coll2.begin(), coll2.end(),
              printCountedPtr);
    cout << endl << endl;

    /* modify values at different places
     * - square third value in coll1
     * - negate first value in coll1
     * - set first value in coll2 to 0
     */
    *coll1[2] *= *coll1[2];

```

```
(**coll1.begin()) *= -1;
(**coll2.begin()) = 0;

// print contents of both collections again
for_each (coll1.begin(), coll1.end(),
          printCountedPtr);
cout << endl;
for_each (coll2.begin(), coll2.end(),
          printCountedPtr);
cout << endl;
}
```