

```

/* 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 <list>
#include <iostream>
#include <algorithm>
#include <iterator>
using namespace std;

void printLists (const list<int>& l1, const list<int>& l2)
{
    cout << "list1: ";
    copy (l1.cbegin(), l1.cend(), ostream_iterator<int>(cout, " "));
    cout << endl << "list2: ";
    copy (l2.cbegin(), l2.cend(), ostream_iterator<int>(cout, " "));
    cout << endl << endl;
}

int main()
{
    // create two empty lists
    list<int> list1, list2;

    // fill both lists with elements
    for (int i=0; i<6; ++i) {
        list1.push_back(i);
        list2.push_front(i);
    }
    printLists(list1, list2);

    // insert all elements of list1 before the first element with value 3 of
list2
    // - find() returns an iterator to the first element with value 3
    list2.splice(find(list2.begin(), list2.end(), // destination position
                    3),
                list1); // source list
    printLists(list1, list2);

    // move first element of list2 to the end
    list2.splice(list2.end(), // destination position
                list2, // source list
                list2.begin()); // source position
    printLists(list1, list2);

    // sort second list, assign to list1 and remove duplicates
    list2.sort();
    list1 = list2;
    list2.unique();
    printLists(list1, list2);
}

```

```
    // merge both sorted lists into the first list
    list1.merge(list2);
    printLists(list1, list2);
}
```