```
/* 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 "algostuff.hpp"
using namespace std;

int main()
{
    vector<int> coll1 = { 1, 1, 2, 3, 4, 5, 6, 7, 8, 9 };
    PRINT_ELEMENTS(coll1,"coll1: ");

    // check whether coll1 is sorted
    if (is_sorted (coll1.begin(), coll1.end())) {
        cout << "coll1 is sorted" << endl;
    }
    else {
        cout << "coll1 is not sorted" << endl;
    }

    map<int,string> coll2;
    coll2 = { {1,"Bill"}, {2,"Jim"}, {3,"Nico"}, {4,"Liu"}, {5,"Ai"} };
    PRINT_MAPPED_ELEMENTS(coll2,"coll2: ");

    // define predicate to compare names
    auto compareName = [](const pair<int,string>& e1,
                          const pair<int,string>& e2){
                            return e1.second<e2.second;
                       };

    // check whether the names in coll2 are sorted
    if (is_sorted (coll2.cbegin(), coll2.cend(),
                   compareName)) {
        cout << "names in coll2 are sorted" << endl;
    }
    else {
        cout << "names in coll2 are not sorted" << endl;
    }

    // print first unsorted name
    auto pos = is_sorted_until (coll2.cbegin(), coll2.cend(),
                                compareName);
    if (pos != coll2.end()) {
        cout << "first unsorted name: " << pos->second << endl;
    }
}
```