```
/* 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 <unordered_set>
#include <numeric>
#include "print.hpp"
using namespace std;

int main()
{
    // create and initialize unordered set
    unordered_set<int> coll = { 1,2,3,5,7,11,13,17,19,77 };

    // print elements
    // - elements are in arbitrary order
    PRINT_ELEMENTS(coll);

    // insert some additional elements
    // - might cause rehashing and create different order
    coll.insert({-7,17,33,-11,17,19,1,13});
    PRINT_ELEMENTS(coll);

    // remove element with specific value
    coll.erase(33);

    // insert sum of all existing values
    coll.insert(accumulate(coll.begin(),coll.end(),0));
    PRINT_ELEMENTS(coll);

    // check if value 19 is in the set
    if (coll.find(19) != coll.end()) {
        cout << "19 is available" << endl;
    }

    // remove all negative values
    unordered_set<int>::iterator pos;
    for (pos=coll.begin(); pos!= coll.end(); ) {
        if (*pos < 0) {
            pos = coll.erase(pos);
        }
        else {
            ++pos;
        }
    }
    PRINT_ELEMENTS(coll);
}
```