

```

// ENGG 233 Fall 2004 L02 --- Dr. S. A.Norman
// Solution to "Self-Test" on Oct 22 lecture handout.

#include <iostream>
#include <vector>
using namespace std;

void sort_doubles(vector<double>& v);

int main()
{
    vector<double> v;           // Make an empty vector.
    cout << "Please enter a list of positive numbers, "
         << "separated by spaces." << endl;
    cout << "Use a negative number or zero to mark the end of the list." << endl;

    // Put user's positive numbers into v one at a time.
    // WARNING: This relies on the user following instructions properly!
    double user_input;
    while (true) {
        cin >> user_input;
        if (user_input <= 0.0)
            break;
        v.resize(v.size() + 1); // Make room for one more element.
        v.at(v.size() - 1) = user_input; // New element = user_input.
    }

    sort_doubles(v);
    cout << "From smallest to largest the numbers are:" << endl;
    for (int k = 0; k < v.size(); k++)
        cout << v.at(k) << " ";
    cout << endl;
    return 0;
}

// This is the selection_sort function from the Oct. 22 handout, except
// that I changed to the element type to double.
void sort_doubles(vector<double>& v)
{
    int vsize = v.size();

    // One pass through the outer loop body swaps v[i] with the smallest
    // element among v[i], ... , v[vsize-1].
    for (int i = 0; i < vsize - 1; i++) {
        int i_of_min = i;

        // Completion of the inner loop sets i_of_min equal to the index
        // of the smallest element among v[i], ... , v[vsize-1].
        for (int j = i + 1; j < vsize; j++)
            if (v.at(j) < v.at(i_of_min))
                i_of_min = j;

        double temp = v.at(i);
        v.at(i) = v.at(i_of_min);
        v.at(i_of_min) = temp;
    }
}

```