```
// $Id: vecsort.cpp,v 1.28 2015-03-31 18:29:40-07 - - $

//
// NAME
//    vecsort - sort argv lexicographically
//
// SYNOPSIS
//    vecsort [argv...]
//
// DESCRIPTION
//    Takes each element of argv and inserts it into a vector,
//    paired with its index.  Sorts the vector lexicographically,
//    then prints it out.  Includes argv[0] in the sort.
//

#include <algorithm>
#include <cstdlib>
#include <iostream>
#include <utility>
#include <vector>
using namespace std;

using stringint = pair<string,int>;
using vecstrint = vector<stringint>;

template <typename pair_collection>
void printvec (const string &label, const pair_collection &vec) {
   for (const auto& item: vec) {
      cout << label << ": {" << item.first << ", " << item.second
           << "}" << endl;
   }
   cout << endl;
}

// Sort by string then by int.
// On second test, after "or":
// not (2.f < 1.f) means 2.f >= 1.f means 1.f <= 2.f
// But 1.f < 2.f is false, so this means 1.f == 1.f
bool by_string (const stringint &si1, const stringint &si2) {
   return si1.first < si2.first
       or (not (si2.first < si1.first) and si1.second < si2.second);
}

// Sort by int then by string.
bool by_int (const stringint &si1, const stringint &si2) {
   return si1.second < si2.second
       or (not (si2.second < si1.second) and si1.first < si2.first);
}

```

```

int main (int argc, char **argv) {
   vecstrint vec {{"foo",  3}, {"bar", 14}, {"baz",  1},
                  {"qux", 77}, {"qux",  7}, {"qux", 14},
                  {"foo", 14}, {"bar",  1}, {"baz",  3}};
   for_each (&argv[1], &argv[argc],
             [&](char* arg) {vec.push_back ({arg, arg - argv[0]}); });
   printvec ("Unsorted", vec);
   sort (vec.begin(), vec.end());
   printvec ("Default", vec);
   sort (vec.begin(), vec.end(), by_string);
   printvec ("By string", vec);
   sort (vec.begin(), vec.end(), by_int);
   printvec ("By int", vec);
   return EXIT_SUCCESS;
}

//TEST// ./vecsort >vecsort.lis 2>&1
//TEST// mkpspdf vecsort.ps vecsort.cpp* vecsort.lis

```

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: starting vecsort.cpp
vecsort.cpp:
        $Id: vecsort.cpp,v 1.28 2015-03-31 18:29:40-07 - - $
g++ -g -O0 -Wall -Wextra -rdynamic -std=gnu++11 vecsort.cpp -o vecsort -lglut -lGLU -lGL -lX11 -lrt -lm
rm -f vecsort.o
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ mkc: finished vecsort.cpp
```

```
Unsorted: {foo, 3}
Unsorted: {bar, 14}
Unsorted: {baz, 1}
Unsorted: {qux, 77}
Unsorted: {qux, 7}
Unsorted: {qux, 14}
Unsorted: {foo, 14}
Unsorted: {bar, 1}
Unsorted: {baz, 3}

Default: {bar, 1}
Default: {bar, 14}
Default: {baz, 1}
Default: {baz, 3}
Default: {foo, 3}
Default: {foo, 14}
Default: {qux, 7}
Default: {qux, 14}
Default: {qux, 77}

By string: {bar, 1}
By string: {bar, 14}
By string: {baz, 1}
By string: {baz, 3}
By string: {foo, 3}
By string: {foo, 14}
By string: {qux, 7}
By string: {qux, 14}
By string: {qux, 77}

By int: {bar, 1}
By int: {baz, 1}
By int: {baz, 3}
By int: {foo, 3}
By int: {qux, 7}
By int: {bar, 14}
By int: {foo, 14}
By int: {qux, 14}
By int: {qux, 77}

```