ECE2036: (Week 13) Introduction to "Iterators" in Standard Template Libraries

In the Standard Template Library (STL) we have "containers," "iterators," and "algorithms."

Containers are just the data structures with member functions – like vectors.

vector <type> name(size);

Iterators are LIKE pointers to the data structures that help to manipulate them the syntax to declare an iterator is:

vector <type>::iterator name;

Algorithms are some standard functions that will perform common actions on containers , like sorting, but they typically work through iterators to perform the functions.

#include <algorithm>

```
//This program takes in a file containing strings … changes the
//file to have an alphabetized list of elements.

#include <iostream>
#include <string>
#include <vector>
#include <algorithm>
#include <fstream>
using namespace std;

int main(int argc, char * * argv)
{

  ifstream inputFile;
  ofstream outputFile;

  if (argc < 2)
     cout << "Please input the filename" << endl;
  else
  {
    inputFile.open(argv[1], ios::in);

     if (inputFile)
     {
        string inputString;
        vector <string> names;

        while ( inputFile >> inputString)
           names.push_back(inputString);

        inputFile.close();

        sort(names.begin(), names.end());

        outputFile.open(argv[1],ios::out);

        for (vector <string>::iterator it = names.begin(); it < names.end(); ++it)
           outputFile << ( *it) << endl;

        outputFile.close();

     }
     else
     {
       cout << "Error in opening file" << endl;
     }

  }
  return 0; }
```