Lab06 Created/Modified by Laith Assaf

**List.h:**

#define LIST\_SIZE 20

#include "City.h"

class list

{ public:

list(); // Constructor - empty list

bool insert(city arg); // Add a city

void display(ostream& out); // Output data

void sort(string arg); // Sort by distance from arg

int size(); // Return number of cities

float dist(int src, int dst); // Distance to another city

private:

int len; // Number of used cities

city map[LIST\_SIZE]; // Data set

};

**List.cpp:**

#include <iostream>

#include <iomanip>

using namespace std;

#include "List.h"

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* list()

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

list::list()

{ len = 0;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* insert()

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

bool list::insert(city arg)

{

// Check to see if there is room

if(len>=LIST\_SIZE) return false;

// Add to array

map[len++] = arg;

// Return success

return true;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* display()

\* Display the list of cities, latitudes, longitudes and distances

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void list::display(ostream &out)

{

// Print header

out << "City Lat Long Dist" << endl;

out << "-------------- ------ ------ ------" << endl;

// Print data rows

for(int i = 0; i < len; i++) {

// City name with padding to 17 characters

out << left << setw(17) << map[i].name;

// Latitude with padding to 5 characters

out << fixed << setprecision(2) << setw(5) << map[i].latitude;

// Longitude with padding to 8 characters

out << " " << fixed << setprecision(2) << setw(8) << map[i].longitude;

// Distance to first city on same line

out << " " << fixed << setprecision(2) << dist(0, i) << endl;

}

out << endl;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* dist()

\* Calculate the distance between each city and the first city

\* sqrt((latsrc-latdst)^2 + (longsrc-longdst)^2)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

float list::dist(int src, int dst)

{

// Calculate the distance

return sqrt(pow(map[src].latitude - map[dst].latitude, 2) +

pow(map[src].longitude - map[dst].longitude, 2));

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* size()

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

int list::size()

{

return len;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* sort()

\* Search for the matching city name in the list. Swap that city with the first element in the list.

\* Use a bubble or to order the cities in ascending order based on the distance from the home city (map index 0).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void list::sort(string arg)

{

bool found = false;

int i = 0;

int j = 0;

city temp;

// Find the city with the matching name

while (i < len && !found)

{

if(map[i].name == arg)

{

// Swap the city with the first city

city temp = map[0];

map[0] = map[i];

map[i] = temp;

found = true;

}

i++;

}

// Bubble sort the cities based on the distance from the home city

for(int i = 0; i < len; i++)

{

for(int j = 0; j < len - 1; j++)

{

if(dist(0, j) > dist(0, j + 1))

{

city temp = map[j];

map[j] = map[j + 1];

map[j + 1] = temp;

}

}

}

}

**Output:**

Enter file name: Cities01.txt

10 Cities

City Lat Long Dist

-------------- ------ ------ ------

Lansing 42.73 -84.55 0.00

Detroit 42.33 -83.04 1.56

Flint 43.01 -83.68 0.91

Grand-Rapids 42.96 -85.66 1.13

Jackson 42.27 -84.47 0.47

Kalamazoo 42.23 -85.55 1.12

Ann-Arbor 42.22 -83.75 0.95

Mt-Pleasant 43.60 -84.78 0.90

Clare 43.82 -84.77 1.11

Saginaw 43.42 -83.95 0.91

Enter home city: Clare

City Lat Long Dist

-------------- ------ ------ ------

Clare 43.82 -84.77 0.00

Mt-Pleasant 43.60 -84.78 0.22

Saginaw 43.42 -83.95 0.91

Lansing 42.73 -84.55 1.11

Grand-Rapids 42.96 -85.66 1.24

Flint 43.01 -83.68 1.36

Jackson 42.27 -84.47 1.58

Kalamazoo 42.23 -85.55 1.77

Ann-Arbor 42.22 -83.75 1.90

Detroit 42.33 -83.04 2.28

(base) laithassaf@Laiths-MacBook-Air-2 Lab06 % ./program

Enter file name: Cities02.txt

8 Cities

City Lat Long Dist

-------------- ------ ------ ------

Midland 43.62 -84.23 0.00

Bay-City 43.59 -83.89 0.34

Mt-Pleasant 43.60 -84.78 0.55

Oscoda 44.42 -83.33 1.20

Clare 43.82 -84.77 0.58

Saginaw 43.42 -83.95 0.34

Traverse-City 44.75 -85.60 1.78

Ludington 43.96 -86.44 2.24

Enter home city: Midland

City Lat Long Dist

-------------- ------ ------ ------

Midland 43.62 -84.23 0.00

Bay-City 43.59 -83.89 0.34

Saginaw 43.42 -83.95 0.34

Mt-Pleasant 43.60 -84.78 0.55

Clare 43.82 -84.77 0.58

Oscoda 44.42 -83.33 1.20

Traverse-City 44.75 -85.60 1.78

Ludington 43.96 -86.44 2.24