CS205 C/ C++ Programming - Lab Assignment 3

Name: 邱煜（Qiu Yu）

SID: 11611127

Part1 - Analysis

The problem is to calculate the flying distance between two cities. The cities should in the given file `world_cities.csv`. This csv file stores some cities' `city name`, `province or state name(may be absent)`, `country name`, `latitude`, `longtitude`.

The user just need to enter two cities' name only. No need to input their latitudes and longtitudes. This program will search their latitudes and longtitudes from file `world_cities_csv`.

And then the program use the below function to calculate the distance between the two cities and output the result.

> phi = 90 - latitude
>
> theta = longitude
>
> c = sin(phi1 * pi / 180) * sin(phi2 * pi / 180) * cos((theta1-theta2)* pi / 180) +cos(phi1 * pi / 180) * cos(phi2 * pi / 180)
>
> d = R*arccos(c)

The program should be in a loop that keeps running until the user input the exit command `bye` (case-insensitive)

And there're some special situation need to handle:

1. The maximum length for name is 25. (there exists a city whose name is larger than 25)
2. The size of the array which stores all the cities' information is 800. (The total number is 988 rows.)
3. The file `world_cities.csv` isn't found.
4. The input city isn't found or its length < 3.
5. The input city is some cities' names' substring, such as "New York", "New". The program should display the result and prompt user to choose it when substring matched.

Part2 - Code

```cpp
//
//  main.cpp
//  lab3
//
//  Created by 邱煜 on 2019/3/29.
//  Copyright © 2019 邱煜. All rights reserved.
//

#include <iostream>
#include <fstream>
#include <string>
#include <cmath>
#include <cstdlib>
#include <algorithm>
#include "def.hpp"

using namespace std;

City cities[arraySize];
int citisNum;
string fCity, sCity;

int main() {
    double fLati, fLong, sLati, sLong;
    double result;

    ifstream file ("world_cities.csv");
    if(!file.is_open()){
        cout << "Open file failed!" << endl;
        exit(9);
    }
    string value;
    int i = 0;
    while(file.good()){
        // city name
        getline(file, value, ',');
        if(value.length())
        {
            cities[i].cityName = value;
            if(value.length()>nameMaxLen){
                cities[i].cityName =
value.substr(0,nameMaxLen);
                cout << value << " is truncated to " <<
value.substr(0,nameMaxLen) << " because its name is too
long." << endl;
                cout << "So please enter the name after
truncated if you want to enter this city." << endl;
```

```
            }
        }
        // province name(ignored)
        getline(file, value, ',');
        // country name
        getline(file, value, ',');
        if(value.length())
        {
            cities[i].countryName = value;
        }
        // latitude
        getline(file, value, ',');
        if(value.length())
        {
            cities[i].latitude = atof(value.c_str());
        }
        // longtitude
        getline(file, value);
        if(value.length())
        {
            cities[i].longtitude = atof(value.c_str());
        }
        // handle the situation that the number of city > the array size
        if(i>arraySize-2){
            cout << "The cities array has been filled, others data after " << i+1 << " row are not loaded." << endl;
            break;
        }
        i++;
    }
    file.close();
    citisNum = i-1;
    if(i<arraySize-1){
        cout << "Loading successfully!\n" << endl;
    }

    while(1){
        cout << "-----------------------------------------------------" << endl;
        cout << "Enter the first city's name(Enter \"bye\" to exit): ";
        getline(cin, fCity);
        trim(fCity);
        if(toUpper(fCity) == "BYE"){
            cout << "Bye" << endl;
```

```
            exit(0);
        }
        int idx1 = handleInput(fCity, 1);
        while(idx1 == -1){
            cout << "Enter the first city's name(Enter \"bye\" to exit): ";
            getline(cin, fCity);
            trim(fCity);
            if(toUpper(fCity) == "BYE"){
                cout << "Bye" << endl;
                exit(0);
            }
            idx1 = handleInput(fCity, 1);
        }
        cout << "Enter the second city's name(Enter \"bye\" to exit): ";
        getline(cin, sCity);
        trim(sCity);
        if(toUpper(sCity) == "BYE"){
            cout << "Bye" << endl;
            exit(0);
        }
        int idx2 = handleInput(sCity,2);
        while(idx2 == -1){
            cout << "Enter the second city's name(Enter \"bye\" to exit): ";
            getline(cin, sCity);
            trim(sCity);
            if(toUpper(sCity) == "BYE"){
                cout << "Bye" << endl;
                exit(0);
            }
            idx2 = handleInput(sCity,2);
        }

        fLati = cities[idx1].latitude;
        fLong = cities[idx1].longtitude;
        sLati = cities[idx2].latitude;
        sLong = cities[idx2].longtitude;
        result = calculate(fLati, fLong, sLati, sLong);
        cout << "The distance between " << fCity <<  " and " << sCity << " is " << result << " km." << endl;
        cout << "----------------------------------------\n" << endl;
    }
    return 0;
}
```

```
17
18  // 0 <= phi <= 180 -180 <= theta <= 180
19  // -90 <= latitude <= 90 -180 <= longtitude <= 180
20  double calculate(double fLati, double fLong, double sLati,
 1  double sLong){
13      double phi1, phi2, theta1, theta2;
12      double c;
13      double result;
14      phi1 = 90 - fLati;
15      theta1 = fLong;
16      phi2 = 90 - sLati;
17      theta2 = sLong;
18      c = sin(convert(phi1)) * sin(convert(phi2)) *
 9  cos(convert(theta1-theta2)) + cos(convert(phi1)) *
    cos(convert(phi2)) ;
14      result = R * acos(c);
10      return result;
11  }
12
13  // conversion from angle to radian
14  double convert(double angle){
15      return angle * Pi / 180;
16  }
17
18  // handle all cases - return the city's index in the cities
 9  array
15  // fOrS - 1 means this is the first city; 2 means second.
10  int handleInput(string input, int fOrS){
11      cout << endl;
12
13      // Invalid case - #characters < 3
14      if(input.length() < 3){
15          cout << "Invalid input - Less than 3 characters."
 6  << endl;
15          return -1;
17      }
18
10      // valid cases
10      return findIndex(input, fOrS);
16  }
10
18  int findIndex(string cityName, int fOrS){
16      int j = 0;
15      bool NotFound = true;
16      bool alreadyHave = false;
17      for (int i = 0; i<citisNum; i++) {
 8
```

```
        if (toUpper(cities[i].cityName) ==
toUpper(cityName)) {
            return i;
        }else
if(toUpper(cities[i].cityName).find(toUpper(cityName)) ==
0){
            if (!alreadyHave) {
                cout << "The matched cities you want
including: " << endl;
                alreadyHave = true;
            }
            NotFound = false;
            cout << j+1 << ": " << cities[i].cityName  <<
endl;
            j++;
        }
    }
    if (NotFound) {
        cout << "City " << cityName << " not found!" <<
endl;
    }else{
        cout << "Please input the entire name of the exact
city you want.\n" << endl;
    }
    return -1; // not found
}

// clear the whitespaces on both ends of the input string
void trim(string &str){
    if(!str.empty()){
        str.erase(0,str.find_first_not_of(" "));
        str.erase(str.find_last_not_of(" ") + 1);
    }
}

// to upper the input string
string toUpper(string str){
    string new_str = str;
    transform(new_str.begin(), new_str.end(),
new_str.begin(), (int (*)(int))toupper);
    return new_str;
}
```

```
//
//  def.hpp
```

```
3  //  lab3
4  //
5  //  Created by 邱煜 on 2019/3/29.
6  //  Copyright © 2019 邱煜. All rights reserved.
7  //
8
9  #ifndef def_hpp
10 #define def_hpp
11 #include <string>
12 #define Pi 3.141592654
13 #define R 6371
14
15 const int arraySize = 1000;
16 const int nameMaxLen = 35;
17
18 double calculate(double fLati, double fLong, double sLati,
   double sLong);
19 double convert(double angle);
20 int handleInput(std::string input, int fOrS);
21 int findIndex(std::string input, int fOrS);
22 void trim(std::string &str);
23 std::string toUpper(std::string str);
24
25 struct City{
26     std::string cityName;
27     std::string countryName;
28     double latitude;
29     double longtitude;
30 };
31
32 #endif /* stru_h */
```

H2 Part 3 - Result & Verification

H3 Requirements part1

H4 1.

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Las Palmas de Gran Canaria is truncated to Las Palmas de Gran Canari because its
 name is too long.
So please enter the name after truncated if you want to enter this city.
The cities array has been filled, others data after 800 row are not loaded.
--------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

2.

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!


-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

3.

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Open file failed!
 ✘ Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

4.

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Open file failed!
 ✘ Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!


-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

Part 2

1.

Case - bye

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Open file failed!
 ✘ Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!


-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

Normal case

Test case #1:

```
Input:
    shenzhen
    beijing
Output:
    1941.39
Verification:
    1950
```

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!

-------------------------------------------------------
Enter the first city's name(Enter "bye" to exit):    shenzhen

Enter the second city's name(Enter "bye" to exit): beijing

The distance between shenzhen and beijing is 1941.39 km.
-------------------------------------------------------


-------------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

Travel Route:		
Shenzhen Arpt, Shenzhen, CN (SZX) to		
Beijing Airport, Beijing, CN (BJS)		
One Way Distances:		
Distance	**1210 miles**	**1950 km**
Elite bonus	**0 miles**	**0 km**
Class of service bonus	**0 miles**	**0 km**
Special promotion bonus	**0 miles**	**0 km**
TOTAL	**1210 miles**	**1950 km**

Test case #2:

```
Input:
    new york city
    rio de janeiro
Output:
    7751.43
Verification:
    7760
```

Travel Route:		
New York City Airport, New York City, NY (NYC) to		
Rio De Janeiro Airport, Rio De Janeiro, BR (RIO)		
One Way Distances:		
Distance	**4820 miles**	**7760 km**
Elite bonus	**0 miles**	**0 km**
Class of service bonus	**0 miles**	**0 km**
Special promotion bonus	**0 miles**	**0 km**
TOTAL	**4820 miles**	**7760 km**

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!

------------------------------------------------------
Enter the first city's name(Enter "bye" to exit): new york city

Enter the second city's name(Enter "bye" to exit): rio de janeiro

The distance between new york city and rio de janeiro is 7751.43 km.
------------------------------------------------------

------------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

2.

Case - not found & length < 3

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!

------------------------------------------------------
Enter the first city's name(Enter "bye" to exit): a

Invalid input - Less than 3 characters.
Enter the first city's name(Enter "bye" to exit): b

Invalid input - Less than 3 characters.
Enter the first city's name(Enter "bye" to exit): aca

The matched cities you want including:
1: Acapulco
Please input the entire name of the exact city you want.

Enter the first city's name(Enter "bye" to exit): sfasfaf

City sfasfaf not found!
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

3.

Case - substring matching - e.g. "New York" and "New"

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!

-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit): new york

The matched cities you want including:
1: New York City
Please input the entire name of the exact city you want.

Enter the first city's name(Enter "bye" to exit): new york city

Enter the second city's name(Enter "bye" to exit): new

The matched cities you want including:
1: New Delhi
2: New Orleans
3: New York City
4: Newcastle upon Tyne
5: Newcastle
Please input the entire name of the exact city you want.

Enter the second city's name(Enter "bye" to exit): new delhi

The distance between new york city and new delhi is 11756.6 km.
-----------------------------------------------------


-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

4.

Case - ignore the whitespaces on both ends.

```
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●  ./main
Loading successfully!

-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit):    shenzhen

Enter the second city's name(Enter "bye" to exit): beijing

The distance between shenzhen and beijing is 1941.39 km.
-----------------------------------------------------


-----------------------------------------------------
Enter the first city's name(Enter "bye" to exit): bye
Bye
 Personal@Coreys  ~/github/C-CPP-Assignment/lab3  master ●
```

Part 4 - Difficulties & Solutions

---

1. Some function needs to implements. For example, `ToUpper ()`, `trim ()`. Then I implements them myself.
2. C++ could detect the overflow of the array. I assign a variable to record how much line it have read. Then once it larger than the pre-setting array size, the

program stop reading file and display it.