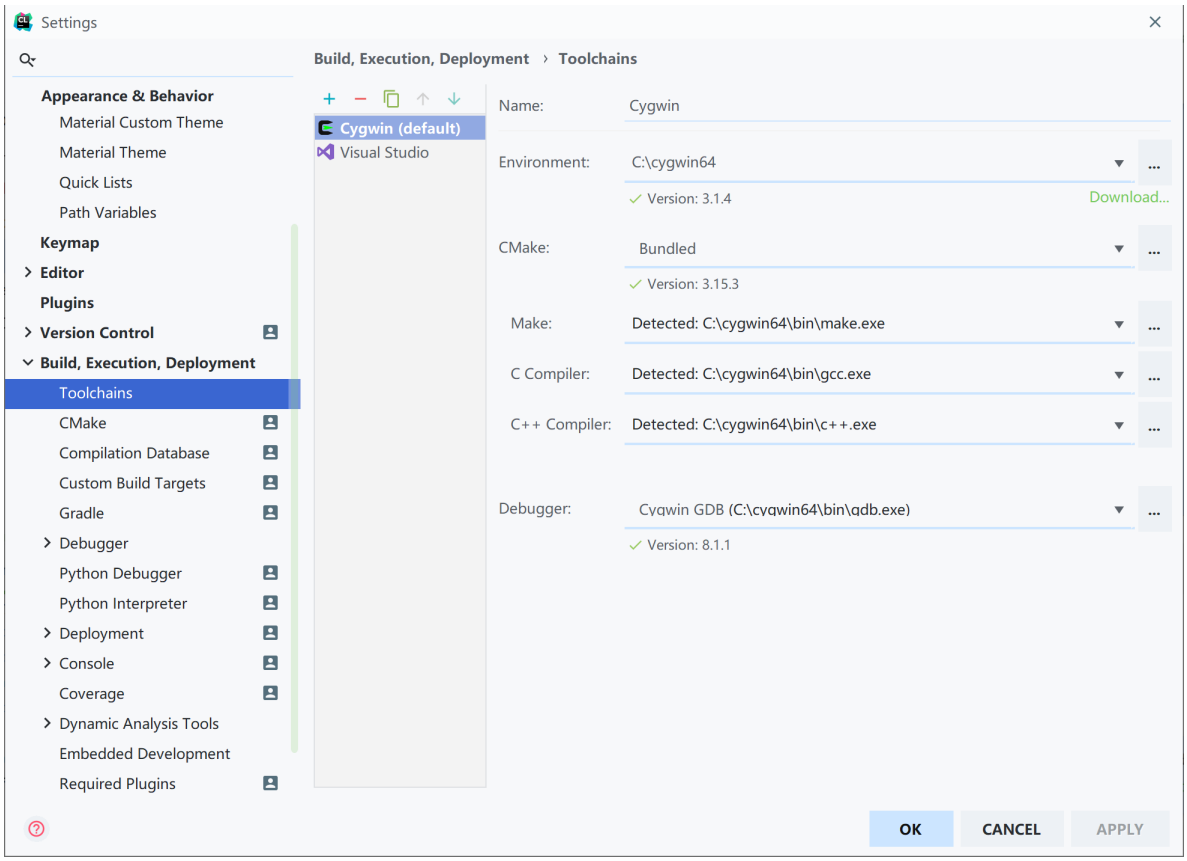
CS205 C/C++ Programming - Lab Assignment 2

Name: 蔡永宁(Cai Yongning)

SID: 11710802

environment:

coding on *CLion, Cygwin, window 10*



compile and run on *Window Subsystem Linux, Ubuntu, VS Code*

Part 1 - Analysis

Step1: program must store the information of city, so create a `struct city`

Step2: The city information is read from CSV and stored in an array of objects `city cities_info[]`

> reference: C++读取CSV文件中的数据

Step3: Loop in until the correct city name is read in, when an incorrect city name is read in, a matching list with sequence number is output, use sequence number to choose again

Step4: Calculation of distance

other: trim string

> reference: trim a string in C++

Part 2 - Code

```
//
// created by Andyb1ance 2020/3/15
//
// assignment2: compute the flying distance between the two and display
//

#include <iostream>
#include <string>
#include <cmath>
#include <fstream>
#include <sstream>
#include <vector>


using namespace std;
#define PI 3.1415926535
#define MAX_NAME_LENGTH 35
#define MAX_ARRAY_LENGTH 1000
struct city {
    char city_name[MAX_NAME_LENGTH];
    char country_name[MAX_NAME_LENGTH];
    string longitude;
    string latitude;
};


double stringToFloat(string);

double stringToInt(string);

double calDistance(string, string, string, string);

double degreeToRad(double);

string trim(string);

int checkNumber(string,int);

int main() {

    //TODO：把这一整块改写成函数
    ifstream csv;
    csv.open("world_cities.csv");
    if(!csv.good()){
        cout << "missing file, please check again";
        return 0;
```

```
    }
    city cities_info[MAX_ARRAY_LENGTH];
    string line;
    int lines=-1;
    while(getline(csv,line)){
        lines++;
        if(lines + 1 > MAX_ARRAY_LENGTH){
            cout << "out of index, data is truncated or not loaded\n";
            return 0;
        }
        istringstream sin(line);
        vector<string> city;
        string temp;
        while(getline(sin,temp,',')) city.push_back(temp);
        if(city[0].size() > MAX_NAME_LENGTH || city[2].size() > MAX_NAME_LENGTH)
{
            cout << "out of index, data is truncated or not loaded\n";
            return 0;
        }
        city[0].copy(cities_info[lines].city_name,city[0].length(),0);
        city[2].copy(cities_info[lines].country_name,city[2].length(),0);
        cities_info[lines].latitude=city[3];
        cities_info[lines].longitude=city[4];
    }

    while(true){

        string first[3];
        string second[3];

        //TODO：感觉思路不太对 得把这一整块改写成函数
        while(true){
            cout << "Enter the first city name: ";
            string temp;
            getline(cin,temp);
            temp=trim(temp);
            if(temp=="bye"){
                return 0;
            }
            if(temp.length() < 3){
                cout << "name shorter than three letters\n";
                continue;
            }else{
                vector<city> matched_list;
                for(int i=0; i<lines; i++){
                    if(temp==cities_info[i].city_name){

                    }
                }
                for(int i=0; i<lines; i++){
                    string temp2=cities_info[i].city_name;
                    if(temp2.find(temp) != string::npos){
                        matched_list.push_back(cities_info[i]);
                    }
                }
                if(matched_list.size()>1){
                    cout << "matched cities: \n";
                    for(int i=0;i<matched_list.size();i++){
```

```
                        cout << i << ". " << matched_list[i].city_name <<", " <<
matched_list[i].country_name << endl;
                    }
                    cout << "choose a city with a number between " << 0 << " and
" << matched_list.size()-1 << "or input other thing to skip" << endl;
                    string number;
                    getline(cin,number);
                    if(checkNumber(number,matched_list.size()-1)!=-1){
                        first[0]=matched_list[stringToInt(number)].latitude;
                        first[1]=matched_list[stringToInt(number)].longitude;
                        first[2]=matched_list[stringToInt(number)].city_name;
                        break;
                    }
                }else if(matched_list.empty()){
                    cout << "no matched city\n";
                }else{
                    first[0]=matched_list[0].latitude;
                    first[1]=matched_list[0].longitude;
                    first[2]=matched_list[0].city_name;
                    break;
                }
            }
        }

        //TODO 使用函数
        while(true){
            cout << "Enter the second city name: ";
            string temp;
            getline(cin,temp);
            temp=trim(temp);
            if(temp=="bye"){
                return 0;
            }
            if(temp.length() < 3){
                cout << "name shorter than three letters\n";
                continue;
            }else{
                vector<city> matched_list;
                for(int i=0; i<lines; i++){
                    string temp2=cities_info[i].city_name;
                    if(temp2.find(temp) != string::npos){
                        matched_list.push_back(cities_info[i]);
                    }
                }
                if(matched_list.size()>1){
                    cout << "matched cities: \n";
                    for(int i=0;i<matched_list.size();i++){
                        cout << i << ". " << matched_list[i].city_name <<", " <<
matched_list[i].country_name << endl;
                    }
                    cout << "choose a city with a number between " << 0 << " and
" << matched_list.size()-1 << " or input other thing to skip" << endl;
                    string number;
                    getline(cin,number);
                    if(checkNumber(number,matched_list.size()-1)!=-1){
                        second[0]=matched_list[stringToInt(number)].latitude;
                        second[1]=matched_list[stringToInt(number)].longitude;
                        second[2]=matched_list[stringToInt(number)].city_name;
```

```
                        break;
                    }
                }else if(matched_list.size() <1){
                    cout << "no matched city\n";
                }else{
                    second[0]=matched_list[0].latitude;
                    second[1]=matched_list[0].longitude;
                    second[2]=matched_list[0].city_name;
                    break;
                }
            }
        }
        double res=calDistance(first[0],second[0],first[1],second[1]);
        cout << "The distance between " << first[2] << " and " << second[2] << "
is " << res << " km" << endl;
    }
    return 0;
}

double stringToFloat(string num) {
    return atof(num.c_str());
}
double stringToInt(string num) {
    return atoi(num.c_str());
}

double degreeToRad(double degree) {
    return degree * PI / 180.0f;
}

double calDistance(string la1, string la2, string lo1, string lo2) {
    double phi1 = degreeToRad(90 - stringToFloat(la1));
    double phi2 = degreeToRad(90 - stringToFloat(la2));
    double theta1 = degreeToRad(stringToFloat(lo1));
    double theta2 = degreeToRad(stringToFloat(lo2));
    double c = sin(phi1) * sin(phi2) * cos(theta1 - theta2) + cos(phi1) *
cos(phi2);
    double d = 6371 * acos(c);
    return d;
}
string trim(string str)
{
    if (str.empty())
        return str;
    str.erase(0,str.find_first_not_of(" \n\r\t"));
    str.erase(str.find_last_not_of(" \n\r\t") + 1);
    return str;
}

int checkNumber(string num,int max) {
    if (num.find_first_not_of("-1234567890") != string::npos) {
        cout << "skip" << endl;
        return -1;
    } else if(stringToInt(num) < 0 || stringToInt(num) > max){
        return -1;
    } else {
        return 0;
    }
```

```
}
```

Part 3 - Result & Verification

Before test #1

```
Initially set the maximum length for names (city and country name) to 25, and
the array size to 800. Your program should issue a warning when data is
truncated or not loaded, but it mustn't crash.
```

```
#define MAX_NAME_LENGTH 25
#define MAX_ARRAY_LENGTH 800
```

```
D:\code\cpp\SUSTech_CS205_cpp\assignment2\cmake-build-debug\assignment2.exe
out of index, data is truncated or not loaded

Process finished with exit code 0
```

Before test #2

```
Rename the file to world_cities.tmp. Run your program. It mustn't crash and
should display a warning about the missing file
```

```
main.cpp
world_cities.tmp
```

```
D:\code\cpp\SUSTech_CS205_cpp\assignment2\cmake-build-debug\assignment2.exe
missing file, please check again
Process finished with exit code 0
```

Test case #1

```
Input:
New
3
New
4
bye

Output:
Enter the first city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4, or input other thing to skip
3
Enter the second city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4 or input other thing to skip
```

```
4
The distance between Newcastle upon Tyne and Newcastle is 16805 km
Enter the first city name: bye
```

```
andyb1ance@LAPTOP-M482O4RG:/mnt/d/code/cpp/SUSTech_CS205_cpp/assignment2$ g++ -o assignment2 main.cpp
andyb1ance@LAPTOP-M482O4RG:/mnt/d/code/cpp/SUSTech_CS205_cpp/assignment2$ ./assignment2
Enter the first city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4, or input other thing to skip
3
Enter the second city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4 or input other thing to skip
4
The distance between Newcastle upon Tyne and Newcastle is 16805 km
Enter the first city name: bye
```

Test case #2

```
Input:
New
3
New
4
Mos
0
Mos
1
bye
Ouput:
Enter the first city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4, or input other thing to skip
3
Enter the second city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4 or input other thing to skip
4
The distance between Newcastle upon Tyne and Newcastle is 16805 km
Enter the first city name: Mos
matched cities:
0. Moscow, Russia
1. Moscow, United States
2. Mosul, Iraq
choose a city with a number between 0 and 2, or input other thing to skip
```

```
0
Enter the second city name: Mos
matched cities:
0. Moscow, Russia
1. Moscow, United States
2. Mosul, Iraq
choose a city with a number between 0 and 2 or input other thing to skip
1
The distance between Moscow and Moscow is 5304.39 km
Enter the first city name: bye
```

```
andyb1ance@LAPTOP-M482O4RG:/mnt/d/code/cpp/SUSTech_CS205_cpp/assignment2$ ./assignment2
Enter the first city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4, or input other thing to skip
3
Enter the second city name: New
matched cities:
0. New Delhi, India
1. New Orleans, United States
2. New York City, United States
3. Newcastle upon Tyne, United Kingdom
4. Newcastle, Australia
choose a city with a number between 0 and 4 or input other thing to skip
4
The distance between Newcastle upon Tyne and Newcastle is 16805 km
Enter the first city name: Mos
matched cities:
0. Moscow, Russia
1. Moscow, United States
2. Mosul, Iraq
choose a city with a number between 0 and 2, or input other thing to skip
0
Enter the second city name: Mos
matched cities:
0. Moscow, Russia
1. Moscow, United States
2. Mosul, Iraq
choose a city with a number between 0 and 2 or input other thing to skip
1
The distance between Moscow and Moscow is 5304.39 km
Enter the first city name: bye
```

Test case #3

```
Input:
new
mos
skip it!
mos
0
bye
Output:
Enter the first city name: new
no matched city
Enter the first city name: mos
matched cities:
0. Hermosillo, Mexico
choose a city with a number between 0 and 0, or input other thing to skip
skip it!
skip
Enter the first city name: mos
matched cities:
0. Hermosillo, Mexico
```

```
choose a city with a number between 0 and 0, or input other thing to skip
0
Enter the second city name: bye
```

```
andyb1ance@LAPTOP-M482O4RG:/mnt/d/code/cpp/SUSTech_CS205_cpp/assignment2$ ./assignment2
Enter the first city name: new
no matched city
Enter the first city name: mos
matched cities:
0. Hermosillo, Mexico
choose a city with a number between 0 and 0, or input other thing to skip
skip it!
skip
Enter the first city name: mos
matched cities:
0. Hermosillo, Mexico
choose a city with a number between 0 and 0, or input other thing to skip
0
Enter the second city name: bye
andyb1ance@LAPTOP-M482O4RG:/mnt/d/code/cpp/SUSTech_CS205_cpp/assignment2$
```

Part 4 - Difficulties & Solutions

I want to write the part that reads CSV files and reads user input as a function, but because I am not familiar with functions passing array Pointers, I have not solved it yet