

# CS205 C/ C++ Programming - Lab Assignment4

**Name:**金冬阳 (Jin Dongyang)

**SID:**11911221

## Part 1 - Analysis

环境:

CLion 2020.3.1x64

操作系统: Windows10 (家庭中文版64位)

CPU: Intel Core i5-9300H

内存: 8G

**Define some functions, and then consider how to solve some data that violate the input requirements**

## Part 2 - Code

Q1-Q3

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <fstream>
```

```

#include <string>
using namespace std;
#define SAVE_FILE "lab_records.csv"
struct Student {
    int sid;
    int labs[14];
};
vector<int> g_studentid{};
vector<Student> g_students{};
int readInt(int below);

int generateSID();

Student generateStudent();

int randomInt(int below,int upper);

using namespace std;

void question1() {
    cout << "please input student number: ";

    int n = readInt(1);

    for (int i = 0; i < n; ++i) {
        g_students.push_back(generateStudent());
    }

    for (auto &student : g_students) {
        cout << student.sid << ": ";

        for (int i = 0; i < 14; i++) {
            if(i==13){
                cout << student.labs[i]<<endl;
            }else{
                cout << student.labs[i]<<" ";
            }
        }
    }
}

void question2() {
    cout << "The following students are absent from class twice or
more:" << endl;

```

```

    for (auto &student : g_students) {
        int absentNumber = 0;
        for (auto &lab : student.labs) {
            if (lab == 0) {
                absentNumber++;
            }
        }

        if (absentNumber >= 2) {
            cout << student.sid << " is absent for " << absentNumber <<
" times" << endl;
        }

    }
}

void question3() {
    ofstream file(SAVE_FILE);
    if (!file.good()) {
        cout << "failed" << endl;
        exit(0);
    }

    for (auto &student : g_students) {
        file << student.sid << ",";

        for (int i = 0; i < 14; i++) {
            if(i==13){
                file << student.labs[i]<<endl;
            }else{
                file << student.labs[i]<<",";
            }
        }
    }

    file.close();

    cout<< SAVE_FILE << " is successfully writen." << endl;
}

int main() {

    question1();

```

```

    cout << endl;

    question2();

    cout << endl;

    question3();

    cout << endl;

    return 0;
}

Student generateStudent() {
    Student s{};
    int sid = generateSID();
    while (true){
        if(std::count(g_studentid.begin(), g_studentid.end(), sid))
        { //Prevent repeated student numbers
            sid = generateSID();
        }else{
            g_studentid.push_back(sid);
            break;
        }
    }
    s.sid = sid;
    for (auto &lab : s.labs) {
        lab = randomInt(0, 5);
    }
    return s;
}

int randomInt(int below, int upper) {
    int acontain_bcontain=(rand()%(upper-below+1))+ below;

    return acontain_bcontain;
}

int generateSID() {
    int year = randomInt(2000, 2020);
    int number = randomInt(0, 9999);
    int sid=year * 10000 + number;

    return sid;
}

```

```

int readInt(int below) {
    int number;
    cin>>number;
    if (cin.fail()) {
        cout << "input error, program exit" << endl;
        exit(0);
    }else if(number<below){
        cout << "input is too small" << endl;
        exit(0);
    }
    return number;
}

```

Q4

```

#include <iostream>
#include <algorithm>
#include <fstream>
#include <string>
using namespace std;
#define READ_FILE "lab_records.csv"

using namespace std;

int main() {
    ifstream file(READ_FILE);
    if (!file.good()) {
        cout<< READ_FILE <<" can not be opened." << endl;
        exit(0);
    }

    int student_number = 0;
    int lab[15]{};
    int total = 0;

    string line_csv;
    while (getline(file, line_csv)) {

        string delimiter = ",";
        int pos = 0;
        int col = 0;
    }
}

```

```

        string temp[15];

        while ((pos = line_csv.find(delimiter)) != string::npos) {
            temp[col++] = line_csv.substr(0, pos);
            line_csv.erase(0, pos + delimiter.length());
        }
        temp[col] = line_csv;

        student_number++;

        for (int i = 0; i < 14; i++) {
            lab[i] += stoi(temp[i+1]);
            total += stoi(temp[i+1]);
        }

    }
    file.close();
    double total_average = (double)total / student_number / 14;
    cout<<"total average points are "<<total_average<<endl;
    for (int i = 0; i < 14; i++) {
        double average = (double)lab[i] / student_number;
        if (average < total_average) {
            cout << "lab" << i+1 << " "<<average<<endl;
        }
    }
    return 0;
}

```

Q5

```

#include <iostream>
#include <string>
#include <algorithm>
#define EXIT 5
#define INVALID 6

using namespace std;

int findNum(const string &toSearch);
const char *g_commands[] = {"start", "stop", "restart", "reload",
    "status", "exit"};
int main() {

```

```

string input;
while (true) {
    cout << "> ";
    getline(cin, input);
    int index = findNum(input);
    switch (index) {
        case EXIT:
            cout << "program exit" << endl;
            return 0;
        case INVALID:
            cout << "Invalid command" << endl;
            break;
        default:
            cout << "command " << input << " recognized" << endl;
            break;
    }
}

int findNum(const string &toSearch) {
    return distance(cbegin(g_commands), find(cbegin(g_commands),
    cend(g_commands), toSearch));
}

```

## Part 3 - Result & Verification

### Test case #1:

Q1-Q3

input:

**output:**

```
please input student number: 12
20208467: 4, 4, 5, 4, 0, 0, 4, 2, 5, 5, 1, 3, 1, 5
20131942: 3, 0, 3, 0, 2, 3, 4, 4, 3, 2, 2, 5, 5, 0
20081538: 3, 4, 5, 1, 1, 0, 5, 3, 2, 3, 3, 2, 3, 1
20086868: 5, 2, 4, 3, 3, 1, 5, 3, 1, 4, 4, 5, 2, 0
20150106: 4, 2, 4, 4, 2, 3, 2, 3, 4, 2, 0, 3, 3, 2
20152623: 0, 4, 0, 2, 4, 2, 5, 4, 0, 3, 2, 1, 5, 4
20112082: 3, 5, 3, 5, 1, 0, 0, 0, 3, 2, 5, 0, 5, 5
20029072: 0, 3, 5, 3, 3, 0, 4, 0, 5, 0, 5, 5, 3, 4
20202052: 2, 4, 3, 4, 4, 4, 3, 5, 1, 3, 1, 5, 3, 3
20148909: 1, 2, 5, 0, 2, 4, 2, 4, 3, 2, 0, 5, 0, 5
20016359: 0, 5, 2, 3, 1, 3, 2, 2, 1, 2, 2, 4, 0, 3
20063199: 0, 4, 1, 0, 5, 1, 0, 2, 0, 2, 1, 5, 2, 1
```

The following students are absent from class twice or more:

```
20208467 is absent for 2 times
20131942 is absent for 3 times
20152623 is absent for 3 times
20112082 is absent for 4 times
20029072 is absent for 4 times
20148909 is absent for 3 times
20016359 is absent for 2 times
20063199 is absent for 4 times
```

lab\_records.csv is successfully written.

**Test case #2:**

Q4

**input:**

```
null
```

**output:**



```
total average points are 2.63095
lab1 2.08333
lab4 2.41667
lab5 2.33333
lab6 1.75
lab9 2.33333
lab10 2.5
lab11 2.16667
```

## Test case #3:

### input:

```
start
stop
eeee
exit
```

### output:

```
> start
command start recognized
> stop
command stop recognized
> eeee
Invalid command
> exit
program exit
```

## Part 4 - Difficulties & Solutions

---

nothing