

Jeremy Scheuerman

COSC 120

Project 2

- a. We want to take in data from the .dat file for each student and calculate the grades. It needs to account for quiz scores, labs projects midterms and attendance. After taking these all in it needs to find the averages and calculate the letter grade. It then needs to output to a separate .dat file with the first and last name and final grade.
- b. Student Array student student_array holds list of student objects which holds the quiz,lab,project, and midterm grades

Quiz array int quiz_array holds student's quizzes

Project array int project_array holds student's projects

Quiz array int quiz_array holds student's quizzes

Quiz array int quiz_array holds student's quizzes
- c. Function get_averages (student John) uses the student object as input then goes through it and calculates the grades of all the quizzes tests exams and projects saves the averages in variables also within the student object.
- d. Function final _scores (student John) Once again uses the student object for input and manipulates data within it) uses the info in the variables and calculates the final score
- e. Function debug_output (Student Object) takes in data from the student object and outputs it to the console for easy debugging

Source Code:

#include <iostream>

```
#include <iomanip>

#include <string>

#include <fstream>

using namespace std;

const int QUIZ_SIZE = 10;

const int LAB_SIZE = 10;

const int PROJECT_SIZE = 3;

const int MIDTERM_SIZE = 3;

const int SECTION_SIZE = 19;

//define constants

class Student {

    //student object to hold their info

public:

    string name_first = "";

    string name_last = "";

    //name

    int quiz_array[QUIZ_SIZE];

    int lab_array[LAB_SIZE];

    int project_array[PROJECT_SIZE];

    int midterm_array[MIDTERM_SIZE];

    //arrays for assignments

    double quiz_avg = 0;

    double lab_avg = 0;
```

```
double project_avg = 0;

double midterm_avg = 0;

//averages for assignments

int final_exam = 0;

double attendance = 0;

char letter_grade;

double total_grade = 0;

};

void get_average(Student &John) {

    double quiz_total = 0;

    double lab_total = 0;

    double project_total = 0;

    double midterm_total = 0;

    //total scores for temporary

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

        quiz_total += John.quiz_array[i];

    }

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

        project_total += John.project_array[i];

    }

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

        lab_total += John.lab_array[i];
```

```

    }

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

        midterm_total += John.midterm_array[i];

    }

    //populate totals from arrays

    John.quiz_avg = quiz_total / QUIZ_SIZE;

    John.lab_avg = lab_total / LAB_SIZE;

    John.project_avg = project_total / PROJECT_SIZE;

    John.midterm_avg = midterm_total / MIDTERM_SIZE;

    //find averages

}

void final_score(Student &John) {

    const int grade_90 = 90;

    const int grade_80 = 80;

    const int grade_70 = 70;

    const int grade_60 = 60;

    const int grade_0 = 0;

    //declare constants

    double quiz_scaled = John.quiz_avg * .1;

    double lab_scaled = John.lab_avg * .1;

    double midterm_scaled = John.midterm_avg * .3;

    double project_scaled = John.project_avg * .2;

    double final_scaled = John.final_exam * .3;

```

```

//scale points

double particp_points = John.attendance * 100;

John.total_grade = quiz_scaled + lab_scaled + project_scaled + final_scaled
                  + midterm_scaled - particp_points;

//calculate total points

if (John.total_grade >= grade_90) {

    if (John.project_avg >= 90) {

        John.letter_grade = 'A';

    } else {

        John.letter_grade = 'B';

    }

} else if ((John.total_grade >= grade_80)

          && (John.total_grade < grade_90)) {

    if (John.project_avg >= 80) {

        John.letter_grade = 'B';

    } else {

        John.letter_grade = 'C';

    }

} else if ((John.total_grade >= grade_70)

          && (John.total_grade < grade_80)) {

    if (John.project_avg >= 70) {

        John.letter_grade = 'C';

    } else {

```

```

        John.letter_grade = 'D';
    }
} else if ((John.total_grade >= grade_60)
           && (John.total_grade < grade_70)) {
    John.letter_grade = 'D';
} else if ((John.total_grade >= grade_0) && (John.total_grade < grade_60)) {
    John.letter_grade = 'F';
} else {
    John.letter_grade = 'P';
    //if the grade cannot be calculated it gets P for pending
}

//calculates the letter grade

}

void debug_output(Student &John) {
    //a short debug function to assist me in making the program
    cout << endl << "check";
    cout << endl << "Quiz " << John.quiz_avg;
    cout << endl << "exams " << John.midterm_avg;
    cout << endl << "projects " << John.project_avg;
    cout << endl << "labs " << John.lab_avg;
    cout << endl << "Final Exam " << John.final_exam;
    cout << endl << "total " << John.total_grade;

```

```
}
```

```
int main() {
```

```
    string line;
```

```
    Student student_list[SECTION_SIZE];
```

```
    ifstream dataIn;
```

```
    ofstream dataOut;
```

```
//define file stream
```

```
    dataIn.open("grade120.dat");
```

```
    dataOut.open("letter120.dat");
```

```
    if (dataIn.fail()) {
```

```
        cout << "Opening the file failed";
```

```
    }
```

```
    getline(dataIn, line);
```

```
    getline(dataIn, line);
```

```
//skip first 2 lines
```

```
    for (int i = 0; i < SECTION_SIZE; i++) {
```

```
        dataIn >> student_list[i].name_first;
```

```
        dataIn >> student_list[i].name_last;
```

```
        for (int j = 0; j < 10; j++) {
```

```
            dataIn >> student_list[i].quiz_array[j];
```

```
        }
```

```

    for (int j = 0; j < 10; j++) {

        dataIn >> student_list[i].lab_array[j];

    }

    for (int j = 0; j < 3; j++) {

        dataIn >> student_list[i].project_array[j];

    }

    for (int j = 0; j < 3; j++) {

        dataIn >> student_list[i].midterm_array[j];

    }

    dataIn >> student_list[i].final_exam;

    dataIn >> student_list[i].attendance;

    get_average(student_list[i]);

    final_score(student_list[i]);

    //do calculations

}

dataOut << "First Name      Last Name    Final Grade" << endl;

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

    dataOut << left << student_list[i].name_first << "          " << left

        << setw(21) << student_list[i].name_last << left

        << student_list[i].letter_grade << endl;

    //print out for each student

}

```



```
cout << "The grades have been calculated and sent to letter120.dat";  
return 0;  
}
```