

CS 150 Lab 10 More Functions

The main objective of today's lab is to continue writing functions.

Be sure your output looks exactly like the specified output.

Be sure to submit the completed project to CS150-02 Lab by Friday at 5pm.

Show the instructor or TA your solution after each function is implemented below.

Lab 10.1

Write a complete C++ program in a project **10_1_Functions** that will process a class of student grades. The data file "grades.txt" is as follows:

```
5 10.0 10.0 15.0 50.0 100.0
smith 9.0 10.0 12.0 46.0 88.0
jones 8.0 7.0 15.0 48.0 75.0
done 0.0 0.0 0.0 0.0 0.0
```

The first line is the number of scores for each student and the maximum number of points on each assignment. Each subsequent line of data is a student's last name followed by the scores they achieved on each assignment.

I have written most of the program that will process each student's grades. You must write each function definition and add a small amount of code in the main that will loop until the student "done" is read from the file. In the main loop, you will need to call the function readScores and printIndividualResults with the correct arguments.

The output results must look exactly like the following:

```
*****
*           Grading Program
*****
   Name    Average
   ----    -
   smith    89.19
   jones    82.70
Press any key to continue . . .
```

```
#include <iostream>
#include <string>
#include <fstream>
#include <iomanip>

using namespace std;

void printTitle (string);
void printColumnHeadings ();
double readScores (ifstream &, int);
void printIndividualResults (string, double, double);

int main ()
{
    const string GRADING_TITLE = "Grading Program";
    const string FILE_OPEN_ERROR = "Error opening file";
    const string GRADES_TXT = "grades.txt";
```

```

const string NO_STUDENT = "done";

double totalPointsPossible, totalPointsAchieved;
int numScores;
ifstream inputFile;
string name;

inputFile.open (GRADES_TXT);
if (inputFile.fail ())
{
    cout << FILE_OPEN_ERROR << endl;
    exit (EXIT_FAILURE);
}

printTitle (GRADING_TITLE);
printColumnHeadings ();

inputFile >> numScores;
totalPointsPossible = readScores (inputFile, numScores);

// Write loop logic here to process each student in the class. The pseudo-code is
// read a name
// while (name is not the last student)
// {
//   call readScores to get the points the student has achieved
//   call printIndividualResults to print the student's results
//   get the next name
// }

inputFile.close ();
return EXIT_SUCCESS;
}

//Function definitions go here

```

Optional Challenge:

Show a histogram of grades

Histogram

```

A's: ***
B's: *
C's: ****
D's: *
E's: *

```

- 1) Your program is to compile without any errors or warnings.
- 2) The entire program is to be documented as well as each function is to be documented.
- 3) I have placed the undocumented main.cpp in the CS150-02 Public folder on Turing. Create a new project and then copy this code into your main.cpp.

Once your project is complete, place your solution PUNetIDLabs into the CS150-02 Drop folder on Turing. Your solution is to have ALL previous projects completely working and correct.