Homework 6

100 Points

Functions

Assignments & Grading:

A. 20Points – Circle: class exercise // **Hw_6A.cpp**
B. 20Points – Rectangle: Find and fix errors // **Hw_6B.cpp**
C. 60Points – Performers// See below // **Hw_6C.cpp**.

Run each program once and save the output at the end of the source file as a comment. Compress the source file, input and output files and upload the compressed file: 22A_LastName_FirstName_H6.zip

Project Performers

A particular talent competition has five judges, each of whom awards a score between 0 and 10 to each performer. Fractional scores, such as 8.3, are allowed. A performer's final score is determined by dropping the highest and the lowest score received then averaging the three remaining scores. Write a program that calculates the score of each performer and determines the winner. Assume there's only one winner!

Input: Read data from an input file named **performers.txt**. First create the input file: copy and paste the following data into a new text file: **performers.txt**

```
John   7.0 7.8 7.1 7.9 7.5
Dan    8.3 2.9 9.8 9.2 9.7
Mary   8.3 8.0 8.9 7.9 8.5
Andrew 9.1 8.9 9.0 8.7 9.1
Sue    9.0 8.8 9.1 9.7 9.3
Linda  8.2 8.4 8.9 9.3 8.5
David  9.0 5.6 8.9 9.9 7.3
Ann    7.9 8.2 7.6 8.1 8.0
```

You may assume that a performer is identified by its unique first name (no two performers have the same first name).

Output: Write the number of participants, the name of the winner and the winner's score to the screen. Write to a file named **results.txt** the names of the performers and their final score.

// NOTE: You must write documentation for each function – see next page

Design: Your program should include 7 functions in addition to **main()**, as shown below:

```
printInfo – to display a welcome message and some information about the program
open the input file
open the output file
for each customer in the file repeat the following:
        getScores – to read a performer’s name and scores // see getRectangle in 6B
        calculateScore – to calculate a performer’s score; // see calculateRectangle
        this function calls two functions:
                findLowest – to find the lowest score
                findHighest – to find the highest score
        writeScore – to write to the output file a performer’s score  // see writeRectangle
close the input file
close the output file
printEnd – to display an end-of-the-program message
```

NOTE: You must write documentation for each function. Choose a style or create your own style and be consistent. A one-line comment is not an acceptable style.

```
//*********************************************************
// This function accepts the number of hours worked as an
// argument and returns the employee’s pay for
// non-overtime hours.
//*********************************************************

/**********************************************************
This function accepts the number of hours worked as an argument and
returns the employee’s pay for
non-overtime hours.
*/

/**~*~*
This function accepts the number of hours worked as an argument and
returns the employee’s pay for non-overtime hours.
*~**/
```