

Write a Program: Competition Score (continued)

Use Assignment 7a as your starting point. The description of this assignment is rather long, but you have already written most of the code – you just need to modularize it appropriately and account for multiple athletes.

Your goal is to create a program that will determine average scores for a group of **up to 10** athletes in a competition. There are 10 judges who each award scores between 0 and 10. The lowest and highest scores are thrown out, and each athlete's score is the average of the eight remaining scores.

I provided 2 scorecard files for you to use. The scorecards have two lines for each athlete. The first line has the name of the athlete and the second line has 10 space-separated scores between 0 and 10. The number of athletes in the 2 scorecards is different.

Your program should get the scorecard file name from the user, read the scorecard, and create the same output as in assignment 7a for each athlete.

Your program should have the following:

- Include 4 comment lines at the top: description of the program, author, section, and date. (1 point)
- Global named constants for the maximum number of athletes and the number of scores per athlete. (.5 points)
- Local named constants as appropriate. Remember that each function is responsible for its own functionality. (1 point)
- An array to hold the names of the athletes, an array to hold error messages if an error is encountered while reading data for individual athletes, and a two-dimensional array to hold the scores from all judges for all athletes. (1.5 points)
- A function (Function 1) that reads the athletes' names and scores from the scorecard.
 - The function should store the names in the name array (1 point)

- The function should store the scores in the two- dimensional score array. Do not forget to *ignore* (hint) the end line character from the stream after you read each line with the scores so you can read the next athlete's name correctly. (2 points)
- For each athlete, if a score outside the valid range is encountered the function should store an error message in the error array and proceed to reading the next athlete's data. Otherwise, an empty string should be stored in the error array. (1 point)
- The function should return the number of athletes it successfully read from the scorecard (including the ones that will be disqualified because of invalid scores). This number can be 10 if the scorecard has at least 10 athletes, or less if the end of file is encountered sooner. You should not attempt to read the data for more than 10 athletes. **Hint1:** An input stream object will evaluate to false after any read operation that encounters an EOF. **Hint2:** Function getline() returns its first parameter (the input stream from which it reads). (2 points)
- In this assignment (unlike 7a) you don't need to handle the scenario when there's an insufficient number of scores for an athlete. If, while reading the scores, the stream evaluates to false just assume that reading the current athlete's data is unsuccessful and return the number of athletes you have already read by that point. **Hint:** That number will be your loop counter value. (1 point)
- A function (Function 2) that processes the scores and computes the average for one athlete.
 - This function finds the highest and the lowest scores to throw away, computes the average of the eight remaining scores, and returns it. (2 points)
 - It also displays the 10 scores to the console and reports the two scores that were dropped. (1 point)
- The main function should:
 - Ask the user to enter the filename of the scorecard. (1 point)
 - Call Function 1 to read the scorecard and get the number of athletes whose data was successfully read. (1 point)
 - For each such athlete: (3 points)
 - display the athlete's name

- check if there was an error validating the scores and display a message that the athlete is disqualified
- if the error message is an empty string call Function 2 to compute the average score
- display the average score on the console

Notes:

- When you run your program, you should test it with both scorecards. You should also test your program when the scorecard file is not present. This is a total of 3 runs. Insufficient console output is a **1-point** deduction.
- Pay attention to where you create and how you initialize your variables. Unsafe code is a **1-point** deduction.
- Remember to create named constants for what's appropriate. Use the standard convention for constant names. Improper/insufficient use of named constants is an up to **1-point** deduction.
- Comment your code. Uncommented code is a **1-point** deduction.

Example Output:

First run

```
Please enter the scorecard name: scorecard.txt
```

```
ERROR: the scorecard scorecard.txt could not be opened
```

Second run

```
Please enter the scorecard name: scorecard1.txt
```

```
Mirabella Jones's results:
```

```
7.50, 8.80, 7.00, 8.10, 8.00, 9.80, 9.30, 8.90, 9.10, 9.00
```

```
The highest score of 9.80 and the lowest score of 7.00 were  
dropped
```

```
The average score is 8.59
```

```
Ruth Mendez's results:
```

```
9.80, 8.50, 6.00, 8.80, 8.60, 7.10, 7.80, 8.00, 7.20, 8.30
```

```
The highest score of 9.80 and the lowest score of 6.00 were  
dropped
```

```
The average score is 8.04
```

```
Melvin Ingram's results:
```

```
9.90, 7.30, 6.30, 7.00, 6.80, 6.20, 8.90, 9.50, 6.50, 6.00
```

The highest score of 9.90 and the lowest score of 6.00 were dropped

The average score is 7.31

Tara Silva's results:

8.10, 7.10, 9.40, 7.20, 9.20, 6.40, 9.50, 8.40, 6.70, 6.60

The highest score of 9.50 and the lowest score of 6.40 were dropped

The average score is 7.84

Joann Gardner's results:

6.90, 8.00, 8.70, 8.90, 9.10, 7.50, 8.20, 6.30, 8.40, 6.20

The highest score of 9.10 and the lowest score of 6.20 were dropped

The average score is 7.86

Jeff Barnes's results:

6.40, 7.20, 8.30, 8.60, 7.90, 6.00, 7.10, 6.70, 9.50, 9.90

The highest score of 9.90 and the lowest score of 6.00 were dropped

The average score is 7.71

Lucille Dixon's results:

9.50, 6.50, 9.30, 9.40, 8.50, 8.70, 6.20, 9.70, 8.70, 8.20

The highest score of 9.70 and the lowest score of 6.20 were dropped

The average score is 8.60

Krista James's results:

8.40, 9.40, 8.10, 6.30, 6.10, 8.60, 9.60, 9.10, 9.90, 8.80

The highest score of 9.90 and the lowest score of 6.10 were dropped

The average score is 8.54

Naomi Sanders's results:

7.00, 7.20, 8.70, 9.10, 9.60, 6.60, 9.40, 9.80, 8.40, 7.60

The highest score of 9.80 and the lowest score of 6.60 were dropped

The average score is 8.38

Ricky McCarthy's results:

9.80, 7.20, 9.00, 8.50, 6.20, 6.50, 9.10, 8.40, 8.10, 8.70

The highest score of 9.80 and the lowest score of 6.20 were dropped

The average score is 8.19

Third run

Please enter the scorecard name: scorecard2.txt

Mirabella Jones's results:

7.50, 8.80, 7.00, 8.10, 8.00, 9.80, 9.30, 8.90, 9.10, 9.00

The highest score of 9.80 and the lowest score of 7.00 were dropped

The average score is 8.59

Ruth Mendez's results:

9.80, 8.50, 6.00, 8.80, 8.60, 7.10, 7.80, 8.00, 7.20, 8.30

The highest score of 9.80 and the lowest score of 6.00 were dropped

The average score is 8.04

Invalid scores

Melvin Ingram is disqualified

Tara Silva's results:

8.10, 7.10, 9.40, 7.20, 9.20, 6.40, 9.50, 8.40, 6.70, 6.60

The highest score of 9.50 and the lowest score of 6.40 were dropped

The average score is 7.84

Invalid scores

Joann Gardner is disqualified

Jeff Barnes's results:

6.40, 7.20, 8.30, 8.60, 7.90, 6.00, 7.10, 6.70, 9.50, 9.90

The highest score of 9.90 and the lowest score of 6.00 were dropped

The average score is 7.71