# Assignment Description

You will write a program that processes the results from an esports tournament.  Read the username and score from scores.txt (a sample will be provided in the starter code). As you read data from the file, you should be keeping track of the top 3 players. Display the average score and the gold, silver, and bronze medal winners at the end of the program.

# GitHub URL (optional)

<https://github.com/wesleyhixon/Programming-Assignments/tree/b7c236934c180f012565605378e19cc4840f5136/M04%20Programming%20Assignment%201>

# Readme Documentation

Input Information: Input is a .txt file containing a list of usernames followed by their tournament score

Output Information: Output is a formatted table containing the usernames of each competitor along with their username. In addition, the top 3 winners are output as well as the average score and number of competitors.

# Flowchart Screen Shots (optional)

Screen shot(s) here

# UML and Use Case Diagrams (optional)

Screen shot(s) here

# Source Code of All files (.h, .cpp)

1. #include <iostream>
2. #include <iomanip>
3. #include <fstream>
4. #include <istream>
5. #include <string>
6. #include <sstream>
7. using namespace std;
8. /\*
9. Program Name: High Score
10. Author: Wesley Hixon
11. Date last updated: 6/24/2024
12. Purpose: To process results from an esports tournament, finding the top 3 scores.
13. \*/
15. int main(){
17. // Creating the table header
18. cout << "IvyGames Tournament Results:" << endl;
19. cout << setw(35) << setfill('-') << "-" << endl;
20. cout << setfill(' ');
21. cout << "| " << setw(15) << left << "Username" << "| " << setw(15) << "Score" << "|"<< endl;
22. cout << setw(35) << setfill('-') << "-" << endl;
23. cout << setfill(' ');
24. // Declaring variables
25. string fileName = "scores.txt";
26. string line;
27. int bestScore = 0, secondBestScore = 0, thirdBestScore = 0, averageScore = 0, totalCompetitors = 0, totalScore = 0;
28. string goldMedalWinner, silverMedalWinner, bronzeMedalWinner;
29. // Opening the file containing the tournament scores
30. ifstream scoreFile;
31. scoreFile.open(fileName);
32. // Looping through each line of the tournament file
33. while(getline(scoreFile, line)){
34. string userName;
35. int userScore;
37. // Creating a stringstream variable from the line to extract the data
38. stringstream ss(line);
39. // Extracting data
40. ss >> userName >> userScore;
41. // Outputting next line in the table
42. cout << setfill(' ');
43. cout << "| " << setw(15) << left << userName << "| " << setw(15) << userScore << "|"<< endl;
44. cout << setw(35) << setfill('-') << "-" << endl;
45. // Increment the total num of competitors
46. // And add the player's score to the total score
47. // This will be useful to calculate the average score at the end
48. totalCompetitors++;
49. totalScore += userScore;
50. if(userScore >= bestScore){
51. // If a new best score is found,
52. // Displace the previous second best score to 3nd place
53. thirdBestScore = secondBestScore;
55. // and displace the previous best score to 2nd place.
56. secondBestScore = bestScore;
58. // Finally, replace the previous best with the new best.
59. bestScore = userScore;
61. // Do the same for the usernames
62. bronzeMedalWinner = silverMedalWinner;
63. silverMedalWinner = goldMedalWinner;
64. goldMedalWinner = userName;
65. }
66. else if(userScore < bestScore && userScore >= secondBestScore){
67. // If a new second best is found,
68. // displace the previous second best score to third
69. thirdBestScore = secondBestScore;
71. // and replace the second best
72. secondBestScore = userScore;
73. // Do the same for the usernames
74. bronzeMedalWinner = silverMedalWinner;
75. silverMedalWinner = userName;
76. }
77. else if(userScore < secondBestScore && userScore >= thirdBestScore){
78. // If a new third best score is found, replace it
79. thirdBestScore = userScore;
80. // Do the same for the username
81. bronzeMedalWinner = userName;
82. }
83. }
84. // Calculate the average score
85. averageScore = totalScore/totalCompetitors;
86. // Output the winners
87. cout << "The gold medal winner is " << goldMedalWinner << " with " << bestScore << " points." << endl;
88. cout << "The silver medal winner is " << silverMedalWinner << " with " << secondBestScore << " points." << endl;
89. cout << "The bronze medal winner is " << bronzeMedalWinner << " with " << thirdBestScore << " points." << endl;
91. // Output the average score along with the number of competitors
92. cout << "The average score was " << averageScore << " over a total of " << totalCompetitors << " competitors." << endl;
94. // Provide sincere congratulations
95. cout << "Congratulations to all the winners! See you at the next tournament.";
96. return 0;
97. }

# Three Use Case Screen Shots

Only included part of the table because it’s so long it doesn’t fit in the screenshot.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

Please make sure all screen shots and text are clearly viewable for faster / easier grading!