**Problem #1**

**Player Ranking**

You need to process the input file to gather some information about players. Then you need to compute the ranking of the players and summarize the information in another output file. Store the information of each player in an appropriately designed structure. Also, after processing of a file is done, remember to close it.

Following information is stored in the input file:

* **playerInfo.txt**: Line #1 contains the total number of players, *n*. Each of next *n* lines contains first name (no spaces) , total # of match played by the player, and total score of the player.

Computations:

* *Average = total\_score / # of match\_played*
* Ranking = better average means better ranking.
* **Sort the players according to ranking.**

Your output is another file as described below:

* **ranking.txt**: This file should contain player information in increasing order of ranking. For each player, a line contains ranking, followed by first name, # of matches, total score, and average.

Write the necessary code. Then write necessary data files to test your code rigorously. Then ask your teacher to evaluate your assignment.

**Mark Distribution:**

Opening files = 2

Reading files = 5

Designing and Storing in structure = 5

Sorting = 3

Writing on files = 3

Closing files = 2

Sorting Algorithm: Modify it according to your need

1. void bubble\_sort(int list[], int n)
2. {
3. int c, d, t;
5. for (c = 0 ; c < n - 1; c++)
6. {
7. for (d = 0 ; d < n - c - 1; d++)
8. {
9. if (list[d] > list[d+1]){// here you should properly compare the players
10. */\* Swapping \*/*
11. t = list[d];
12. list[d] = list[d+1];
13. list[d+1] = t;
14. }
15. }
16. }
17. }