PDS Lab Section 16

Set A Max Marks=45

(Lab Test 2) –November 10, 2023
Time: 9AM to 11:30AM

The top two lines of your programs must contain the following information:

//Roll No.: <Type in your roll no.>
//Name: <Type in your name>

You have to give names to your C files as specified below and upload them in Moodle well before time. Please read the instructions given below.

Document your programs meaningfully using appropriately named variables and sufficient amount of comments. There will be marks for documentation and proper code indentation.

1. Write a program to manage conduction of indoor game tournaments in a student's hostel. Define a global structure type named **stud**. The members of this structure are roll: unsigned integer, name: character string of length 20, and points: integer. Create four global arrays of **stud** type named **badminton, chess, carrom,** and **tt** of size 10 each. Also define a global array of **stud** type named **tmp** of size 30. Based on this, complete the following functions:

a) **main:** Prompt the user about how many students have participated in the games badminton, chess, carrom, and tt tournaments. Based on this, get the details of the students (roll, name, and points) from the user for the students who have participated in the games of badminton, chess, carrom, and tt and fill the structures of the corresponding arrays, while taking care that there are no duplicate students (identified by roll number) are present in an array. For example, a student with roll number 111 can be present in two arrays, say **badminton** and **carrom** or all arrays **badminton, carrom, chess**, and **tt**, but the student 111 cannot be present twice or more times in the same array, say **tt**. Two or more students can have identical names, but must have different roll numbers. Whenever the user enters details of two students with identical roll numbers, retain the last and ignore the previous entry. Display the four arrays properly formatted.

Next, call the functions **findChamp, findRunnersUp,** and **findConsolationWinner to** display the details of the student who has won maximum number of points in three games (champion), who has won maximum number of points in two games (runners up), and who has won maximum number of points in one of the games (consolation prize winner).

[5 Marks]

b) **findChamp:** This function should identify the students who have scored the maximum points considering their best performance in any three of the four games. Compute total score in the students who have participated in three or more games and store the details of each of these students in the array **tmp.** Students who have played in less than 3 games should not be considered. For each such student, exactly one entry in the **tmp** array should be made. If a student has participated in four

games, his best 3 performance must be considered. While storing the details of a student, make sure that the points for all the three considered games that the student has participated are added up. Display the details of all the students stored in **tmp** array properly formatted (also display the games that they have participated in and the points scored in each). Also display the details of the champion student who has secured the maximum point (also display the games that he/she has participated and the points scored in each). If there are multiple champion students with the same maximum point, then they would be declared joint winners. Display the details of all of the joint winners.

[10 Marks]

c) **findRunnersUp:** This function should store the details of the students who have scored the maximum considering their best performance in any two of the four games in the array **tmp,** and then display the details of these students properly formatted (including the games that they have participated and the points won in each). If a student has participated in more than 2 games, his/her best 2 performance must be considered. Students who have participated in less than two games should not be considered. While storing the details of a student, make sure that the points for the two games that the student has participated are added up. It should also display the details of the runners up student who has secured the maximum point(including the games that he/she has participated and the points won in each). If there are multiple runners up students with the same maximum points, they should be declared joint winners. Display the details of all of them.

[10 Marks]

d) **findConsolationWinner:** This function should store the details of the students who have scored the maximum points considering their best performance in any one of the four games in the array **tmp,** and then display the details of these students (including the games that they have participated and the points won in each) properly formatted. It should also display the details of the consolation prize student who has secured the maximum point (also display the games that he/she has participated and the points won in that). If there are multiple consolation prize winner students with the same maximum point, they should be declared joint winners. Display the details of all of them.

[10 Marks]

e) **displayMeritList:** This function should for each game display the details of all participants in merit order. Whenever there is a tie among a set of participants, the details of the participants in the tie should be displayed in alphabetic order of their name. If the names as well as points are identical, then display the details of those students in order of roll number.

[10 Marks]

Note: The winners for the three awards are independently chosen. That is, the Champion can also get runners up as well as consolation prizes.

---------- End ----------