Ranick Patra

The TCS Global Coding Contest

00 Hr 02 Min 25 Sec

- Guidelines
- Coding Area
- Editor | Compile & Run History
- Submissions
- Feedback Form
- Result
- Dashboard

- Graphs

# Coding Area

A B C D E F G H

ONLINE EDITOR (B)

## Score Keeping

### Problem Description

N chess engines are competing in a tournament. Each engine will compete with every other engine in round robin manner in each round.

You have to find the engine that scores the most (highest rank) and the least scores (lowest ranked). Also answer the queries which will be asked.

The participants are seeded and their seed number corresponds to the participant's number. For example, if the participant is P1, then the seed number of that participant is 1.

The ranking of the engines is based on the total scores received across all the rounds, with one that scored the highest being the highest ranked, and the one that scored the least being lowest ranked.

The engines having same score will be ranked according to the following rules:

1. The participant which is having higher score with black pieces against the corresponding participant, with whom the score is tied, will be placed higher.

2. If the participants have same score with black pieces while playing against each other, then the participant who is seeded lower will be ranked higher than the other participant.

The output will be the participant id and score of the highest and lowest ranked engine, with their scores. In addition, responses to the queries need to be output.

### Constraints

0<N<21

0<S<19

M<1001

### Input Format

First line of input will contain the number of participants, N.

Second line will contain the number of rounds, R.

Next ((n*(n-1)*R)/2) will contain 4 data items corresponding to

<PwhitePblackScoreroundNumber>

Where

· Pwhite is the name of Engine playing white

· Pblack is the name of Engine playing black

· Score will be from the set {1-0, 0-1 or 0.5-0.5}

· roundNumber corresponds to the round number

Example: <P1 P2 1-0 1>

Here,

P1 is the participant playing in white, 1 here represent the seeding of the participant.

P2 is the participant playing in black, 2 here also represent the seeding of the participant.

1-0 is the score of the match, the scores can either be (1-0, 0-1, 0.5-0.5)

roundNumber is the round in which the match is played.

Next line contains the number of queries, M.

Next M lines contains the queries.

Queries will be of two types:

1. If the length of the query is 2, then we have to find the difference between the scores of participants after all the rounds. Example, 2 1

We will have to find the difference between the scores of 2 and 1 after all the rounds are over.

2. If the length of the query is 3, then we have to find the difference between the cumulative scores of participants after that round is over. Example, 2 1 4

We will have to find the difference between the scores of 2 and 1 after fourth round is over.

## + Output

First Line contains the participant's name, who gets the highest rank along with its score delimited by ':'

Second Line contains the participant's name, who gets the lowest rank along with its score delimited by ':'

Next M Lines, contains the answers to the queries asked.

## Test Case

## Explanation

**Example 1**

Input:

3

3

P1 P2 0.5-0.5 1

P1 P3 0-1 1

P2 P3 0.5-0.5 1

P2 P1 1-0 2

P3 P1 0-1 2

P3 P2 0.5-0.5 2

P1 P2 0-1 3

P1 P3 0.5-0.5 3

P2 P3 0-1 3

2

3 1

2 1 2

Output:

P3:3.5

P1:2.0

0.0

1.0

Explanation:

P3 and P2 have same score but P3 will have higher rank because, P3 has won more (while playing as black) against P2 when they played head to head.

Also, in the query 3 1, means the difference between the scores of 3 and 1 after all the rounds are over that is 0 as P3 has won one match, P1 has won one match and one match was drawn between them.

Also, in the query 2 1 2, means the difference between the scores of 2 and 1 after second round is over that is 1 as match between them is a draw in first round and P2 won the game in second round.

**Example 2**

Input:

4

2

P2 P1 0.5-0.5 1

P3 P1 0.5-0.5 1

P4 P1 1-0 1

P3 P2 1-0 1

P4 P2 0-1 1

P4 P3 0.5-0.5 1

P1 P2 0-1 2

P1 P3 1-0 2

P1 P4 0.5-0.5 2

P2 P3 1-0 2

P2 P4 0-1 2

P3 P4 0.5-0.5 2

3

3 2 2

1 2

3 2 1

Output:

P4:3.5

P3:2.5

0.0

1.0

1.0

Explanation:

Here P4 and P2 have the same score, also they have same score when they play as black against each other but P4 is ranked higher because it is seeded lower than P2.

## Upload Solution [ Question : B ]

☐ I, **ranick patra** confirm that the answer submitted is my own.

☐ Took help from online sources (attributions)

Choose a File ...



© 2018 Tata Consultancy Services Limited. All Rights Reserved.