MOBA MATCHMAKING

Cloudflight Coding Contest
30th October 2020

Ever since the early 2000s, a shift in paradigm when it comes to sports has taken place. The last winter olympiad has taken place in 2042. The last summer olympics in 2052. The environmental impact of hosting such events grew and grew and no sustainable solution was ever found. Even though in 2020 the esports market covered just under 5% of the global sports figures, in 2050, that number was 38% and in 2075, just under 74%. It has now become clear that the future of sports will no longer involve vaulting and running but other, less physically but more mentally demanding activities.

The MOBA (Multiplayer-Online-Battle-Arena) game genre has been a staple of this esports movement ever since the beginning. It involves two teams of 4-6 players (depending on the game) battling it out in an arena. Each player has one hero which they control. In between each other, games had a hard time with matchmaking within their training seasons. There was never a unifying solution for ranking MOBA players which could be applicable for all games. Great players were often times grouped up with bad ones and thus create unbalanced teams.

CODING CONTEST

YOUR TASK

- Find an efficient solution for MOBA matchmaking
- Ranking (but not only) should play a big role in how teams are formed before a training match is started


LEVEL 1

Corpo:

Alright, listen up rookie. I might look like just another suit and tie but before you can even call me “sir”, you’ll have to prove your worth. I’m a busy guy, being paid for being busy.

Here’s something to start off with. Fail, and I’ll find another who won’t.

Task for Level 1:

Find the maximum score!

- One possible way to find the best player is to consider the scores from each game as an absolute value and choose the player that scored the most points.
- In the input file you will find a list of one versus one game, complete with the points scored by each player.
- Find the **player** that obtained the **highest score** in any game.
- Print that player's id and its top score.
- In case there are players with the **same score**, the one with the **lowest player id** comes before the others.

	Input	Output
Format	gameCount playerCount player1Id scorePlayer1 player2Id scorePlayer2 *… repeated for each game...* player1Id scorePlayer1 player2Id scorePlayer2	maxScorePlayerId maxScore
Types	gameCount - Integer. Represents the number of games played for this test case playerCount - Integer. Represents the number of players involved in this test case. player1Id - Integer. Id of the first player involved in the game. player1Id < playerCount. player1Score - Integer. Number of points obtained by the first player. player2Id - Integer. Id of the second player involved in the game. player2Id < playerCount. player2Score - Integer. Number of points obtained by the second player.	maxScorePlayerId - Integer. Id of the player that obtained the maximum score. maxScore - Integer. The max score obtained throughout all matches.
Example	*9 6* *0 997 5 999* *1 280 5 981* *2 893 4 914* *2 264 4 188* *0 187 1 23* *0 777 1 414* *2 993 5 883* *4 596 5 894* *1 97 5 770*	*5 999*


PLAY
TO
WIN
MOBA
MATCHMAKING
GOOD LUCK