Programming
Homework assignment 2
Data processing. File input and output

General information

Deadline: March 29th, 23:59:59
Submission: through the Canvas LMS as a zip archive

Introduction

In the second graded homework assignment, you will implement an application for managing a sporting tournament. You can pick a sport and tournament of your own choice provided it meets the following requirements:

- It should be a tournament held in group(s) (not a knockout system with quarterfinals, semi-finals and final). In football, this can be a national league, like the Russian premier league or Italian Serie A, or a single champions league group.
- Points awarded for each outcome (win, draw, loss) are the same throughout the competition.
- The tournament results are determined only by the final ranking of teams. No additional matches are played to break the ties.

The input file should will contain results of matches played in a tournament. You need to aggregate these results, rank teams according to the rules of the tournament and show the ranking tables on request from the user.

The goal of the assignment is to practice with Python data structures and file input-output.

Description of tasks

Basic task (1 tournament): 5 points

- Pick a web resource that exports results of sporting events in text formats. As a default choice, you can take results of a football league from the following resource. At this step, you are free to interact with your groupmates and discuss the sporting side of the problem (rules of different tournaments, how results are calculated, etc). However, the implementation of each program should still be an individual result.

The source data file should not contain a ranking table - this should be computed by your application.

- Create a new Python project. Make the source file part of the project. Load data from the file to program memory. The source file may contain many statistical points about each match, you just need the following:
  - Date of the match
  - Team names
  - Score

  Aggregate all results and make a ranking of teams. Add a separate file to your project that shows the rules of the competition, which are especially important for breaking the ties (when two or more teams score the same number of points). For example, the following rules are used for the English Premier league (source):

  If any clubs finish with the same number of points, their position in the Premier League table is determined by goal difference, then the number of goals scored. If the teams still cannot be separated, they will be awarded the same position in the table.

- Make a simple request-response workflow for the user with the following functions:
  - Show all matches of a given team
  - Show matches played on a given date
  - Show the ranking table (should be visualized as a true table with columns properly aligned). The ranking table should contain the following columns:
    * Ranking place - note, that this can be the same for several teams if the teams cannot be separated
    * Team name
    * Number of games played
    * Number of wins
    * Number of draws
    * Number of losses
    * Goal difference
    * Points

    The exact indicators may differ depending on the chosen sport and tournament.

Extra task 1 (Multiple tournaments with the same ranking regulations): 3 points

Adjust your program in such a way that it loads data from multiple source files, each storing results of 1 tournament. The rules for ranking teams within each tournament are the same and can be hardcoded in your application.

The program should provide a 2-level menu. The top level menu is a selection of tournaments. The inner menu is the same as in the basic task (3 functions)

Extra task 2 (Multiple tournaments with different ranking regulations): 2 points

Adjust your program, so that it supports managing tournaments with different tie breaking rules. For example, compare the rules used in the English Premier league (above) with the ones used in the Italian Serie A league:

> *If two or more teams are tied in points (for any place), the deciding tie-breakers are as follows:*
>
> 1. *Head-to-head records (results and points)*
> 2. *Goal difference of head-to-head games*
> 3. *Goal difference overall*
> 4. *Higher number of goals scored*

Please make sure you add all the required files that describe the ranking regulations for each tournament that you decide to include in your project.

To get the maximal score for this task, you need to come up with an approach that would minimize hard-coding of tournament rules in the application code.

General requirements

Please read the following requirements carefully, as they affect the final grade. Failure to comply with these will cause a downgrade:

- All the source data should be loaded once at program startup
- Use functions (and possibly, modules) to achieve a better structure of your program
- Add protection against incorrect user input, absense of files or errors in their format

Homework defense

After all submissions are received, the course teachers may choose to schedule an obligatory defense to determine the final grade of each student.