Programming project / C++
Table Tennis Management System

Student 1: Muntean Darius

Student 2: Hajdu Patrick

I.Task

Student 1 responsibilities:

- Designed the data structure: Match, Player
- Implemented:
    - add() function (parses and validates match input)
    - File input and output for storing match data in matches.txt
    - File operations on player.txt
    - update_stats(), add_player(), player_in_file()

Student 2 responsibilities:

- Handled user input, program control flow (menu(), main())
- Implemented:
    - leaderboard() function
    - Sorting logic for leaderboard (by win-rate, wins, losses)
    - player() function to show individual stats and match history
    - header(), clear() and terminal interface formatting (colors, layout)

II.Data structures used by the team

- Player (Stores a player's statistics):
    - string name
    - int wins, losses
- Match (Represents match result):
    - string player1, string player2
    - int score1, int score2

Programming project / C++

Table Tennis Management System

III.File Structure

- players.txt – stores every player's statistics
  - Format: <player_name>,<wins>,<losses>
- matches.txt – stores every correct match result introduced
  - Format: <player1>,<score1>,<score2>,<player2>

IV.Interaction with the Executable

App 1 has the following options:

- Add Match – for adding a new match result
  - <player1> <score1> / <player2> <score2>
  - Scores 1 and 2 must be positive and smaller or equal to 21, with 1 score needing to be exactly 21

App 2 has the following options:

- Leaderboard
  - By win-rate : 1
  - By wins: 2
  - By losses: 3
  - Match history: <player_name> - showcases player match history and statistics