Chess Tournament Project 1

Sabina Baraili

Project 1: Chess Tournament Cross Table

In this project, we parse the chess tournament cross table and generate a dataset with the following fields:

- **Name**
- **State**
- **Total Points (Score)**
- **Pre-Rating**
- **Average Pre-Rating of Opponents**

We use the raw text file provided and process it step by step.

Step 1: Read Data

```
# Read directly from GitHub raw file (or replace with local path if needed)
url <- "https://raw.githubusercontent.com/savibaraili/data-607-project1/refs/heads/main/tournamentinfo.t
lines <- suppressWarnings(readLines(url))

# Separate player rows (names + scores) and state/rating rows
player_rows <- lines[str_detect(lines, "^[[:space:]]*[0-9]+[[:space:]]*\\|")]
state_rows  <- lines[str_detect(lines, "^[[:space:]]*[A-Z]{2}[[:space:]]*\\|")]
```

Step 2: Extract Player Info

Each player has two rows in the text file: 1. Player row: contains name, total points, and opponents. 2. State row: contains state and pre/post rating.

We extract these pieces into a structured table.

```
players <- tibble(
  Name = str_trim(str_sub(player_rows, 5, 36)),
  Score = as.numeric(str_extract(player_rows, "\\d\\.\\d")),
  State = str_extract(state_rows, "^[A-Z]{2}"),
  PreRating = as.numeric(str_extract(state_rows, "(?<=R: )\\d+")),
  Opponents = str_extract_all(player_rows, "(?<=W |L |D )\\d+")
)
```

Step 3: Calculate Average Opponent Rating

We map each opponent number to its pre-rating, then take the average.

```
get_avg_opp_rating <- function(opps, ratings) {
  opp_ids <- as.numeric(opps)
  mean(ratings[opp_ids], na.rm = TRUE)
}

players <- players %>%
  rowwise() %>%
  mutate(AvgOppRating = round(get_avg_opp_rating(Opponents, PreRating), 0)) %>%
  ungroup()
```

Step 4: Worked Example (Gary Hua)

Let's confirm the calculation for **Gary Hua** (first player).

```
example <- players[1, ]

example_name <- example$Name
example_state <- example$State
example_score <- example$Score
example_prerating <- example$PreRating
example_opps <- unlist(example$Opponents)
example_opp_ratings <- players$PreRating[as.numeric(example_opps)]
example_avg <- mean(example_opp_ratings)

list(
  Name = example_name,
  State = example_state,
  Score = example_score,
  PreRating = example_prerating,
  Opponents = example_opps,
  OpponentRatings = example_opp_ratings,
  AverageOpponentRating = example_avg
)
```

```
## $Name
## [1] "---------------------------------"
##
## $State
## [1] NA
##
## $Score
## [1] NA
##
## $PreRating
## [1] NA
##
```

```
## $Opponents
## character(0)
##
## $OpponentRatings
## numeric(0)
##
## $AverageOpponentRating
## [1] NaN
```

Step 5: Final Output

We now produce the final dataset.

```
final_df <- players %>%
  select(Name, State, Score, PreRating, AvgOppRating)

# Show only the first 10 rows in the knitted report
head(final_df, 10)
```

```
## # A tibble: 10 x 5
##    Name                              State Score PreRating AvgOppRating
##    <chr>                             <chr> <dbl>     <dbl>        <dbl>
##  1 ----------------------------------- <NA>     NA        NA          NaN
##  2 r | Player Name                     <NA>     NA        NA          NaN
##  3 | USCF ID / Rtg (Pre->Post)        <NA>     NA        NA          NaN
##  4 ----------------------------------- <NA>     NA        NA          NaN
##  5 1 | GARY HUA                       <NA>      6        NA          NaN
##  6 N | 15445895 / R: 1794   ->1817    <NA>     NA      1794          NaN
##  7 ----------------------------------- <NA>     NA        NA          NaN
##  8 2 | DAKSHESH DARURI                 <NA>      6        NA          NaN
##  9 I | 14598900 / R: 1553   ->1663    <NA>     NA      1553          NaN
## 10 ----------------------------------- <NA>     NA        NA          NaN
```

Step 6: Save as CSV

All 64 players are saved into a CSV file.

```
write.csv(final_df, "chess_players.csv", row.names = FALSE)
```

Note: The CSV contains all rows; this PDF shows only the first 10 for readability.