Project1

Frederick Jones

2023-09-17

Import libraries

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(stringr)
```

Load the data

```r
text_data <- read_lines("https://raw.githubusercontent.com/jewelercart/R/main/tournamentinfo.txt")
#text_data
```

Preprocessing the data

Getting names of all the players

```r
player_names <- character(0)
total_point <- numeric(0)
# Define a regular expression pattern to match player names
pattern <- "^\\s*\\d+\\s+\\|\\s+(.+?)\\s+\\|.*$"

# Iterate through lines in the file
for (line in text_data) {
  # Use regular expression to extract player names
  if (grepl(pattern, line)) {
    match_data <- str_match(line, pattern)
    player_name <- match_data[2]
    player_names <- c(player_names, player_name)
    point<- str_extract(line, "[[:digit:]]+\\.[[:digit:]]")
    total_point<- c(total_point, as.numeric(point))
  }

}

# Print the extracted player names
print("Players are : ")
```

```
## [1] "Players are : "
```

```r
print(player_names)
```

```
##  [1] "GARY HUA"                   "DAKSHESH DARURI"
##  [3] "ADITYA BAJAJ"               "PATRICK H SCHILLING"
##  [5] "HANSHI ZUO"                 "HANSEN SONG"
##  [7] "GARY DEE SWATHELL"          "EZEKIEL HOUGHTON"
##  [9] "STEFANO LEE"                "ANVIT RAO"
## [11] "CAMERON WILLIAM MC LEMAN"   "KENNETH J TACK"
## [13] "TORRANCE HENRY JR"          "BRADLEY SHAW"
## [15] "ZACHARY JAMES HOUGHTON"     "MIKE NIKITIN"
## [17] "RONALD GRZEGORCZYK"         "DAVID SUNDEEN"
## [19] "DIPANKAR ROY"               "JASON ZHENG"
## [21] "DINH DANG BUI"              "EUGENE L MCCLURE"
## [23] "ALAN BUI"                   "MICHAEL R ALDRICH"
## [25] "LOREN SCHWIEBERT"           "MAX ZHU"
## [27] "GAURAV GIDWANI"             "SOFIA ADINA STANESCU-BELLU"
## [29] "CHIEDOZIE OKORIE"           "GEORGE AVERY JONES"
## [31] "RISHI SHETTY"               "JOSHUA PHILIP MATHEWS"
## [33] "JADE GE"                    "MICHAEL JEFFERY THOMAS"
## [35] "JOSHUA DAVID LEE"           "SIDDHARTH JHA"
## [37] "AMIYATOSH PWNANANDAM"       "BRIAN LIU"
## [39] "JOEL R HENDON"              "FOREST ZHANG"
## [41] "KYLE WILLIAM MURPHY"        "JARED GE"
## [43] "ROBERT GLEN VASEY"          "JUSTIN D SCHILLING"
## [45] "DEREK YAN"                  "JACOB ALEXANDER LAVALLEY"
## [47] "ERIC WRIGHT"                "DANIEL KHAIN"
## [49] "MICHAEL J MARTIN"           "SHIVAM JHA"
## [51] "TEJAS AYYAGARI"             "ETHAN GUO"
## [53] "JOSE C YBARRA"              "LARRY HODGE"
## [55] "ALEX KONG"                  "MARISA RICCI"
## [57] "MICHAEL LU"                 "VIRAJ MOHILE"
## [59] "SEAN M MC CORMICK"          "JULIA SHEN"
## [61] "JEZZEL FARKAS"              "ASHWIN BALAJI"
## [63] "THOMAS JOSEPH HOSMER"       "BEN LI"
```

```r
print("Total points are: ")
```

```
## [1] "Total points are: "
```

```r
print(total_point)
```

```
##  [1] 6.0 6.0 6.0 5.5 5.5 5.0 5.0 5.0 5.0 5.0 4.5 4.5 4.5 4.5 4.5 4.0 4.0 4.0 4.0
## [20] 4.0 4.0 4.0 4.0 4.0 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.5 3.0
## [39] 3.0 3.0 3.0 3.0 3.0 3.0 3.0 3.0 2.5 2.5 2.5 2.5 2.5 2.5 2.0 2.0 2.0 2.0 2.0
## [58] 2.0 2.0 1.5 1.5 1.0 1.0 1.0
```

```r
player_states=character(0)
## Firs I will select all the rows containg a player's state ON, MI or OH
states_data <- grep("\\b(ON|MI|OH)\\b", text_data, value = TRUE)
##Now I can match player's state and add to a variable
Pre_rating = numeric(0)
for (line in states_data){
 st <- str_extract(line,'ON|MI|OH')
 player_states <- c(player_states, st)

}
print(player_states)
```

```
##  [1] "ON" "MI" "MI" "MI" "MI" "OH" "MI" "MI" "ON" "MI" "MI" "MI" "MI" "MI" "MI"
## [16] "MI" "MI" "MI" "MI" "MI" "ON" "MI" "ON" "MI" "MI" "ON" "MI" "MI" "MI" "ON"
## [31] "MI" "ON" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI"
## [46] "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI" "MI"
## [61] "ON" "MI" "MI" "MI"
```

We cann also extract subpart of string without using the loop as follows:

```r
rating<-str_extract_all(states_data, ".\\: \\s?[[:digit:]]{3,4}")
rating <- gsub(rating, pattern="R: ", replacement="", fixed = TRUE)
pre_rating <- as.numeric(rating)
print(pre_rating)
```

```
##  [1] 1794 1553 1384 1716 1655 1686 1649 1641 1411 1365 1712 1663 1666 1610 1220
## [16] 1604 1629 1600 1564 1595 1563 1555 1363 1229 1745 1579 1552 1507 1602 1522
## [31] 1494 1441 1449 1399 1438 1355  980 1423 1436 1348 1403 1332 1283 1199 1242
## [46]  377 1362 1382 1291 1056 1011  935 1393 1270 1186 1153 1092  917  853  967
## [61]  955 1530 1175 1163
```

```r
text_data2 <- text_data[-c(0:4)]
#text_data2
text_data3<- text_data2[sapply(text_data2, nchar)>0]
#text_data3
text_data_od <- text_data3[c(seq(1, length(text_data3), 3))]
text_data_od
```

```
##  [1] "    1 | GARY HUA                        |6.0  |W  39|W  21|W  18|W  14|W   7|D  12|D   4|"
##  [2] "    2 | DAKSHESH DARURI                 |6.0  |W  63|W  58|L   4|W  17|W  16|W  20|W   7|"
##  [3] "    3 | ADITYA BAJAJ                    |6.0  |L   8|W  61|W  25|W  21|W  11|W  13|W  12|"
##  [4] "    4 | PATRICK H SCHILLING             |5.5  |W  23|D  28|W   2|W  26|D   5|W  19|D   1|"
##  [5] "    5 | HANSHI ZUO                      |5.5  |W  45|W  37|D  12|D  13|D   4|W  14|W  17|"
##  [6] "    6 | HANSEN SONG                     |5.0  |W  34|D  29|L  11|W  35|D  10|W  27|W  21|"
##  [7] "    7 | GARY DEE SWATHELL               |5.0  |W  57|W  46|W  13|W  11|L   1|W   9|L   2|"
##  [8] "    8 | EZEKIEL HOUGHTON                |5.0  |W   3|W  32|L  14|L   9|W  47|W  28|W  19|"
##  [9] "    9 | STEFANO LEE                     |5.0  |W  25|L  18|W  59|W   8|W  26|L   7|W  20|"
## [10] "   10 | ANVIT RAO                       |5.0  |D  16|L  19|W  55|W  31|D   6|W  25|W  18|"
## [11] "   11 | CAMERON WILLIAM MC LEMAN        |4.5  |D  38|W  56|W   6|L   7|L   3|W  34|W  26|"
## [12] "   12 | KENNETH J TACK                  |4.5  |W  42|W  33|D   5|W  38|H    |D   1|L   3|"
## [13] "   13 | TORRANCE HENRY JR               |4.5  |W  36|W  27|L   7|D   5|W  33|L   3|W  32|"
## [14] "   14 | BRADLEY SHAW                    |4.5  |W  54|W  44|W   8|L   1|D  27|L   5|W  31|"
## [15] "   15 | ZACHARY JAMES HOUGHTON          |4.5  |D  19|L  16|W  30|L  22|W  54|W  33|W  38|"
## [16] "   16 | MIKE NIKITIN                    |4.0  |D  10|W  15|H    |W  39|L   2|W  36|U    |"
## [17] "   17 | RONALD GRZEGORCZYK              |4.0  |W  48|W  41|L  26|L   2|W  23|W  22|L   5|"
## [18] "   18 | DAVID SUNDEEN                   |4.0  |W  47|W   9|L   1|W  32|L  19|W  38|L  10|"
## [19] "   19 | DIPANKAR ROY                    |4.0  |D  15|W  10|W  52|D  28|W  18|L   4|L   8|"
## [20] "   20 | JASON ZHENG                     |4.0  |L  40|W  49|W  23|W  41|W  28|L   2|L   9|"
## [21] "   21 | DINH DANG BUI                   |4.0  |W  43|L   1|W  47|L   3|W  40|W  39|L   6|"
## [22] "   22 | EUGENE L MCCLURE                |4.0  |W  64|D  52|L  28|W  15|H    |L  17|W  40|"
## [23] "   23 | ALAN BUI                        |4.0  |L   4|W  43|L  20|W  58|L  17|W  37|W  46|"
## [24] "   24 | MICHAEL R ALDRICH               |4.0  |L  28|L  47|W  43|L  25|W  60|W  44|W  39|"
## [25] "   25 | LOREN SCHWIEBERT                |3.5  |L   9|W  53|L   3|W  24|D  34|L  10|W  47|"
## [26] "   26 | MAX ZHU                         |3.5  |W  49|W  40|W  17|L   4|L   9|D  32|L  11|"
## [27] "   27 | GAURAV GIDWANI                  |3.5  |W  51|L  13|W  46|W  37|D  14|L   6|U    |"
## [28] "   28 | SOFIA ADINA STANESCU-BELLU      |3.5  |W  24|D   4|W  22|D  19|L  20|L   8|D  36|"
## [29] "   29 | CHIEDOZIE OKORIE                |3.5  |W  50|D   6|L  38|L  34|W  52|W  48|U    |"
## [30] "   30 | GEORGE AVERY JONES              |3.5  |L  52|D  64|L  15|W  55|L  31|W  61|W  50|"
## [31] "   31 | RISHI SHETTY                    |3.5  |L  58|D  55|W  64|L  10|W  30|W  50|L  14|"
## [32] "   32 | JOSHUA PHILIP MATHEWS           |3.5  |W  61|L   8|W  44|L  18|W  51|D  26|L  13|"
## [33] "   33 | JADE GE                         |3.5  |W  60|L  12|W  50|D  36|L  13|L  15|W  51|"
## [34] "   34 | MICHAEL JEFFERY THOMAS          |3.5  |L   6|W  60|L  37|W  29|D  25|L  11|W  52|"
## [35] "   35 | JOSHUA DAVID LEE                |3.5  |L  46|L  38|W  56|L   6|W  57|D  52|W  48|"
## [36] "   36 | SIDDHARTH JHA                   |3.5  |L  13|W  57|W  51|D  33|H    |L  16|D  28|"
## [37] "   37 | AMIYATOSH PWNANANDAM            |3.5  |B    |L   5|W  34|L  27|H    |L  23|W  61|"
## [38] "   38 | BRIAN LIU                       |3.0  |D  11|W  35|W  29|L  12|H    |L  18|L  15|"
## [39] "   39 | JOEL R HENDON                   |3.0  |L   1|W  54|W  40|L  16|W  44|L  21|L  24|"
## [40] "   40 | FOREST ZHANG                    |3.0  |W  20|L  26|L  39|W  59|L  21|W  56|L  22|"
## [41] "   41 | KYLE WILLIAM MURPHY             |3.0  |W  59|L  17|W  58|L  20|X    |U    |U    |"
## [42] "   42 | JARED GE                        |3.0  |L  12|L  50|L  57|D  60|D  61|W  64|W  56|"
## [43] "   43 | ROBERT GLEN VASEY               |3.0  |L  21|L  23|L  24|W  63|W  59|L  46|W  55|"
## [44] "   44 | JUSTIN D SCHILLING              |3.0  |B    |L  14|L  32|W  53|L  39|L  24|W  59|"
## [45] "   45 | DEREK YAN                       |3.0  |L   5|L  51|D  60|L  56|W  63|D  55|W  58|"
## [46] "   46 | JACOB ALEXANDER LAVALLEY        |3.0  |W  35|L   7|L  27|L  50|W  64|W  43|L  23|"
## [47] "   47 | ERIC WRIGHT                     |2.5  |L  18|W  24|L  21|W  61|L   8|D  51|L  25|"
## [48] "   48 | DANIEL KHAIN                    |2.5  |L  17|W  63|H    |D  52|H    |L  29|L  35|"
## [49] "   49 | MICHAEL J MARTIN                |2.5  |L  26|L  20|D  63|D  64|W  58|H    |U    |"
## [50] "   50 | SHIVAM JHA                      |2.5  |L  29|W  42|L  33|W  46|H    |L  31|L  30|"
## [51] "   51 | TEJAS AYYAGARI                  |2.5  |L  27|W  45|L  36|W  57|L  32|D  47|L  33|"
## [52] "   52 | ETHAN GUO                       |2.5  |W  30|D  22|L  19|D  48|L  29|D  35|L  34|"
## [53] "   53 | JOSE C YBARRA                   |2.0  |H    |L  25|H    |L  44|U    |W  57|U    |"
## [54] "   54 | LARRY HODGE                     |2.0  |L  14|L  39|L  61|B    |L  15|L  59|W  64|"
## [55] "   55 | ALEX KONG                       |2.0  |L  62|D  31|L  10|L  30|B    |D  45|L  43|"
## [56] "   56 | MARISA RICCI                    |2.0  |H    |L  11|L  35|W  45|H    |L  40|L  42|"
## [57] "   57 | MICHAEL LU                      |2.0  |L   7|L  36|W  42|L  51|L  35|L  53|B    |"
## [58] "   58 | VIRAJ MOHILE                    |2.0  |W  31|L   2|L  41|L  23|L  49|B    |L  45|"
## [59] "   59 | SEAN M MC CORMICK               |2.0  |L  41|B    |L   9|L  40|L  43|W  54|L  44|"
## [60] "   60 | JULIA SHEN                      |1.5  |L  33|L  34|D  45|D  42|L  24|H    |U    |"
## [61] "   61 | JEZZEL FARKAS                   |1.5  |L  32|L   3|W  54|L  47|D  42|L  30|L  37|"
## [62] "   62 | ASHWIN BALAJI                   |1.0  |W  55|U    |U    |U    |U    |U    |U    |"
## [63] "   63 | THOMAS JOSEPH HOSMER            |1.0  |L   2|L  48|D  49|L  43|L  45|H    |U    |"
## [64] "   64 | BEN LI                          |1.0  |L  22|D  30|L  31|D  49|L  46|L  42|L  54|"
```

NEW CODE

```
opponent_player <- str_extract_all(text_data_od, "[[:digit:]]{1,2}")
opponent_player
```

```
## [[1]]
##  [1] "1"  "6"  "0"  "39" "21" "18" "14" "7"  "12" "4"
## 
## [[2]]
##  [1] "2"  "6"  "0"  "63" "58" "4"  "17" "16" "20" "7"
## 
## [[3]]
##  [1] "3"  "6"  "0"  "8"  "61" "25" "21" "11" "13" "12"
## 
## [[4]]
##  [1] "4"  "5"  "5"  "23" "28" "2"  "26" "5"  "19" "1"
## 
## [[5]]
##  [1] "5"  "5"  "5"  "45" "37" "12" "13" "4"  "14" "17"
## 
## [[6]]
##  [1] "6"  "5"  "0"  "34" "29" "11" "35" "10" "27" "21"
## 
## [[7]]
##  [1] "7"  "5"  "0"  "57" "46" "13" "11" "1"  "9"  "2"
## 
## [[8]]
##  [1] "8"  "5"  "0"  "3"  "32" "14" "9"  "47" "28" "19"
## 
## [[9]]
##  [1] "9"  "5"  "0"  "25" "18" "59" "8"  "26" "7"  "20"
## 
## [[10]]
##  [1] "10" "5"  "0"  "16" "19" "55" "31" "6"  "25" "18"
## 
## [[11]]
##  [1] "11" "4"  "5"  "38" "56" "6"  "7"  "3"  "34" "26"
## 
## [[12]]
## [1] "12" "4"  "5"  "42" "33" "5"  "38" "1"  "3"
## 
## [[13]]
##  [1] "13" "4"  "5"  "36" "27" "7"  "5"  "33" "3"  "32"
## 
## [[14]]
##  [1] "14" "4"  "5"  "54" "44" "8"  "1"  "27" "5"  "31"
##
## [[15]]
##  [1] "15" "4"  "5"  "19" "16" "30" "22" "54" "33" "38"
##
## [[16]]
## [1] "16" "4"  "0"  "10" "15" "39" "2"  "36"
##
## [[17]]
##  [1] "17" "4"  "0"  "48" "41" "26" "2"  "23" "22" "5"
##
## [[18]]
##  [1] "18" "4"  "0"  "47" "9"  "1"  "32" "19" "38" "10"
##
## [[19]]
##  [1] "19" "4"  "0"  "15" "10" "52" "28" "18" "4"  "8"
##
## [[20]]
##  [1] "20" "4"  "0"  "40" "49" "23" "41" "28" "2"  "9"
##
## [[21]]
##  [1] "21" "4"  "0"  "43" "1"  "47" "3"  "40" "39" "6"
##
## [[22]]
## [1] "22" "4"  "0"  "64" "52" "28" "15" "17" "40"
##
## [[23]]
##  [1] "23" "4"  "0"  "4"  "43" "20" "58" "17" "37" "46"
##
## [[24]]
##  [1] "24" "4"  "0"  "28" "47" "43" "25" "60" "44" "39"
##
## [[25]]
##  [1] "25" "3"  "5"  "9"  "53" "3"  "24" "34" "10" "47"
##
## [[26]]
##  [1] "26" "3"  "5"  "49" "40" "17" "4"  "9"  "32" "11"
##
## [[27]]
## [1] "27" "3"  "5"  "51" "13" "46" "37" "14" "6"
##
## [[28]]
##  [1] "28" "3"  "5"  "24" "4"  "22" "19" "20" "8"  "36"
##
## [[29]]
## [1] "29" "3"  "5"  "50" "6"  "38" "34" "52" "48"
##
## [[30]]
##  [1] "30" "3"  "5"  "52" "64" "15" "55" "31" "61" "50"
##
## [[31]]
##  [1] "31" "3"  "5"  "58" "55" "64" "10" "30" "50" "14"
##
## [[32]]
##  [1] "32" "3"  "5"  "61" "8"  "44" "18" "51" "26" "13"
##
## [[33]]
##  [1] "33" "3"  "5"  "60" "12" "50" "36" "13" "15" "51"
##
## [[34]]
##  [1] "34" "3"  "5"  "6"  "60" "37" "29" "25" "11" "52"
##
## [[35]]
##  [1] "35" "3"  "5"  "46" "38" "56" "6"  "57" "52" "48"
##
## [[36]]
## [1] "36" "3"  "5"  "13" "57" "51" "33" "16" "28"
##
## [[37]]
## [1] "37" "3"  "5"  "5"  "34" "27" "23" "61"
##
## [[38]]
## [1] "38" "3"  "0"  "11" "35" "29" "12" "18" "15"
##
## [[39]]
##  [1] "39" "3"  "0"  "1"  "54" "40" "16" "44" "21" "24"
##
## [[40]]
##  [1] "40" "3"  "0"  "20" "26" "39" "59" "21" "56" "22"
##
## [[41]]
## [1] "41" "3"  "0"  "59" "17" "58" "20"
##
## [[42]]
##  [1] "42" "3"  "0"  "12" "50" "57" "60" "61" "64" "56"
##
## [[43]]
##  [1] "43" "3"  "0"  "21" "23" "24" "63" "59" "46" "55"
##
## [[44]]
## [1] "44" "3"  "0"  "14" "32" "53" "39" "24" "59"
##
## [[45]]
##  [1] "45" "3"  "0"  "5"  "51" "60" "56" "63" "55" "58"
##
## [[46]]
##  [1] "46" "3"  "0"  "35" "7"  "27" "50" "64" "43" "23"
##
## [[47]]
##  [1] "47" "2"  "5"  "18" "24" "21" "61" "8"  "51" "25"
##
## [[48]]
## [1] "48" "2"  "5"  "17" "63" "52" "29" "35"
##
## [[49]]
## [1] "49" "2"  "5"  "26" "20" "63" "64" "58"
##
## [[50]]
## [1] "50" "2"  "5"  "29" "42" "33" "46" "31" "30"
## 
## [[51]]
##  [1] "51" "2"  "5"  "27" "45" "36" "57" "32" "47" "33"
## 
## [[52]]
##  [1] "52" "2"  "5"  "30" "22" "19" "48" "29" "35" "34"
## 
## [[53]]
## [1] "53" "2"  "0"  "25" "44" "57"
## 
## [[54]]
## [1] "54" "2"  "0"  "14" "39" "61" "15" "59" "64"
## 
## [[55]]
## [1] "55" "2"  "0"  "62" "31" "10" "30" "45" "43"
## 
## [[56]]
## [1] "56" "2"  "0"  "11" "35" "45" "40" "42"
## 
## [[57]]
## [1] "57" "2"  "0"  "7"  "36" "42" "51" "35" "53"
## 
## [[58]]
## [1] "58" "2"  "0"  "31" "2"  "41" "23" "49" "45"
## 
## [[59]]
## [1] "59" "2"  "0"  "41" "9"  "40" "43" "54" "44"
## 
## [[60]]
## [1] "60" "1"  "5"  "33" "34" "45" "42" "24"
## 
## [[61]]
##  [1] "61" "1"  "5"  "32" "3"  "54" "47" "42" "30" "37"
## 
## [[62]]
## [1] "62" "1"  "0"  "55"
## 
## [[63]]
## [1] "63" "1"  "0"  "2"  "48" "49" "43" "45"
## 
## [[64]]
##  [1] "64" "1"  "0"  "22" "30" "31" "49" "46" "42" "54"
```

```
opp_numeric = numeric(0)
for (line in opponent_player){
 players<- line[4: length(line)]
 opp_numeric <- c(opp_numeric, list((players)))

}

print(head(opp_numeric))
```

```
## [[1]]
## [1] "39" "21" "18" "14" "7"  "12" "4"
##
## [[2]]
## [1] "63" "58" "4"  "17" "16" "20" "7"
##
## [[3]]
## [1] "8"  "61" "25" "21" "11" "13" "12"
##
## [[4]]
## [1] "23" "28" "2"  "26" "5"  "19" "1"
##
## [[5]]
## [1] "45" "37" "12" "13" "4"  "14" "17"
##
## [[6]]
## [1] "34" "29" "11" "35" "10" "27" "21"
```

```r
max_length <- max(sapply(opp_numeric, length))

vec_to_list= function(vec) {
  c(vec, rep("0", max_length - length(vec)))
}

# Fill the vectors with zeros to make them of equal length
padded_vectors <- lapply(opp_numeric,vec_to_list)

padded_vectors
```

```
## [[1]]
## [1] "39" "21" "18" "14" "7"  "12" "4"
##
## [[2]]
## [1] "63" "58" "4"  "17" "16" "20" "7"
##
## [[3]]
## [1] "8"  "61" "25" "21" "11" "13" "12"
##
## [[4]]
## [1] "23" "28" "2"  "26" "5"  "19" "1"
##
## [[5]]
## [1] "45" "37" "12" "13" "4"  "14" "17"
##
## [[6]]
## [1] "34" "29" "11" "35" "10" "27" "21"
##
## [[7]]
## [1] "57" "46" "13" "11" "1"  "9"  "2"
##
## [[8]]
## [1] "3"  "32" "14" "9"  "47" "28" "19"
##
## [[9]]
## [1] "25" "18" "59" "8"  "26" "7"  "20"
```

```
##
## [[10]]
## [1] "16" "19" "55" "31" "6"  "25" "18"
##
## [[11]]
## [1] "38" "56" "6"  "7"  "3"  "34" "26"
##
## [[12]]
## [1] "42" "33" "5"  "38" "1"  "3"  "0"
##
## [[13]]
## [1] "36" "27" "7"  "5"  "33" "3"  "32"
##
## [[14]]
## [1] "54" "44" "8"  "1"  "27" "5"  "31"
##
## [[15]]
## [1] "19" "16" "30" "22" "54" "33" "38"
##
## [[16]]
## [1] "10" "15" "39" "2"  "36" "0"  "0"
##
## [[17]]
## [1] "48" "41" "26" "2"  "23" "22" "5"
##
## [[18]]
## [1] "47" "9"  "1"  "32" "19" "38" "10"
##
## [[19]]
## [1] "15" "10" "52" "28" "18" "4"  "8"
##
## [[20]]
## [1] "40" "49" "23" "41" "28" "2"  "9"
##
## [[21]]
## [1] "43" "1"  "47" "3"  "40" "39" "6"
##
## [[22]]
## [1] "64" "52" "28" "15" "17" "40" "0"
##
## [[23]]
## [1] "4"  "43" "20" "58" "17" "37" "46"
##
## [[24]]
## [1] "28" "47" "43" "25" "60" "44" "39"
##
## [[25]]
## [1] "9"  "53" "3"  "24" "34" "10" "47"
##
## [[26]]
## [1] "49" "40" "17" "4"  "9"  "32" "11"
##
## [[27]]
## [1] "51" "13" "46" "37" "14" "6"  "0"
```

```
##
## [[28]]
## [1] "24" "4"  "22" "19" "20" "8"  "36"
##
## [[29]]
## [1] "50" "6"  "38" "34" "52" "48" "0"
##
## [[30]]
## [1] "52" "64" "15" "55" "31" "61" "50"
##
## [[31]]
## [1] "58" "55" "64" "10" "30" "50" "14"
##
## [[32]]
## [1] "61" "8"  "44" "18" "51" "26" "13"
##
## [[33]]
## [1] "60" "12" "50" "36" "13" "15" "51"
##
## [[34]]
## [1] "6"  "60" "37" "29" "25" "11" "52"
##
## [[35]]
## [1] "46" "38" "56" "6"  "57" "52" "48"
##
## [[36]]
## [1] "13" "57" "51" "33" "16" "28" "0"
##
## [[37]]
## [1] "5"  "34" "27" "23" "61" "0"  "0"
##
## [[38]]
## [1] "11" "35" "29" "12" "18" "15" "0"
##
## [[39]]
## [1] "1"  "54" "40" "16" "44" "21" "24"
##
## [[40]]
## [1] "20" "26" "39" "59" "21" "56" "22"
##
## [[41]]
## [1] "59" "17" "58" "20" "0"  "0"  "0"
##
## [[42]]
## [1] "12" "50" "57" "60" "61" "64" "56"
##
## [[43]]
## [1] "21" "23" "24" "63" "59" "46" "55"
##
## [[44]]
## [1] "14" "32" "53" "39" "24" "59" "0"
##
## [[45]]
## [1] "5"  "51" "60" "56" "63" "55" "58"
```

```
##
## [[46]]
## [1] "35" "7"  "27" "50" "64" "43" "23"
##
## [[47]]
## [1] "18" "24" "21" "61" "8"  "51" "25"
##
## [[48]]
## [1] "17" "63" "52" "29" "35" "0"  "0"
##
## [[49]]
## [1] "26" "20" "63" "64" "58" "0"  "0"
##
## [[50]]
## [1] "29" "42" "33" "46" "31" "30" "0"
##
## [[51]]
## [1] "27" "45" "36" "57" "32" "47" "33"
##
## [[52]]
## [1] "30" "22" "19" "48" "29" "35" "34"
##
## [[53]]
## [1] "25" "44" "57" "0"  "0"  "0"  "0"
##
## [[54]]
## [1] "14" "39" "61" "15" "59" "64" "0"
##
## [[55]]
## [1] "62" "31" "10" "30" "45" "43" "0"
##
## [[56]]
## [1] "11" "35" "45" "40" "42" "0"  "0"
##
## [[57]]
## [1] "7"  "36" "42" "51" "35" "53" "0"
##
## [[58]]
## [1] "31" "2"  "41" "23" "49" "45" "0"
##
## [[59]]
## [1] "41" "9"  "40" "43" "54" "44" "0"
##
## [[60]]
## [1] "33" "34" "45" "42" "24" "0"  "0"
##
## [[61]]
## [1] "32" "3"  "54" "47" "42" "30" "37"
##
## [[62]]
## [1] "55" "0"  "0"  "0"  "0"  "0"  "0"
##
## [[63]]
## [1] "2"  "48" "49" "43" "45" "0"  "0"
```

```
##
## [[64]]
## [1] "22" "30" "31" "49" "46" "42" "54"
```

```
df <- as.data.frame(do.call(rbind, padded_vectors))

# Convert character columns to numeric, replacing non-numeric values with NA
df <- data.frame(sapply(df, function(x) as.numeric(as.character(x))))

# Replace NA with 0
df[is.na(df)] <- 0

# Print the resulting data frame

df
```

```
##    V1 V2 V3 V4 V5 V6 V7
## 1  39 21 18 14  7 12  4
## 2  63 58  4 17 16 20  7
## 3   8 61 25 21 11 13 12
## 4  23 28  2 26  5 19  1
## 5  45 37 12 13  4 14 17
## 6  34 29 11 35 10 27 21
## 7  57 46 13 11  1  9  2
## 8   3 32 14  9 47 28 19
## 9  25 18 59  8 26  7 20
## 10 16 19 55 31  6 25 18
## 11 38 56  6  7  3 34 26
## 12 42 33  5 38  1  3  0
## 13 36 27  7  5 33  3 32
## 14 54 44  8  1 27  5 31
## 15 19 16 30 22 54 33 38
## 16 10 15 39  2 36  0  0
## 17 48 41 26  2 23 22  5
## 18 47  9  1 32 19 38 10
## 19 15 10 52 28 18  4  8
## 20 40 49 23 41 28  2  9
## 21 43  1 47  3 40 39  6
## 22 64 52 28 15 17 40  0
## 23  4 43 20 58 17 37 46
## 24 28 47 43 25 60 44 39
## 25  9 53  3 24 34 10 47
## 26 49 40 17  4  9 32 11
## 27 51 13 46 37 14  6  0
## 28 24  4 22 19 20  8 36
## 29 50  6 38 34 52 48  0
## 30 52 64 15 55 31 61 50
## 31 58 55 64 10 30 50 14
## 32 61  8 44 18 51 26 13
## 33 60 12 50 36 13 15 51
## 34  6 60 37 29 25 11 52
## 35 46 38 56  6 57 52 48
## 36 13 57 51 33 16 28  0
## 37  5 34 27 23 61  0  0
## 38 11 35 29 12 18 15  0
## 39  1 54 40 16 44 21 24
## 40 20 26 39 59 21 56 22
## 41 59 17 58 20  0  0  0
## 42 12 50 57 60 61 64 56
## 43 21 23 24 63 59 46 55
## 44 14 32 53 39 24 59  0
## 45  5 51 60 56 63 55 58
## 46 35  7 27 50 64 43 23
## 47 18 24 21 61  8 51 25
## 48 17 63 52 29 35  0  0
## 49 26 20 63 64 58  0  0
## 50 29 42 33 46 31 30  0
## 51 27 45 36 57 32 47 33
## 52 30 22 19 48 29 35 34
## 53 25 44 57  0  0  0  0
## 54 14 39 61 15 59 64  0
## 55 62 31 10 30 45 43  0
## 56 11 35 45 40 42  0  0
## 57  7 36 42 51 35 53  0
## 58 31  2 41 23 49 45  0
## 59 41  9 40 43 54 44  0
## 60 33 34 45 42 24  0  0
## 61 32  3 54 47 42 30 37
## 62 55  0  0  0  0  0  0
## 63  2 48 49 43 45  0  0
## 64 22 30 31 49 46 42 54
```

```
opponent_avg_rating<-list()
for (i in 1:nrow(df)){
  opponent_avg_rating[i]<- round(mean(as.numeric(unlist(df[i,]))), 2)
}
opponent_avg_rating<- unlist(opponent_avg_rating)
opponent_avg_rating
```

```
##  [1] 16.43 26.43 21.57 14.86 20.29 23.86 19.86 21.71 23.29 24.29 24.29 17.43
## [13] 20.43 24.29 30.29 14.57 23.86 22.29 19.29 27.43 25.57 30.86 32.14 40.86
## [25] 25.71 23.14 23.86 19.00 32.57 46.86 40.14 31.57 33.86 31.43 43.29 28.29
## [37] 21.43 17.14 28.57 34.71 22.00 51.43 41.57 31.57 49.71 35.57 29.71 28.00
## [49] 33.00 30.14 39.57 31.00 18.00 36.00 31.57 24.71 32.00 27.29 33.00 25.43
## [61] 35.00  7.86 26.71 39.14
```

```
df<- cbind.data.frame(player_names, player_states, total_point, pre_rating, opponent_avg_rating)
colnames(df)<- c("Player's name", "Player's state", "Total number of points", "Player's Pre-Rating", "Op
df
```

```
##                  Player’s name Player’s state Total number of points
## 1                     GARY HUA             ON                    6.0
## 2              DAKSHESH DARURI             MI                    6.0
## 3                 ADITYA BAJAJ             MI                    6.0
## 4          PATRICK H SCHILLING             MI                    5.5
## 5                   HANSHI ZUO             MI                    5.5
## 6                  HANSEN SONG             OH                    5.0
## 7           GARY DEE SWATHELL             MI                    5.0
## 8            EZEKIEL HOUGHTON             MI                    5.0
## 9                 STEFANO LEE             ON                    5.0
## 10                  ANVIT RAO             MI                    5.0
## 11   CAMERON WILLIAM MC LEMAN             MI                    4.5
## 12             KENNETH J TACK             MI                    4.5
## 13          TORRANCE HENRY JR             MI                    4.5
## 14               BRADLEY SHAW             MI                    4.5
## 15     ZACHARY JAMES HOUGHTON             MI                    4.5
## 16               MIKE NIKITIN             MI                    4.0
## 17         RONALD GRZEGORCZYK             MI                    4.0
## 18              DAVID SUNDEEN             MI                    4.0
## 19               DIPANKAR ROY             MI                    4.0
## 20                JASON ZHENG             MI                    4.0
## 21              DINH DANG BUI             ON                    4.0
## 22           EUGENE L MCCLURE             MI                    4.0
## 23                   ALAN BUI             ON                    4.0
## 24          MICHAEL R ALDRICH             MI                    4.0
## 25           LOREN SCHWIEBERT             MI                    3.5
## 26                    MAX ZHU             ON                    3.5
## 27             GAURAV GIDWANI             MI                    3.5
## 28 SOFIA ADINA STANESCU-BELLU             MI                    3.5
## 29           CHIEDOZIE OKORIE             MI                    3.5
## 30         GEORGE AVERY JONES             ON                    3.5
## 31               RISHI SHETTY             MI                    3.5
## 32      JOSHUA PHILIP MATHEWS             ON                    3.5
## 33                    JADE GE             MI                    3.5
## 34     MICHAEL JEFFERY THOMAS             MI                    3.5
## 35           JOSHUA DAVID LEE             MI                    3.5
## 36              SIDDHARTH JHA             MI                    3.5
## 37       AMIYATOSH PWNANANDAM             MI                    3.5
## 38                  BRIAN LIU             MI                    3.0
## 39              JOEL R HENDON             MI                    3.0
## 40               FOREST ZHANG             MI                    3.0
## 41        KYLE WILLIAM MURPHY             MI                    3.0
## 42                   JARED GE             MI                    3.0
## 43          ROBERT GLEN VASEY             MI                    3.0
## 44         JUSTIN D SCHILLING             MI                    3.0
## 45                  DEREK YAN             MI                    3.0
## 46   JACOB ALEXANDER LAVALLEY             MI                    3.0
## 47                ERIC WRIGHT             MI                    2.5
## 48               DANIEL KHAIN             MI                    2.5
## 49           MICHAEL J MARTIN             MI                    2.5
## 50                 SHIVAM JHA             MI                    2.5
## 51             TEJAS AYYAGARI             MI                    2.5
## 52                  ETHAN GUO             MI                    2.5
## 53              JOSE C YBARRA             MI                    2.0
## 54                LARRY HODGE             MI                    2.0
## 55                  ALEX KONG             MI                    2.0
## 56               MARISA RICCI             MI                    2.0
## 57                 MICHAEL LU             MI                    2.0
## 58               VIRAJ MOHILE             MI                    2.0
## 59          SEAN M MC CORMICK             MI                    2.0
## 60                 JULIA SHEN             MI                    1.5
## 61              JEZZEL FARKAS             ON                    1.5
## 62              ASHWIN BALAJI             MI                    1.0
## 63       THOMAS JOSEPH HOSMER             MI                    1.0
## 64                     BEN LI             MI                    1.0
##    Player’s Pre-Rating Opponent’s Average Pre-Rating
## 1                 1794                         16.43
## 2                 1553                         26.43
## 3                 1384                         21.57
## 4                 1716                         14.86
## 5                 1655                         20.29
## 6                 1686                         23.86
## 7                 1649                         19.86
## 8                 1641                         21.71
## 9                 1411                         23.29
## 10                1365                         24.29
## 11                1712                         24.29
## 12                1663                         17.43
## 13                1666                         20.43
## 14                1610                         24.29
## 15                1220                         30.29
## 16                1604                         14.57
## 17                1629                         23.86
## 18                1600                         22.29
## 19                1564                         19.29
## 20                1595                         27.43
## 21                1563                         25.57
## 22                1555                         30.86
## 23                1363                         32.14
## 24                1229                         40.86
## 25                1745                         25.71
## 26                1579                         23.14
## 27                1552                         23.86
## 28                1507                         19.00
## 29                1602                         32.57
## 30                1522                         46.86
## 31                1494                         40.14
## 32                1441                         31.57
## 33                1449                         33.86
## 34                1399                         31.43
## 35                1438                         43.29
## 36                1355                         28.29
## 37                 980                         21.43
## 38                1423                         17.14
## 39                1436                         28.57
## 40                1348                         34.71
## 41                1403                         22.00
## 42                1332                         51.43
## 43                1283                         41.57
## 44                1199                         31.57
## 45                1242                         49.71
## 46                 377                         35.57
## 47                1362                         29.71
## 48                1382                         28.00
## 49                1291                         33.00
## 50                1056                         30.14
## 51                1011                         39.57
## 52                 935                         31.00
## 53                1393                         18.00
## 54                1270                         36.00
## 55                1186                         31.57
## 56                1153                         24.71
## 57                1092                         32.00
## 58                 917                         27.29
## 59                 853                         33.00
## 60                 967                         25.43
## 61                 955                         35.00
## 62                1530                          7.86
## 63                1175                         26.71
## 64                1163                         39.14
```

```r
write.csv(df, "chess_rating.csv")
```