

Ярылгасимов Султан  
Отчет по домашнему заданию №1

**1-задание.** Создать матрицу, ввести имена строк и столбцов, транспонировать ее.  
Код:

```
mat <- matrix(seq(1, 16), nrow = 4, ncol = 4, byrow=TRUE)
colnames(mat) <- seq(10, 40, 10)
rownames(mat) <- c("A", "B", "C", "D")
mat
t(mat)
```

Вывод:

```
> mat <- matrix(seq(1, 16), nrow = 4, ncol = 4, byrow=TRUE)
> colnames(mat) <- seq(10, 40, 10)
> rownames(mat) <- c("A", "B", "C", "D")
> mat
  10 20 30 40
A  1  2  3  4
B  5  6  7  8
C  9 10 11 12
D 13 14 15 16
> t(mat)
   A B  C D
10 1  5  9 13
20 2  6 10 14
30 3  7 11 15
40 4  8 12 16
> |
```

**2-задание.** Создать матрицу из отдельных строк и столбцов, вывести ее размерность.  
Код:

```
a <- c(1, 2, 3, 4)
b <- c("a", "b", "c", "d", "e")
c <- c("word", "letter", "text")
new_mat <- rbind(a, b, c)
new_mat
cbind(a, b, c)
```

Вывод:

```
> a <- c(1, 2, 3, 4)
> b <- c("a", "b", "c", "d", "e")
> c <- c("word", "letter", "text")
> new_mat <- rbind(a, b, c)
Предупреждение:
В rbind(a, b, c) :
  number of columns of result is not a multiple of vector length (arg 1)
> new_mat
  [,1] [,2] [,3] [,4] [,5]
a "1"  "2"  "3"  "4"  "1"
b "a"  "b"  "c"  "d"  "e"
c "word" "letter" "text" "word" "letter"
> cbind(a, b, c)
      a b c
[1,] "1" "a" "word"
[2,] "2" "b" "letter"
[3,] "3" "c" "text"
[4,] "4" "d" "word"
[5,] "1" "e" "letter"
Предупреждение:
В cbind(a, b, c) :
  number of rows of result is not a multiple of vector length (arg 1)
> |
```

Замечание: предупреждения сохранены намеренно для того, чтобы посмотреть как функция ведет себя в случае векторов разных длин.

**3-задание.** Проверить, является элемент [2,2] в этой матрице числом (целым или вещественным), логическим значением, пропуском, не-числом.

Код:

```
class(new_mat[2, 2])
new_mat[2, 2] = 3
new_mat
if (class(new_mat[2, 2]) == "character") {
  "CHR"
} else if (class(new_mat[2, 2] == "integer")){
  "INT"
} else if (class(new_mat[2, 2] == "numeric")) {
  "NUM"
} else if (class(new_mat[2, 2]) == "logical") {
  "LOG"
} else if (new_mat[2, 2] == "NaN") {
  "Not-a-Number"
} else if (new_mat[2, 2] == "NA") {
  "Not applicable"
}
```

Вывод:

```
> class(new_mat[2, 2])
[1] "character"
> new_mat[2, 2] = 3
> new_mat
  [,1] [,2] [,3] [,4] [,5]
a "1"  "2"  "3"  "4"  "1"
b "a"  "3"  "c"  "d"  "e"
c "word" "letter" "text" "word" "letter"
> if (class(new_mat[2, 2]) == "character") {
+   "CHR"
+ } else if (class(new_mat[2, 2] == "integer")){
+   "INT"
+ } else if (class(new_mat[2, 2] == "numeric")) {
+   "NUM"
+ } else if (class(new_mat[2, 2]) == "logical") {
+   "LOG"
+ } else if (new_mat[2, 2] == "NaN") {
+   "Not-a-Number"
+ } else if (new_mat[2, 2] == "NA") {
+   "Not applicable"
+ }
[1] "CHR"
> |
```

Замечание: не уверен, что в задании имелось в виду просто посмотреть функцию class(), но решил заодно проверить и то, как устроен синтаксис if-else.

**4-задание.** Используя команды read.table() и read.csv(), а также file.choose(), загрузить данные из TXT и CSV-файлов и создать из некоторого их подмножества список и таблицу.

Код:

```
match_results <- read.csv(file=file.choose())
match_results <- read.table(file.choose(), header = TRUE, sep = ",", quote =
"\\"", dec = ".",
                           fill = TRUE, comment.char = "")
str(match_results)
match_results[match_results$AwayTeam == "Man United", 1 : 23]
"LAST 10 MAN UNITED RESULTS:"
tail(match_results[match_results$AwayTeam == "Man United", 1 : 23], n = 10)
clubs <- c(levels(match_results$HomeTeam))
```

## "CLUBS PLAYED IN EPL"

### clubs

Вывод:

```
> "LAST 10 MAN UNITED RESULTS:"
[1] "LAST 10 MAN UNITED RESULTS:"
> tail(match_results$match_results$AwayTeam == "Man United", 1 : 23], n = 10)
      Season      DateTime      HomeTeam      AwayTeam      FTHG      FTAG      FTR      HTHG      HTAG      HTR      Referee      HS      AS      HST      AST      HC      AC      HF      AF      HY      AY      HR      AR
10607 2020-21 2021-01-20T20:15:00Z      Fulham      Man United      1      2      A      1      1      D      M Atkinson 12 15      5      5      2      8      6      10      3      1      0      0
10624 2020-21 2021-01-30T17:30:00Z      Arsenal      Man United      0      0      D      0      0      D      M Oliver 17 14      3      3      4      7      9      16      1      3      0      0
10655 2020-21 2021-02-14T14:00:00Z      West Brom      Man United      1      1      D      1      1      D      C Pawson 7 10      3      7      4      5      13      10      5      0      0      0
10680 2020-21 2021-02-28T16:30:00Z      Chelsea      Man United      0      0      D      0      0      D      S Attwell 18 11      6      4      3      4      11      12      1      2      0      0
10686 2020-21 2021-03-03T20:15:00Z      Crystal Palace      Man United      0      0      D      0      0      D      A Marriner 8 11      2      1      4      6      4      13      1      0      0      0
10696 2020-21 2021-03-07T16:30:00Z      Man City      Man United      0      2      A      0      1      A      A Taylor 23 8      6      6      6      2      9      12      0      3      0      0
10731 2020-21 2021-04-11T16:30:00Z      Tottenham      Man United      1      3      A      1      0      H      C Kavanagh 12 12      3      7      7      5      11      15      1      5      0      0
10750 2020-21 2021-04-25T14:00:00Z      Leeds      Man United      0      0      D      0      0      D      C Pawson 6 16      3      4      2      6      21      11      4      1      0      0
10768 2020-21 2021-05-09T14:05:00Z      Aston Villa      Man United      1      3      A      1      0      H      C Kavanagh 11 18      5      7      5      6      13      18      1      1      1      0
10804 2020-21 2021-05-23T16:00:00Z      Wolves      Man United      1      2      A      1      2      A      M Dean 14 9      4      4      6      2      14      3      4      1      0      0
> clubs <- c(levels(match_results$HomeTeam))
> "CLUBS PLAYED IN EPL"
[1] "CLUBS PLAYED IN EPL"
> clubs
[1] "Arsenal"      "Aston Villa"      "Barnsley"      "Birmingham"      "Blackburn"      "Blackpool"      "Bolton"
[8] "Bournemouth"      "Bradford"      "Brighton"      "Burnley"      "Cardiff"      "Charlton"      "Chelsea"
[15] "Coventry"      "Crystal Palace"      "Derby"      "Everton"      "Fulham"      "Huddersfield"      "Hull"
[22] "Ipswich"      "Leeds"      "Leicester"      "Liverpool"      "Man City"      "Man United"      "Middlesbrough"
[29] "Newcastle"      "Norwich"      "Nott'm Forest"      "Oldham"      "Portsmouth"      "QPR"      "Reading"
[36] "Sheffield United"      "Sheffield Weds"      "Southampton"      "Stoke"      "Sunderland"      "Swansea"      "Swindon"
[43] "Tottenham"      "Watford"      "West Brom"      "West Ham"      "Wigan"      "Winbledon"      "Wolves"
> |
```

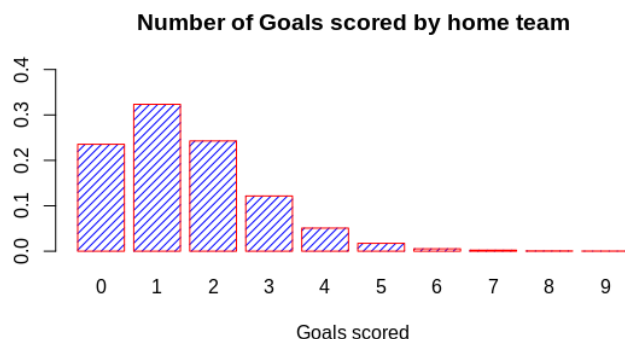
Замечание: таблица выходит очень большая, и поэтому скриншот содержит не весь вывод. Вкратце объясняя, что я делал в этой части кода: прочел файл, функциями указанными в задании, посмотрел как работает таблица и используя это, вывел последние 10 матчей клуба и список всех клубов, которые когда либо играли в АПЛ.

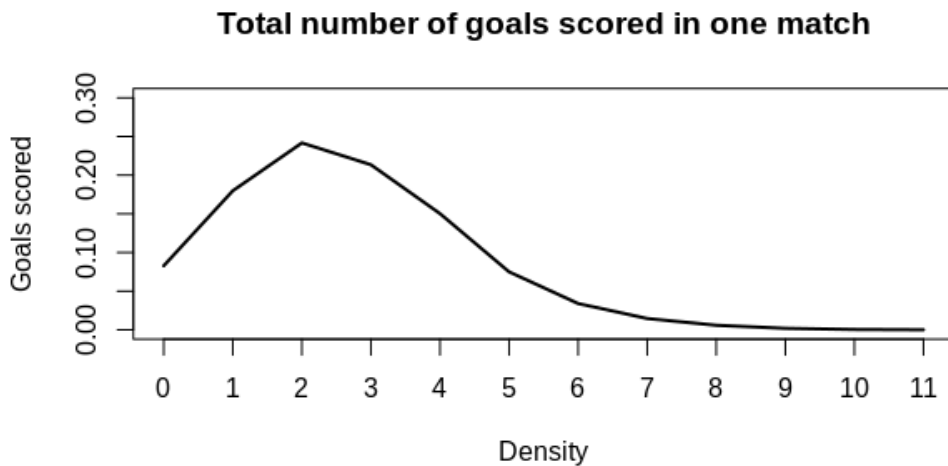
**5-Задание.** Нарисовать график и гистограмму для полученных данных.  
Код:

```
barplot(table(match_results$FTHG)/nrow(match_results),
        main = "Number of Goals scored by home team",
        xlab = "Goals scored",
        density = 20,
        border="red",
        col="blue",
        ylim=c(0, 0.4))

plot(table(match_results$FTHG + match_results$FTAG)/nrow(match_results),
     ylim=c(0, 0.3),
     main = 'Total number of goals scored in one match',
     ylab = 'Goals scored',
     xlab = 'Density',
     type = 'l')
```

Вывод:





**6-задание.** С помощью условного оператора if-else и циклов for, while, repeat проверить условия неотрицательности данных, а также попадания в некоторый диапазон для всех элементов списка и таблицы.

Код:

```
goals_table = data.frame(Season = unique(match_results$Season), Goals_scored = 0)
for(i in seq(1, nrow(match_results), 1)) {
  row <- match_results[i, ]
  factor(row$Season)
  row$Season
  goals_table$Goals_scored[goals_table$Season == row$Season]
  if (row$HomeTeam == 'Man United') {
    goals_table$Goals_scored[goals_table$Season == row$Season] <-
goals_table$Goals_scored[goals_table$Season == row$Season] + row$FTHG
  } else if (row$AwayTeam == 'Man United') {
    goals_table$Goals_scored[goals_table$Season == row$Season] <-
goals_table$Goals_scored[goals_table$Season == row$Season] + row$FTAG
  }
}
```

Вывод:

	Season	Goals_scored
1	1993-94	80
2	1994-95	77
3	1995-96	73
4	1996-97	76
5	1997-98	73
6	1998-99	80
7	1999-00	97
8	2000-01	79
9	2001-02	87
10	2002-03	74
11	2003-04	64
12	2004-05	58
13	2005-06	72
14	2006-07	83
15	2007-08	80
16	2008-09	68
17	2009-10	86
18	2010-11	78
19	2011-12	89
20	2012-13	86
21	2013-14	64
22	2014-15	62
23	2015-16	49
24	2016-17	54
25	2017-18	68
26	2018-19	65
27	2019-20	66
28	2020-21	73

> |