

SPRAWOZDANIE

Zajęcia: Analiza Procesów Ucznia

Prowadzący: prof. dr hab. Vasyl Martsenyuk

Laboratorium: Zajecie 1

Data: 25.02.2019

Temat: "Podstawy języka R"

Wariant 4

Artur Kozieł
Informatyka II stopień,
stacjonarne ,
1 semestr,

1. Polecenie:

Celem jest nabycie podstawowej znajomości języka R rozwiązując zadanie tworzenia i wyświetlenia ramki danych odpowiednio do określonego wariantu.

2. Wprowadzane dane:

- (a) Do zmiennej `a` podstaw wartość wyrażenia $4 * \sin(\pi)$. Do zmiennej `b` podstaw potrójną wartość zmiennej `a`. Wywołaj funkcję sprawdzającą, która z wartości zmiennych jest większa.
- (b) Uruchom i poczytaj dokumentację dla funkcji `max()`.
- (c) Stwórz wektor `a` zawierający liczby od 90 do 115. Policz średnią kwadratów liczb zawartych w wektorze.
- (d) Wyświetl wszystkie funkcje zawierające frazę `max` w swojej nazwie.
- (e) Ustaw dowolny katalog roboczy. Następnie stwórz zmienną `a` zawierającą łańcuch znaków "lodówka z największą pojemnością". Zapisz zmienną `a` obszaru roboczego do pliku w katalogu roboczym. Następnie usuń zmienną `a`. Sprawdź wartość zmiennej `a` (powinno jej brakować). Na końcu wczytaj plik ze zmienną `a` i sprawdź jej wartość.
- (f) Zainstaluj i załaduj pakiet `gridExtra`, który umożliwia m.in. ładną wizualizację danych tabelarycznych. Następnie przy pomocy dokumentacji pakietu znajdź funkcję do wizualizacji danych tabelarycznych. Użyj jej na pierwszych 10 wierszach zbioru danych `rivers`.
- (g) Stwórz wektor zawierający ciąg liczb 1000, 998, 996, . . . 850.
- (h) Stwórz wektora `a` z liczbami od 30 do 5 oraz wektora `b` z liczbami od 11 do 23. Utwórz nowy wektor `d` będący połączeniem wektora `b` i `a` (w takiej kolejności). Wyświetl go.
- (i) Stwórz wektor `nazwa` zawierający nazwy 10 lodówek. Potem stwórz wektory `pojemność_użytkowa_chłodziarki`, `pojemność_użytkowa_zamrażarki`, `cena`, `liczba_opinii` zawierające kolejno dane 10 lodówek. Następnie stwórz ramkę danych lodówki złożoną z wektorów `pojemność_użytkowa_chłodziarki`, `pojemność_użytkowa_zamrażarki`, `cena`, `liczba_opinii`. Wylicz średnią cenę lodówek. 7
- (j) Do stworzonej w poprzednim zadaniu ramki danych lodówek dodaj wpis zawierający dane nowej lodówki. Wylicz średnią cenę ponownie.

(k) Korzystając z ramki danych lodówki dodaj nową kolumnę określając ocenę klientów. Wpisz do kolumny odpowiednio oceny w skali od 0 do 5 krok 0.5. Dodana kolumna powinna się automatycznie przekonwertować do cech jakościowych (tzw. factors). Wylicz średnią ceny każdej oceny.

(l) Do ramki danych aparaty dodaj kolejne 4 lodówki. Narysuj na wykresie słupkowym liczebność reprezentantów każdej z ocen klientów.

(m) Wykorzystując ramkę danych lodówki pokaż procentowy udział każdej oceny przy pomocy wykresu kołowego oraz wachlarzowego.

(n) Do ramki danych lodówki dodaj nową kolumnę status_opinii z wartościami: "nie ma", "mniej 50 opinii", "50-100 opinii", "więcej 100 opinii" w zależności od liczby opinii. Zamień dodaną kolumnę na cechy jakościowe. Następnie przy pomocy wykresu kołowego wyrysuj procentowy udział lodówek o konkretnym statusie opinii.

(o) Wykorzystując ramkę danych lodówki stwórz zdanie o każdej z lodówek postaci: nazwa + " ma ocenę klientów " + ocena_klient'ow + " bo ma liczbę opinii" + liczba_opinii. Plus oznacza konkatenację łańcuchów i wartości.

(p) Zachować ramkę danych w pliku .csv. Załadować ramkę danych z pliku .csv Dane (10 lodówek) pobrać ze strony <http://www.euro.com.pl>

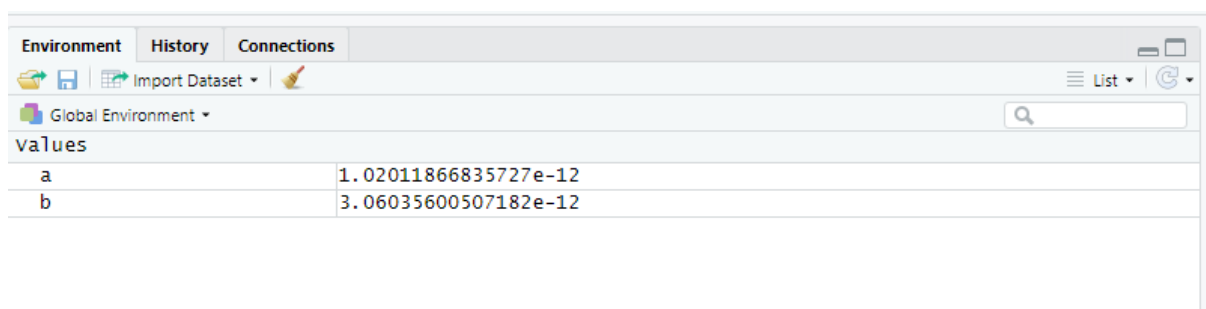
3. Wykorzystane komendy i otrzymane wyniki

#a)

```
a <- 4*sin(3.141592653589538208792)
```

```
b <- 3*a
```

```
max(a,b)
```



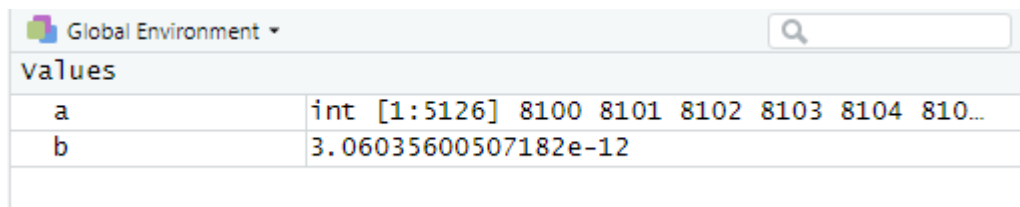
The screenshot shows the RStudio Environment pane with the following data:

Environment	
a	1.02011866835727e-12
b	3.06035600507182e-12

#c)

```
a <- c(90^2:115^2)
```

```
sum(a)
```



The screenshot shows the 'Global Environment' window in RStudio. It contains a table with two rows: 'a' and 'b'. The value for 'a' is 'int [1:5126] 8100 8101 8102 8103 8104 810...' and the value for 'b' is '3.06035600507182e-12'.

Global Environment	
Values	
a	int [1:5126] 8100 8101 8102 8103 8104 810...
b	3.06035600507182e-12

```
#e)
```

```
setwd("C:/Users/Kamil/Documents/Rstudio")
```

```
a <- "lodówka z największą pojemnością "
```

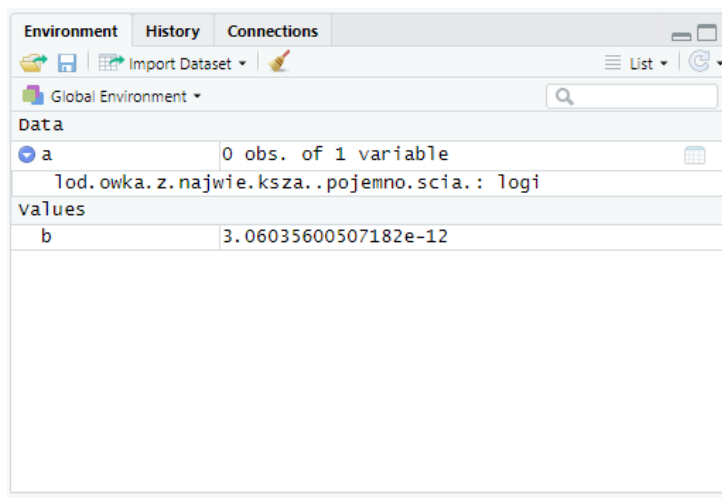
```
write(a, file='pkt_e.csv')
```

```
rm(a)
```

```
a
```

```
a <- read.csv(file = 'pkt_e.csv', sep = ',')
```

```
a
```



The screenshot shows the 'Environment' window in RStudio. It displays a data frame 'a' with 0 observations of 1 variable. The variable is named 'lod.owka.z.najwie.ksza..pojemno.scia.: logi'. The value for 'b' is '3.06035600507182e-12'.

Environment	
Data	
a	0 obs. of 1 variable
lod.owka.z.najwie.ksza..pojemno.scia.: logi	
Values	
b	3.06035600507182e-12

```
#f)
```

```
install.packages("gridExtra")
```

```
library("gridExtra")
```

```
g <- tableGrob(rivers)
```

```
grid.arrange(g)
```

```

> a <- read.csv(file = 'pkt_e.csv', sep = ',')
> a
[1] lod.owka.z.najwie.ksza..pojemno.scia.
<0 wierszy> (lub 'row.names' o zerowej długości)
> install.packages("gridExtra")
WARNING: Rtools is required to build R packages but is not currently installed. Please down
load and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/gridExtra_2.3.zip'
Content type 'application/zip' length 1109479 bytes (1.1 MB)
downloaded 1.1 MB

package 'gridExtra' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Users\Kamil\AppData\Local\Temp\RtmpAnfwqy\downloaded_packages
> library("gridExtra")
> g <- tableGrob(rivers)
> grid.arrange(g)
`

```

#g)

```
a <- seq(1000, 850, -2)
```

Data	
g	List of 282
Values	
a	num [1:76] 1000 998 996 994 992 990 988 986 ...
b	3.06035600507182e-12

#h)

```
a <- c(30:5)
```

```
b <- c(11:23)
```

```
d <- c(b, a)
```

d

```

- -
> a <- c(30:5)
> b <- c(11:23)
> d <- c(b, a)
> d
[1] 11 12 13 14 15 16 17 18 19 20 21 22
[13] 23 30 29 28 27 26 25 24 23 22 21 20
[25] 19 18 17 16 15 14 13 12 11 10 9 8
[37] 7 6 5
`

```

#i)

```



nazwy <- c('Samsung', 'Beko', 'Elektrolux',
           'Amica', 'SamsungNoFrost', 'LG',

```

```

'Bosh', 'Ravason', 'MPM ',
'Whirlpool')
poj_uzytkowa_chlodziarki <- c('240 "', '220 "', '200 "', '190 "', '210 "', '215 "', '50
"',
'185 "', '160 "', '297 "')
poj_uzytkowa_zamrazarki <- c(130, 140, 100, 150, 140, 135, 145, 100, 115,
150)
cena <- c(2399, 1939, 2499, 1799, 1399, 1995, 2199, 699, 979, 2100)
status_opinni <- c(45, 23, 1, 300, 2, 220, 64, 16, 7, 48)
lodowki <- data.frame(nazwy, poj_uzytkowa_chlodziarki,
poj_uzytkowa_zamrazarki,
, cena, status_opinni)
mean(lodowki$cena)

```

Data		
g	List of 282	
lodowki	10 obs. of 5 variables	
values		
a	int [1:26] 30 29 28 27 26 25 24 23 22 21 ...	
b	int [1:13] 11 12 13 14 15 16 17 18 19 20 ...	
cena	num [1:10] 2399 1939 2499 1799 1399 ...	
d	int [1:39] 11 12 13 14 15 16 17 18 19 20 ...	
nazwy	chr [1:10] "Samsung" "Beko" "Elektrolux" "Amica" "SamsungNoFrost" ...	
poj_uzytkowa_chlodziar...	chr [1:10] "240 \" "220 \" "200 \" "190 \" "210 \" "215 \" "...	
poj_uzytkowa_zamrazarki	num [1:10] 130 140 100 150 140 135 145 100 115 150	
status_opinni	num [1:10] 45 23 1 300 2 220 64 16 7 48	

#j)

```
newRow <- data.frame(nazwy = 'INdesit', poj_uzytkowa_chlodziarki = '259 "',  
  poj_uzytkowa_zamrazarki = 150,  
  ,cena = 2899, status_opinni = 74)
```

```
lodowki <- rbind(lodowki, newRow)
```

```
mean(lodowki$cena)
```

Data	
g	List of 282
lodowki	10 obs. of 5 variables
newRow	1 obs. of 5 variables
values	

#k)

```
lodowki$ocena <- c('5', '5', '3', '5', '4', '5', '5', '5', '3', '2', '5')
```

```
aggregate(lodowki$cena, list(lodowki$ocena), mean)
```

#l)

```
newRow <- data.frame(nazwy = 'INdesit', poj_uzytkowa_chlodziarki = '259 "',  
  poj_uzytkowa_zamrazarki = 150,  
  ,cena = 2899, status_opinni = 74, ocena = 5)
```

```
lodowki <- rbind(lodowki, newRow)
```

```
newRow <- data.frame(nazwy = 'Bosh Frosty', poj_uzytkowa_chlodziarki =  
  '230 "', poj_uzytkowa_zamrazarki = 155,  
  ,cena = 1899, status_opinni = 78, ocena = 5)
```

```
lodowki <- rbind(lodowki, newRow)
```

```
newRow <- data.frame(nazwy = 'INdesit Mini', poj_uzytkowa_chlodziarki =  
  '129 "', poj_uzytkowa_zamrazarki = 50,  
  ,cena = 670, status_opinni = 64, ocena = 3)
```

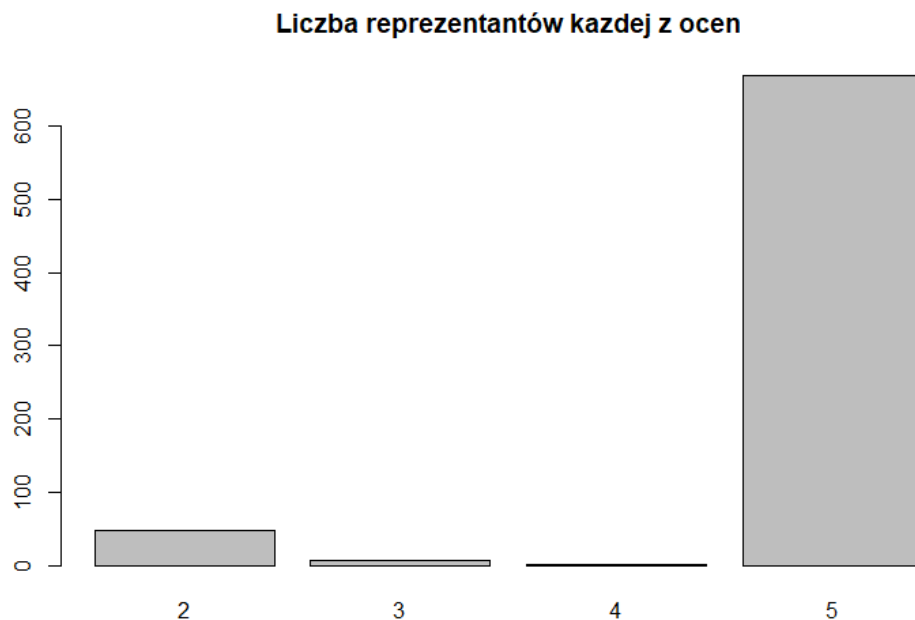
```
lodowki <- rbind(lodowki, newRow)
```

```
newRow <- data.frame(nazwy = 'Samsung Nofrost Max',
  poj_uzytkowa_chlodziarki = '290 "', poj_uzytkowa_zamrazarki = 250,
  ,cena = 3899, status_opinni = 51, ocena = 5)

lodowki <- rbind(lodowki, newRow)
```

```
dane <- aggregate(lodowki$status_opinni, list(lodowki$ocena), sum)

barplot(dane[,2], names.arg = dane[,1], main = 'Liczba reprezentantów kazdej z
ocen')
```



```
#m)

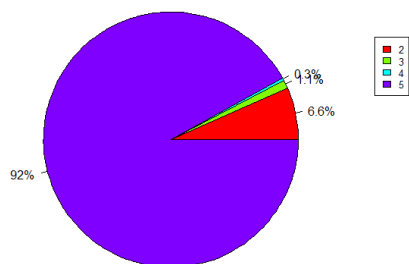
#wykres kolowy

labels <- round(dane[,2]/sum(dane[,2]) * 100, 1)

labels <- paste(labels, "% ", sep="")

pie(dane[,2], radius = 1, col = rainbow(length(dane[,2])),
  labels = labels)

legend(1.6, 0.8, dane[,1], cex=0.8, fill=rainbow(length(dane[,2])))
```

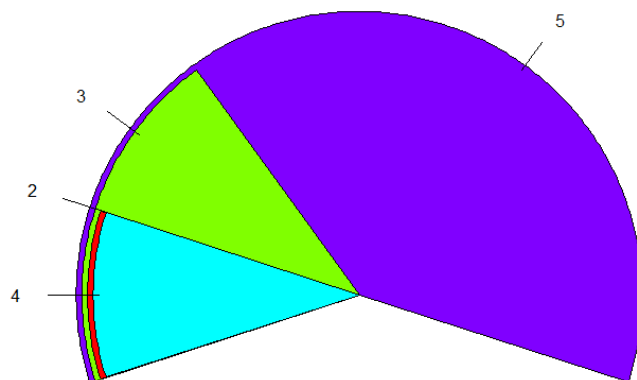
#wykres wachlarzowy

```
install.packages("plotrix")
```

```
library(plotrix)
```

```
percentage <- table(lodowki$ocena) / length(lodowki$ocena)
```

```
fan.plot(percentage, labels = names(percentage), main = "Procentowy udział  
oceny")
```



#n)

```
new_column <- ifelse(lodowki$status_opinni>100,'więcej 100 opinii',
```

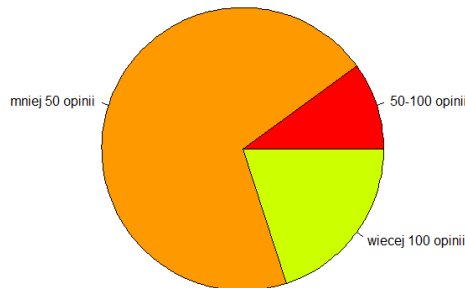
```
  ifelse(lodowki$status_opinni>=50, '50-100 opinii',
```

```
    ifelse(lodowki$status_opinni>0, 'mniej 50 opinii', 'nie ma')))
```

```
lodowki['status_opinii'] <- factor(new_column)
```

#wykres kolowy

```
pie(table(lodowki$status_opinii), radius = 1, col =  
rainbow(length(lodowki$status_opinii)))
```



#o)

```
for (i in 1:length(lodowki$nazwy)){  
  print(paste(lodowki$nazwy[i], 'posiada ocene klientów', lodowki$cena[i], 'bo  
posiada liczbe opinii',  
              lodowki$status_opinii[i]))  
}
```

Global Environment	
y	List of 282
lodowki	10 obs. of 7 variables
newRow	1 obs. of 6 variables
values	
a	int [1:26] 30 29 28 27 26 25 24 23 22 21 ...
b	int [1:13] 11 12 13 14 15 16 17 18 19 20 ...
cena	num [1:10] 2399 1939 2499 1799 1399 ...
d	int [1:39] 11 12 13 14 15 16 17 18 19 20 ...
labels	chr [1:4] "6.6%" "1.1%" "0.3%" "92%"
nazwy	chr [1:10] "Samsung" "Beko" "Elektrolux" "Amica" "SamsungNoFrost" ...
new_column	chr [1:10] "mniej 50 opinii" "mniej 50 opinii" "mniej 50 opinii" ...
percentage	'table' num [1:4(1d)] 0.1 0.2 0.1 0.6
poj_uzytkowa_chlodziar...	chr [1:10] "240 \\" "220 \\" "200 \\" "190 \\" "210 \\" "215 \\" ...
poj_uzytkowa_zamrazarki	num [1:10] 130 140 100 150 140 135 145 100 115 150
status_opinii	num [1:10] 45 23 1 300 2 220 64 16 7 48

#p)

```
write.csv(lodowki, 'lodowki.csv')
```

```
dane <- read.csv('lodowki.csv')
```

4. Wynik działania:

Kod programu dostępny w repozytorium:

<https://github.com/arturkoziel/APU>