Introduction to R **University of Bremen**
FB03 Mathematics and Computer Science Eike Voß, evoss@uni-bremen.de

Exercise 02

Programming tasks should be written in a `.R`-script. Please include your lastname(s) in the file name (e.g. `Ex02_Voss.R`) and submit it via **DoIT!** in stud.ip, where you also find the submission deadline.
To receive sample solutions for the Presence Tasks throughout the course, you need to submit a solution in **DoIT!**.

Presence task 1:

a) The function `mean()` computes the mean value of a vector. Write **your own** function `mymean()` to compute the mean value of all components of a vector. You are not allowed to be cheeky and do this:

```
mymean <- function(x){mean(x)}
```

Hint: Use the functions: `sum()` and `length()`.

b) Write **your own** function `rootn()` to compute the n-th root of a number x (we assume $\mathbb{R} \ni x > 0$).

Presence task 2:

a) Store all natural numbers from 1 to 50 in the variable `x` and a sequence from 0 to 10 with step size 0.2 in the variable `y`. Calculate $\sqrt[3]{x}$ and $\log(y)$.

Multiply out x and y and explain the warning message that appears.

b) Generate the following vectors with R:

$$\underbrace{1 \quad 2 \quad 3}_{40\,\text{times}}$$

$$1 \quad 1 \quad 2 \quad 2 \quad 2 \quad \ldots \quad \underbrace{100 \ldots \quad 100}_{101\,\text{times}}$$

$$\underbrace{1 \quad 1 \quad 2 \quad 2 \quad 2 \quad \ldots \quad \underbrace{100 \ldots \quad 100}_{101\,\text{times}}}_{4\,\text{times}}$$

Note: For the second example you can use `rep()` with the parameter `times`.

c) Generate a matrix `M` with 20 columns and the following row entries:

```
Row 1: numbers between 0.5 and 10 with spacing 0.5
row 2: column 1-10 with entry 3, column 11-20 with entry 0
Line 3: 1, 2, 1, 2, 1, 2, 1, 2, ...
Line 4: 1, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6
Line 5: 1, 2, 1, 2, 3, 1, 2, 3, 4, 1, 2, 3, 4, 5, 0, 0, 0, 0, 0,0
Line 6: 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 5, 6, 7, 8
Line 7: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1
```

Do not use `for`-loops. Create vectors **not** in a way that is not explicitly, e.g. **do not do**
`line4 <- c(1, 2, 2, 3, 3, 4, 4, 4, 5, 5, 5, 5, 6, 6, 6, 6)`, but try to use functions like `rep`, `seq` or `a:b`.

You can generate the lines one by one, e.g. `rowX <- c(rep(2:6,times=1:5),5:1)`, and then combine them into a matrix using `M <- rbind(row1, row2, row3,...)`.

Homework task 1 (2 Points):

Define the following vectors:

$$\underbrace{\texttt{"T", "C"}}_{100\text{times}}$$

$$\underbrace{\texttt{"T", .., "T"}}_{100\text{times}},\underbrace{\texttt{"C", .., "C"}}_{101\text{times}}$$

Homework task 2 (4 Points)

Define the matrix

$$X = \begin{pmatrix} 31 & 21 & 12 \\ 22 & 41 & 30 \\ 19 & 64 & 52 \end{pmatrix} \quad \text{and the vector} \quad v = \begin{pmatrix} 12 \\ 2 \\ -10 \end{pmatrix}$$

and compute $\det(X)$, X^{-1}, $X^T X^{-1}$, XX^{-1} and $X^{-1}v$. Is the result of XX^{-1} to be expected?

Hints:

- Use `?matrix`.
- An example matrix with 3 columns is created like this: `matrix(1:6, ncol = 3)`.
- Other commands that are needed: `solve(A)`, `t(A)` (use `?solve` etc. if you don't understand these functions).
- Be careful about the multiplication operator you are using!

Homework task 3 ($7 \cdot 2$ Points):

The command `x <- rnorm(1000, mean=170, sd=10)` generates 1000 realizations of a normally distributed random variable $X \sim \mathrm{N}(\mu, \sigma^2)$ with parameters $\mu = 170$ and $\sigma = 10$. Before running this command, please type `set.seed(453)` at the very beginning of the task (and nowhere else) to set a random seed for reproducibility reasons.

a) Calculate the mean and standard deviation of `x` and let the result be shown in the console. How big is the relative difference of the true value of μ and σ from the calculated values of the mean and standard deviation?

b) Store the first 500 values of `x` in the vector `y` and the last 500 values of `x` in `z`.

c) Store the entries of `x` which `do not` satisfy the conditions $150 \leq x \leq 190$ in `w`.

d) Generate 1000 realizations from a normal distribution with parameters $\mu = 180$ and $\sigma = 10$ and store these in the vector `x1`.

e) Summarize the data `x` and `x1` into 2 columns of a matrix `M` using `cbind()`.

f) Let each row of the matrix `M` define the heights (in cm) of a couple. Determine for how many couples both partners are larger than 190 cm.

g) In how many couples is at least one partner smaller than 150 cm?

More hints:

- `x[x>=150]` returns components of x, where $x \geq 150$.
- `?tail` might help you with one of the tasks.
- In tasks f) and g) you can still work with vectors `x` and `x1`. It is allowed but not necessary to use `M` itself.
- `&` is the component-wise AND, | the component-wise OR. Example: `(x<190)&(x>185)`
- You can do calculations with `TRUE` and `FALSE`; `TRUE` corresponds to 1, `FALSE` to 0. Something like `sum((x < 190) * (x > 185))` actually works.