Introduction to Computers
20.10.2003–24.10.2003

Vita Rutka
e-mail: rutka@mathematik.uni-kl.de
internet: http://www.mathematik.uni-kl.de/~rutka

Exercises for the MATLAB part

If you have a nice solution to some problem, don't hide it! It can be included in the solutions part as an example (solutions will be available at the end of the course). Of course, with your name mentioned on it!

1 Preparation

Exercise 1
Draw a flow chart for the following actions:

1. Mr. X. calls Mr. Y every five minutes till he picks up the phone.
2. Mr. X. says "Hello!" to his friends and colleges, but to his boss he says "Good morning!" in the morning and "Good afternoon!" in other time of the day. To other people he says nothing.
3. If the time is 7.00 a.m., alarm rings and that makes John to wake up. If he is sleepy, he sets alarm to 8.00 a.m. and continues sleeping. Otherwise he gets up, takes a shower and drinks his coffee. Then he reads newspapers and at 8.15 goes to his job. If he was sleepy at 7.00: after alarm at 8.00 he quickly gets up and at 8.15 goes to his job.

2 Basics

2.1 Online-mode

To get started, you can start *MATLAB* and type a command "demo". Just follow the steps.

Another useful commands are "helpdesk" (opens help information) and "lookfor". E.g., if you are looking for some command to invert a matrix, type


```
>> lookfor 'matrix inverse'
```

After that, you can get more details about functions by the "help" command, e.g.,

```
>> help inv
```

Exercise 2
Enter the vector $x = [1\ 2\ 3\ 4]$. Compute $\sin(x)$, $2 * x$ and $2 + x$.

Exercise 3
Enter two vectors: $x = [4\ 5\ 6]$ and $y = [12\ 13\ 14]$. Compute $x+y$, $x' * y$, $x * y'$ and $x. * y$. Do you guess how the *MATLAB* commands "acts" on variables?

Exercise 4
Make two variables: $x1 = 4$ and $x2 =' 4'$. Try to find $2 * x1$ and $2 * x2$. What is the difference?

Exercise 5
Enter the matrix

$$A = \begin{pmatrix} 2 & 3 & 4 \\ 0 & 2 & 3 \\ 6 & 8 & 13 \end{pmatrix} .$$

Find $\sin(A)$, $\exp(A)$ and $\mathrm{expm}(A)$.

Exercise 6
Plot the function

$$\frac{\sin x}{x}$$

in the interval $x \in [-10\,,\,10]$.

2.2 Script-files

Exercise 7
Write "classically" the first program in any programming language: print on the screen "Hello, world!". For that you can write the following script-file:

```
% Script file helloworld.m
% Prints ``Hello, world!''
% Last changes: 19.10.2001 by Vita

clc; % clear the MATLAB window
disp(' ') % leave one empty line
disp(' Hello, world! ')
```

Save this file under the name helloworld.m, and then run into *MATLAB* window by typing [1]

```
>>helloworld
```

Exercise 8
Plot function

$$x \sin x \ .$$

The interval and number of grid points should be given by user.

Check commands *linspace*, *plot* and *input*.

Exercise 9
Plot the following functions given by a parameter $t \in (0, 1)$ (all of them in one figure!):
$1) x = 0, y = 2t$, $2) x = 2t, y = 1$, $3) x = 2, y = 2t$, $4) x = t + 2.5, y = 2 - 2|2t - 1|$, $5) x = 3 + t, y = 1$, $6) x = 5, y = 2t$, $7) x = 5 + t, y = 0$, $8) x = 6.5, y = 2t$, $9) x = 6.5 + t, y = 0$, $10) x = 9 + \cos(2\pi t), y = 1 + \sin(2\pi t)$.
(Check commands *zeros, ones, plot, hold on.*)

Exercise 10
Write a code to compute $n!$, where n is a natural number given by user.

(Check *for* loop construction.)

Exercise 11
Write a code which generates 1000 normally distributed random numbers and then draws the histogram.

[1] It is supposed to be done like this. Unfortunately, nothing is perfect in this world, and also *MATLAB* as well as our computer system not. Sometimes you can get in problems, if you try to run a new created script file. You can get error message "undefined function or variable 'helloworld' ". In such cases, you can use command *clear function* or . . . exit *MATLAB* and re-run it.

(Check commands *randn* and *hist*.)

Exercise 12
In *MATLAB*, it is very important to "think in vectors". The following script file might be good for illustration. . .

```
% script file vectorize.m
% show the importance of "vectorial thinking" in MATLAB
% example for demonstrating:
% have 3D array and find square of each component
% (pointwise multiplication)
% last changes: 19.10.2001

clear;
clc;

nx = 100;
ny = 100;
nz = 100;

M = rand(nx,ny,nz); % with this matrix we'll work

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% The optimal version

disp('Pointwise multiplication the right way')
tic;    % set the timer
M1=M.*M;
t1=toc; % registrate the time

disp(sprintf('time needed: %0.5g',t1))
clear M1 % we do not need it anymore...

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% loop, but not that bad
```

```
disp(' ')
disp('Pointwise multiplication in a bad way,')
disp('memory for the result preallocated')
tic
M2 = zeros(nx,ny,nz); % preallocate memory
for i = 1:nx
  for j = 1:ny
    for k=1:nz
        M2(i,j,k) = M(i,j,k) * M(i,j,k);
    end
  end
end
t2=toc;
disp('Speedup:')
disp(sprintf('Time for the correct way: %0.5g',t1))
disp(sprintf('Time for the loop with memory allocation: %0.5g',t2))
disp(sprintf('Ratio with the correct way: %0.5g',t2/t1))

clear M2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Bad written loop - without reserving the memory

disp(' ')
disp('Pointwise multiplication the worst way,')
disp('memory for the result not preallocated.')
tic
for i = 1:nx
  for j = 1:ny
    for k=1:nz
        M3(i,j,k) = M(i,j,k) * M(i,j,k);
    end
  end
end
t3=toc;
disp('Speedup:')
disp(sprintf('Time for the correct way: %0.5g',t1))
```



```
disp(sprintf('Time for the loop with memory reservation: %0.5g',t2))
disp(sprintf('Time for the loop without memory reservation: %0.5g',t3))
disp(sprintf('Ratio: %0.5g',t3/t1))
disp(sprintf('Ratio: %0.5g',t3/t2))
```

Exercise 13
Finite precision arithmetics.
Often roundoff errors due to finite precision arithmetics play important role. Check the following examples:

1. $10000 - \sum_{k=1}^{100000} 0.1 = 0$
2. $10000 - \sum_{k=1}^{80000} 0.125 = 0$
3. $f(x) = g(x)$ for $f(x) = x\left(\sqrt{x+1} - \sqrt{x}\right)$ and $g(x) = \frac{x}{\sqrt{x+1}-\sqrt{x}}$. In particular, $f(500) = g(500)$.

Check, if this holds also numerically!

Exercise 14
It is known that

$$\sum_{k=1}^{n} k = \frac{n(n+1)}{2} ,$$
$$\sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6} ,$$
$$\sum_{k=1}^{n} k^3 = \frac{n^2(n+1)^2}{4} .$$

Compute these sums and compare the result using the formulae given above. n is given by the user.

Exercise 15
Plot the function

$$f(x) = \begin{cases} 0 & , \quad x \leq -1 \\ x+1 & , \quad -1 < x \leq 0 \\ -x+1 & , \quad 0 < x \leq 1 \\ 0 & , \quad \text{else} \end{cases}$$

in the interval $[-2\,,\,2]$.

(Check the *if-then-else* construction. If you want to implement this in an optimal way, check command *find*).

Exercise 16
Write a code which asks for a grade and gives answer

very good	for 1.0 and 1.3
good	for 1.7, 2.0 and 2.3
acceptable	for 2.7, 3.0 and 3.3
passed	for 3.7 and 4.0
failed	for 4.3, 4.7 and 5.0
there exists no such grade	in other cases

Update your code in such way that after giving answer user has a possibility to repeat his query.

(Check operator *switch*).

Exercise 17
The polynomials

$$T_k(x) = \cos(k \arccos(x))$$

are called *Chebyshev polynomials of the first kind*. Write a code which asks user to enter some number n and plots the first n Chebyshev polynomials in the interval $(-1, 1)$.

Exercise 18
Bezier Curves.
Let $P_0, P_1, \ldots, P_n$ be the control points in $\mathbf{R}^2$. The corresponding *Bezier*

curve is defined by

$$C(t) = \sum_{i=0}^{n} P_i B_{i,n}(t) \quad , \quad t \in [0,1]\,,$$

where $B_{i,n}(t)$ is the so called *Bernstein polynomial*

$$B_{i,n}(t) := \binom{n}{i} t^i (1-t)^(n-i)$$

and the binomial coefficient is defined by

$$\binom{n}{i} = \frac{n!}{(n-k)!k!}\,.$$

Write a program which

1. asks user to enter n
2. reads the coordinates of the points P_k from the mouse clicks of the user
3. calculates and plots the corresponding Bezier curve

(Check the command *ginput*.)

Exercise 19
One possibility to perceive Dirac-δ function, is to take a limit when $n \longrightarrow \infty$ for the following sequence of functions:

$$f_n(x) = \begin{cases} n & , \quad x \in \left[\frac{-1}{2n}\,,\,\frac{1}{2n}\right] \\ 0 & , \quad \text{else}\,. \end{cases}$$

Write a code which displays functions f_n , and after plotting of f_n asks whether to plot f_{n+1}.

Exercise 20
Consider the following potential series:

$$\sin(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i+1}}{(2i+1)!} \quad , \quad |x| < \infty$$

$$\cos(x) = \sum_{i=0}^{\infty} (-1)^i \frac{x^{2i}}{(2i)!} \quad , \quad |x| < \infty$$

$$(1-x)^{-3} = 1 + \frac{1}{2}\sum_{i=1}^{\infty} (i+1)(i+2)x^i \quad , \quad |x| < 1$$

$$\ln(1+x) = \sum_{i=1}^{\infty} (-1)^i \frac{x^i}{i} \quad , \quad |x| < 1 \, .$$

Select some n and replace the infinite series by finite sums, i.e., take only first n terms into account, thus obtaining some approximations of functions above. Plot functions themselves and their approximations in the interval $x \in (0, 0.5)$.

Exercise 21
It is known that for $x \in (0, \pi)$ the following identity holds:

$$\sum_{n=1}^{\infty} \frac{n^2+1}{n^4+1} \cos(nx) = \frac{(\pi - x)^2}{4} - \frac{\pi^2}{12} + \sum_{n=1}^{\infty} \frac{n^2-1}{n^2(n^4+1)} \cos(nx) \, .$$

Replace the infinite series by a finite sum, up to $n = N$ and check how good the identity is satisfied. (Such transformations of infinite series can be used to improve the convergence speed, they are called Kummer's transformations.)

Exercise 22
With some periodicity conditions on function $f(t)$ it can be written as Fourier series

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} (a_n \cos n\varphi t + b_n \sin n\varphi t) \, , \quad \varphi = \frac{2\pi}{T} \, , \tag{1}$$

where T is the period.

Problem which appears, if we want to use Fourier series in numerical approximations, is that only finite number of terms can be used; i.e., we replace (1) with finite series

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{N} (a_n \cos n\varphi t + b_n \sin n\varphi t) \quad , \tag{2}$$

where N is some finite number. Plot the following functions and their truncated Fourier series with $N = 5$ in the interval $[-T, T]$. Parameter L (half of the period, $T = 2L$) should be given by user. Functions are given only in one interval of the periodicity.

1.
$$f(t) = \frac{1}{L}(L - t), \ t \in [0, T]$$
$$a_n = 0 \ , \ b_n = \frac{2}{\pi n} \, .$$

2.
$$f(t) = \begin{cases} t(L-t) & , \ t \in [0, L] \\ -t(-L-t) & , \ t \in [-L, 0] \end{cases}$$
$$a_n = 0$$
$$b_n = \begin{cases} 0 & , \ n \text{ even} \\ \frac{8L^2}{\pi^3 n^3} & , \ n \text{ odd} \end{cases} \, .$$

3.
$$f(t) = \cos\frac{\pi t}{L}, \ t \in [0, L]$$
$$a_0 = 0$$
$$b_n = \begin{cases} 0 & , \ n \text{ odd} \\ \frac{4n}{\pi(n^2-1)} & , \ n \text{ even} \end{cases} \, .$$

Update your code such that after plotting truncated series with $N = 5$ user can choose to increase the precision by taking $N = 10 \, , \, 15 \, , \, \ldots$

Exercise 23
Plot the following families of parametrised curves with parameter $\varphi \in [0 \, , \, 2\pi]$:

1. Evolute of ellipse:
$$\begin{cases} x &= A \cos^3 \varphi \\ y &= B \sin^3 \varphi \, . \end{cases}$$
Let $A = 2$ and B change from 0.5 to 1.5 with step 0.1

2. Epicycloid:

$$\left\{ \begin{array}{lcl} x & = & (a+b)\cos\varphi - b\,\cos\left(\frac{a+b}{b}\varphi\right) \\ y & = & (a+b)\sin\varphi - b\,\sin\left(\frac{a+b}{b}\varphi\right) \end{array} \right. .$$

Let both–a and b to change from 0.8 to 1.2 with step 0.1.

3. Folium of Descartes:

$$\left\{ \begin{array}{lcl} x & = & 3a\,\varphi/\left(1+\varphi^3\right) \\ y & = & 3a\,\varphi^2/\left(1+\varphi^3\right) \end{array} \right. .$$

Let a change from 0 to 10 with step size 2.

4. Deltoid:

$$\left\{ \begin{array}{lcl} x & = & 2a\cos\varphi + a\cos 2\varphi \\ y & = & 2a\sin\varphi - a\sin 2\varphi \end{array} \right. .$$

Let a change from 10 to 15 with step size 0.5.

5. Nephroid:

$$\left\{ \begin{array}{lcl} x & = & \frac{1}{2}a\left(3\cos\varphi - \cos 3\varphi\right) \\ y & = & \frac{1}{2}a\left(3\sin\varphi - \sin 3\varphi\right) \end{array} \right. .$$

Let a change from 0.1 to 0.8 with step size 0.1.

Exercise 24

Write a *function* which for a given input x (scalar or vector) returns the value of

$$y = (1-x)\exp(\sin(x))\,.$$

Exercise 25
Bisection search.

One of the simplest algorithms for solving equation

$$f(x) = 0 \qquad (3)$$

is the so called *bisection search* method. At first, an interval has to be initialised where the root could be. Let us call it interval (a,b). Then, we do:

- If $f((a+b)/2) = 0$, we have found the root $\Rightarrow$ Exit

- If $f(a)$ and $f((a+b)/2)$ both have the same sign, it is clear that the root is somewhere between $(a+b)/2$ and b. So, we reassign $a = (a+b)/2$.
- If $f(a)$ and $f((a+b)/2)$ have different signs, the root is somewhere between a and $(a+b)/2$. So, we reassign $b = (a+b)/2$.
- Continue until $b-a$ is smaller as some prescribed tolerance. When tolerance is reached, set $(a+b)/2$ as approximated solution.

Implement the bisection search and solve numerically the equation

$$(1-x)\exp(\sin(x)) = 0\,.$$

As initial interval take $(-10, 10)$.

Hint: check the commands *while, break, if-then-else* and usage of functions.

Exercise 26
Fixed point iteration.

Another method for solving equation (3) is so called *fix point iteration.* At first, (3) is rewritten in an equivalent form

$$\varphi(x) = x$$

and then the iteration is done as

$$x_{k+1} = \varphi(x)\,,$$

where the initial guess x_0 has to be given. Iteration is stopped when the method stagnates, i.e., $|x_{k+1} - x_k| < tol$ with some given tolerance tol or maximal number of iterations has been reached.

Implement the fix point iteration and solve with it numerically the equation

$$1 - \sqrt{x} - x = 0\,.$$

Exercise 27
Fractals.

Let z be a complex number. Consider iteration

$$z_{n+1} = z_n^2 + C$$

with some complex number C. Depending on C and z_0, the sequence can be divergent.

The fractal pictures can be obtained in the following way:

1. Select the initial values to be in the domain $[-1,1] \times [-1,1]i$. Let z_0 to be the mesh points $z_0^{k,p} = x_k + y_p i$, $k = 1, 2, \ldots, n$ and $p = 1, 2, \ldots, n$. Let $R(z^{k,p}) := R(k,p)$ be the "divergence speed function".
2. Select some maximal radius, say, $R_{max} = 5$ where you say that if $|z_n| > R_{max}$m then the sequence has "diverged". If the divergence criterion is satisfied at the iteration N, set $R(k,p) = N$.
3. Finally, when you have gone through all points, plot the function R.

(Check the functions *while*, *if-then-else* and *contourf*. Compare the output of *contourf* with the output of *surf*.)

3 "Real" problems

Exercise 28
Write a function $bank(S, I, N)$ which returns how much money you have on your account after N months, if the yearly interest rate is I (calculated each month) and each month you add amount S to your account.

Exercise 29
Traffic policeman has observed the following behaviour of car drivers:

experience	*average speed*
1 month	51 km/h
2 months	60 km/h
2.5 months	64 km/h
3 months	70 km/h

Let x be experience; but y–the average speed. Plot the policeman's observations, and try to fit a straight line: least squares approximation, which is the slope

$$y = a + bx \ ,$$

where

$$b = \frac{\sum_{k=1}^{n} (y_k - \bar{y}_n)(x_k - \bar{x}_n)}{\sum_{k=1}^{n} (x_k - \bar{x}_n)^2}$$

and

$$a = \bar{y}_n - b\,\bar{x}_n \ .$$

n is number of measurements, and $\bar{y}_n = \frac{1}{n}\sum_{k=1}^{n} y_k$; $\bar{x}_n = \frac{1}{n}\sum_{k=1}^{n} x_k$.

Continue this line till 24 months. What happens, if you add one more measurement: that the average speed for car drivers with 2 years experience is 120 km/h?

Exercise 30
Create a random vector x of length 100. (E.g., energy of the incoming signal). If $x(i) \geq 0.5$, $y(i) = 1$ (decide there really is a signal). Else $y(i) = 0$ (decide it was just a noise). Plot x and y.

Exercise 31
Radioactive decay of material S with mass m(kg) can be described by ODE

$$\dot{m} = -\lambda m \ , \tag{4}$$

where λ is the decay rate, related with half-life period $T_{1/2}$ as follows:

$$T_{1/2} = \frac{\ln 2}{\lambda}. \tag{5}$$

Assume that at the time $t = 0$ there is m_0 kg of material. The simplest approximation of the ODE (4) in this case is

$$\frac{m(t+h) - m(t)}{h} = -\lambda m(t) \ , \tag{6}$$

leading to

$$m(t+h) = m(t)(1 - \lambda h) \ . \tag{7}$$

(Explicit Euler's method). Consider that initially there are 10 kg of ${}^{238}_{92}U$ with half-life period $T_{1/2} = 23.5$ min. Plot changes in amount of it during 5 min, taking the time step $h = 0.05$ min. Compare the result with the exact solution $m = \exp(-\lambda t)m_0$. What happens if you take $h = 0.01$, $h = 0.1$, $h = 0.15$?

Exercise 32
Population dynamics in NN-town can be described by ODE

$$\frac{dp}{dt} = \frac{1}{25}\,p - \frac{1}{25\cdot 10^6}\,p^2\,.$$

Using similar discretization as before, we obtain

$$p(t+h) = h\left(\frac{1}{25}\,p(t) - \frac{1}{25\cdot 10^6}\,p(t)^2\right) + p(t)\,.$$

In 1970 there were around 8000000 inhabitants in NN-town. Using 200 steps, find how many people are living in NN-town in 2003. Plot the changes in the amount of people. Which result you get, if use 50 time steps? 500 steps? What would happen, if initially there would be around 1000000, 900000 or 11000000 inhabitants?

Exercise 33
This is a famous example leading to Fibbonaci numbers. Initially we have one rabbit pair. The rabbits (counted in pairs) propagate twice and then die. If we denote with x_i i years old couples, $i \in \{1,2\}$, population can be described by following difference equation:

$$\left\{\begin{array}{lcl} \vec{x}(t+1) & = & A\,\vec{x}(t) \\ \vec{x}(0) & = & (1\,,\,0)^T\,, \end{array}\right.$$

where

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}\,.$$

Number of one year old couples in time t then is t-th Fibbonaci number. Compute the first 50 Fibbonaci numbers and display them.

Exercise 34
Phase flow.

Suppose, the flow of the liquid is given by some vector function $\vec{v} = (v_1, v_2)$. Then, velocity of a particle at some position $\vec{x}$ is given by

$$\frac{\partial \vec{x}}{\partial t} = \vec{v}(\vec{x})\,. \qquad (8)$$

We want to find how this particle moves in the liquid. From (8) we get

$$\vec{x}(t+\tau) \approx \vec{x}(t) + \tau\vec{v}(\vec{x}(t))$$

with some small time step τ.

Consider the following velocity fields:

1. $\vec{v} = (x, 5y)$
2. $\vec{v} = (x, -5y)$
3. $\vec{v} = (-x, -5y)$
4. $\vec{v} = (x, x + 0.5y)$
5. $\vec{v} = (x, x - 0.5y)$

Your task:

- Visualise the velocity fields in domain $[-1,1]^2$. (Check the commands *ndgrid* and *quiver*).
- Using approximation above calculate and visualise trajectory of
  - One particle
  - Several particles with different initial positions
  - Several particles initially located along a circle with centre $(0,0)$ and radius 0.5.

Exercise 35
Security problem.

In the WEB page of this course
`http://www.mathematik.uni-kl.de/~rutka/CCOURSE/ccourse.html`
there is a file `code_key.dat`. It contains a very primitive code, where to each letter in the first column (with capital letters) corresponds some other letter in the second column. Punctuation signs remain unchanged. E.g., if the code reads

$$\begin{array}{cc} A & e \\ \vdots & \vdots \\ H & p \\ \vdots & \vdots \\ L & w \\ \vdots & \vdots \\ O & a \\ \vdots & \vdots \end{array}$$

then the string "Hallo!" in coded version would look like "pewwa!". Decoded back with this code, we would get "HALLO!".

Write a program which codes and decodes the message of your choice (download the file `code_key.dat` and assure that your program each time reads the code from file). Check, that you get back the same what you had at the beginning. When you are done with it, try to convert the file `message.dat`, available in the WEB page.

(Check commands *fopen* and *fclose*, as well as *fgetl*, *isstr*, *upper* and *lower*.)