T. Kärkkäinen
Course: Application Programming in MATLAB Environment 2002

Exercises 1 & 2

First we try to make a smooth landing on the usage of MATLAB. The main emphasis is on exploring the built-in demos, *help*-commands and other basic commands of the MATLAB command interface. Second part deals with linear algebraic testing, definition of matrices and computation of eigenvalues and eigenvectors, and illustrations around these issues.

> EH: *It only takes around half an hour for fastest students to go through the first part of this exercise?!*
> TK: *Probably true, but the aim is to learn and also to be able to use these issues in what follows!*

PART I

Problem 1

Open MATLAB. Type commands `help demo` and `help intro`. Go through the built-in introduction to MATLAB by typing the command `intro`.

Problem 2

MATLAB contains many commands that are useful for exploring the necessary commands and examples for realizing your own macros and functions. Such commands are, e.g.,
`help, helpwin, doc, helpdesk, lookfor`.
Use these commands themselves for exploring their usage.
Get also acquainted with `MatLab Topics` and `Search MatLab Index` using `helpdesk`.
Seek (and learn) the way on how to divide a MATLAB-command over multiple lines in a macro file.

Problem 3

When working on the command interface usable commands are, e.g.,
`diary, who(s), clear, close, dir, type, delete, cd, !`
To read and store (numeric) information one can use, e.g., the following commands
`load, save, fscanf, fprintf, dlmread, dlmwrite`
Learn how to use these commands by using your own favorite *help*-possibility (btw, my own choice would be `helpdesk`).

Problem 4

Have a look of the m-files that were shown during the first lecture. Explore the commands that have been utilized in them.

end of PART I

PART II

Problem 5

Create as random as possible test macro for comparing the built-in routine `repmat` in MATLAB and the other given variant `repmat_own.m`.
What features and/or bugs do you find according to this comparison?
Is there some MATLAB command for exploring the actual realization of `repmat`?

Define the following matrices:

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}, \quad \mathbf{A}_2 = \begin{bmatrix} 2 & 1 \\ 1 & 3 \end{bmatrix}, \quad \mathbf{A}_3 = \begin{bmatrix} 2 & \sqrt{6} \\ \sqrt{6} & 3 \end{bmatrix}, \quad \mathbf{A}_4 = \begin{bmatrix} 1 & 2 \\ 2 & -2 \end{bmatrix}, \quad \mathbf{A}_5 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}.$$

Problem 6

Compute the eigenvalues and eigenvectors of these matrices. As stated in the lecture notes, the matrix identity $\mathbf{A}_i\mathbf{U}_i = \mathbf{U}_i\mathbf{D}_i,\ i = 1,\ldots,5$, is always valid between the original matrix and its eigenvectors and eigenvalues in the matrix form. Check if the diagonalization formula $\mathbf{U}^T\mathbf{A}\mathbf{U} = \mathbf{D}$ is also valid? If not, what goes wrong?

Draw in $\mathbf{R}^2$ the natural basis vectors $(\mathbf{e}_1^T,\, \mathbf{e}_2^T) = \{(1,0),\ (0,1)\}$ and the corresponding transformed vectors $\mathbf{U}_i^T\mathbf{e}_1$ and $\mathbf{U}_i^T\mathbf{e}_2$ using the `quiver`-command.

Problem 7

Create an equidistant 2D-mesh in $[-4,4] \times [-4,4]$ (`meshgrid`, cf. `main11.m`). Using `mesh` and `contour` commands illustrate the quadratic forms $\mathbf{x}^T\mathbf{A}_i\mathbf{x},\ i = 1,\ldots,5$, in the given mesh. Because `for`-loops are slow in MATLAB, try also to find a way to compute the values of the quadratic forms in mesh points *without* using them (you probably need to utilize `size` and `reshape` commands). Add to the contour plots of the quadratic forms also the corresponding vectors $\lambda_1^{-\frac{1}{2}}\mathbf{u}_1$ and $\lambda_2^{-\frac{1}{2}}\mathbf{u}_2$. What do you notice?
Finally, realize the same illustrations for the square-root matrices $\sqrt{\mathbf{A}_i},\ i = 1, 2$.

end of PART II