Matlab-guide

Matlab program can be used to make computations and calculations, to run self-made and already existing program listings (macros) and functions, read table-formed data and sound signals for analysing and create tables and sound files. Matlab is an efficient tool for computing vector, matrix and complex number calculations.

Effective usage of Matlab requires the purchasing of additional toolboxes. Some additional toolboxes, e.g. Signal Processing Toolbox, are available on the computers used in the laboratory works of the course Signals and Systems. Therefore, the analysing and producing the signals is easy. Also "Simulink", the simulation tool of Matlab, can be purchased. With help of Simulink you can construct complete systems from smaller sub-systems. These blocks are connected to each other by wiring them with mouse. Simulink is installed on used computers, too.

1. Matlab –short guide

1. Starting up the Matlab:

a) Double-click the Matlab –shortcut  with the left mouse button OR from Start-menu – Programs – Matlab.
b) Wait a moment. Matlab takes a while to start up.
c) The window like in figure 1 will open on the screen. The window consists of the Command prompt, Toolboxes or Workspace and Current directory or Command history -parts and some useful shortcuts. **NOTE! If the window doesn't look like this you can restore the default settings by View - Desktop Layout - Default.**

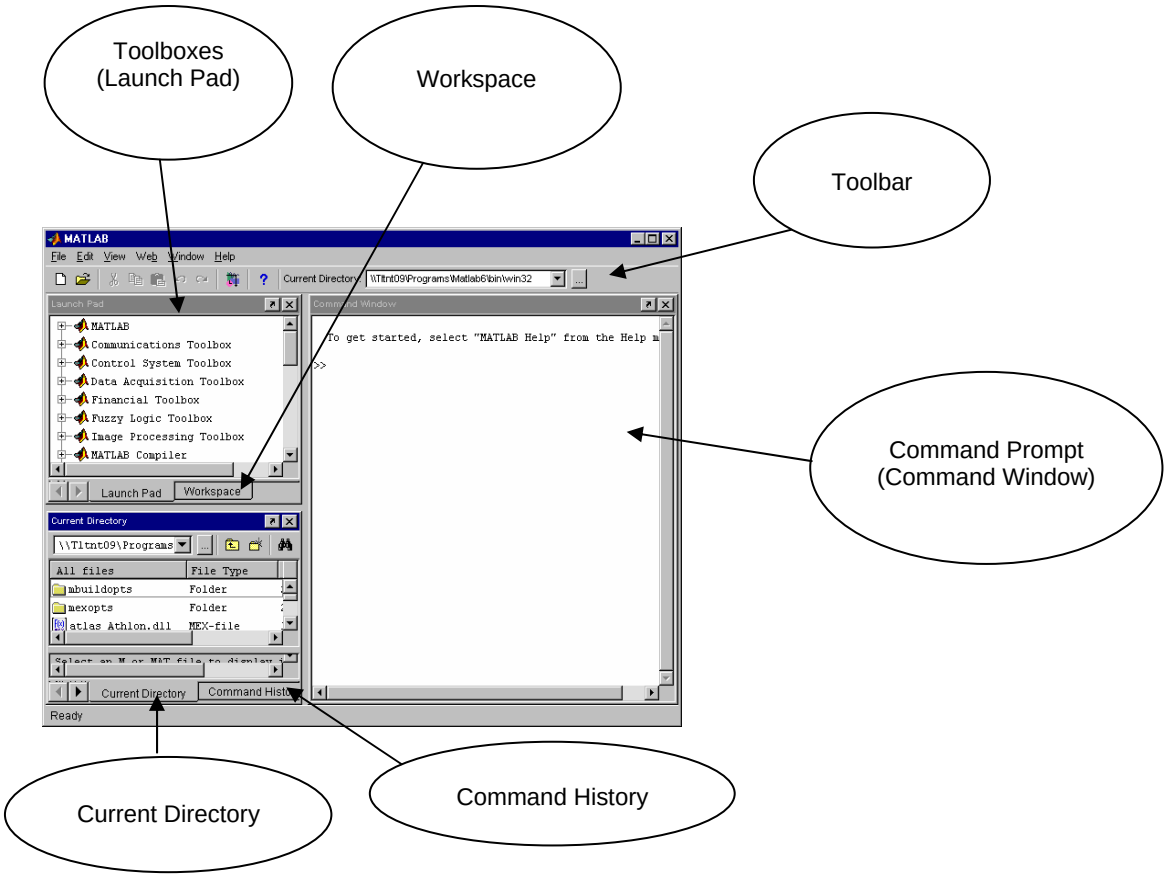

Figure 1. Default Matlab window.

2. You will be able to run the m-files needed in Laboratory works by changing the path that the assistant gives you in the Current Directory window of Matlab (see "Current Directory" in figure 1). This can also be done at the Toolbar's "Current Directory" field.

3. An m-file can be opened for example by double-clicking the wanted filename in the Current Directory window or by writing "`edit filename`" at the Command Prompt. This requires that the Current Directory is selected right.

4. An m-file is run at the Command Prompt by entering the name of the m-file. Remember to save the changes in m-file and check the Current Directory to be right before running the m-file.

5. During the Laboratory works you will have to print quite a lot of graphs and program listings. You can print by clicking the print -button  or selecting the Print command from File-menu. Before printing, check that the print settings are fine and the correct printer is selected. If the graph you are about to print seems to be proper enough, select Page setup from File-menu and choose Size and Position. Click the Fill page -button and Ok so that the figure is printed in the size of the paper. Moreover, you can check how the printed version will look like by selecting Print preview from File-menu. At the end, choose Print.

6. Ending the Laboratory work. Check that you have printed the needed graphs and program listings. Close the figure windows and macros. Enter "`exit`" at the Command Prompt or click the cross in the upper right corner of Matlab window.

2. The parts of Matlab window

Command Window

You can try using the Command Prompt for example by entering the functions:

```
x = log(10)
y = sin(pi*x)/(pi*x) (By the way what is the name of this function?)
z = x*(2-3i)
```

After you have done this, the command window should look like the following (figure 2). If the semicolon ";" is typed at the end of the row, Matlab computes the line without showing the result. If you want to clear the Command window (Command Prompt) select Clear Command Window from Edit-menu.

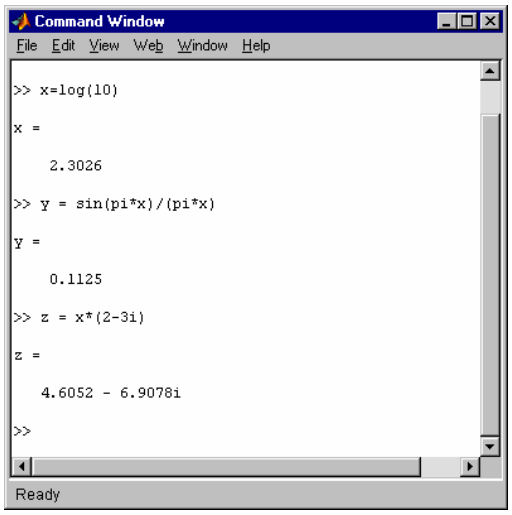

Figure 2. An example view of the Command window.

Launch Pad

Launch Pad is a sub-window where all the Toolboxes currently installed in Matlab can be seen. There is e.g. Signal Processing Toolbox available on the computers used in the Laboratory works. You can browse the toolboxes by clicking the icons with the mouse. Via browsing this sub-window you will find all the instructions, examples, and existing functions that are included in Matlab. Learning to search information from this window is useful.

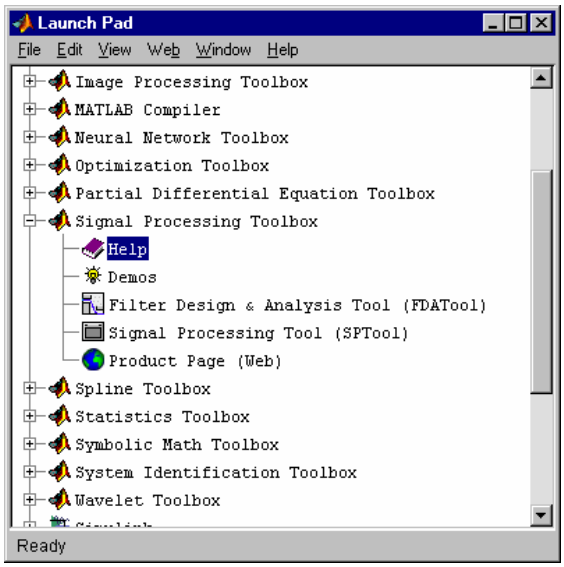

Figure 3. Toolboxes.

Workspace

In the Workspace sub-window the currently existing variables, their dimensions, and types can be seen. The window is got present by clicking Workspace with the mouse (see figure 1). You can call or edit the present variables whenever needed while the session. If you want to clear the Workspace choose Clear Workspace from Edit-menu or type "`clear all`" and enter at the Command Prompt.

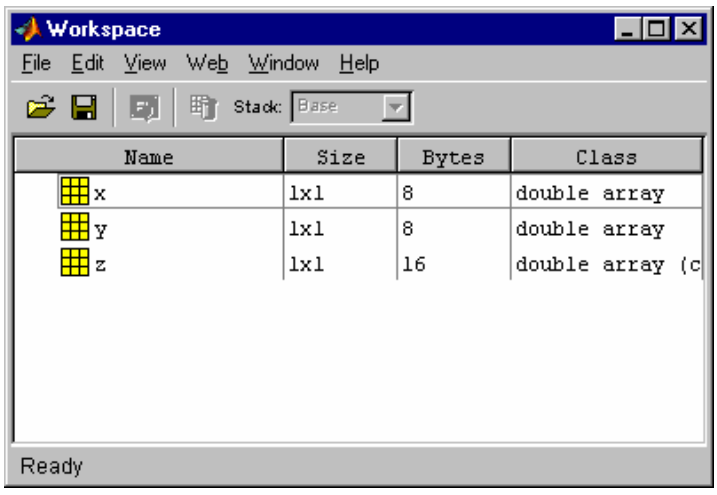

Figure 4. Workspace.

Command History

The history of the recently used commands can be checked from the Command History window (see figure 1). By double-clicking on a command, Matlab executes it again. A more clever way to browse through and re-execute the previously entered commands is to use the arrow-keys (up and down). Matlab executes the command after enter keystroke and therefore, the commands can also be edited before execution. This feature is extremely useful when you have to execute the same command (or a slightly changed version of it) plenty of times. If you need to clear this window, just select Clear Command History from the Edit-menu.

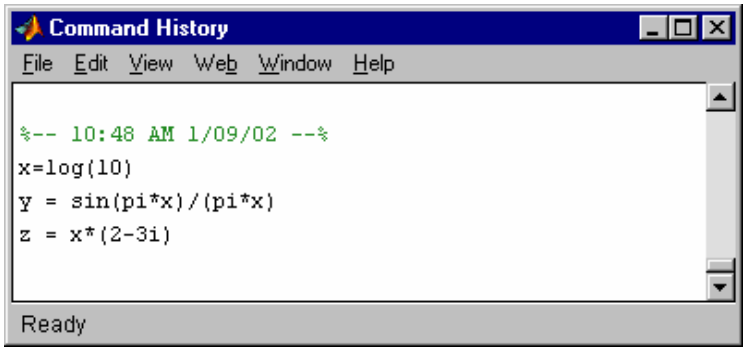

Figure 5. An example of Command History.

Current Directory

In Current Directory window the files and folders can be searched, opened for editing, deleted etc. If you need to run a certain m-file (macro), you will have to define the path where the macro exists in the Current Directory window. For instance, the assistant might announce that the files used in the Laboratory work are located on the hard disk D in the directory courses\signals_and_systems\lab3\ so you have to define the directory in Current Directory window to be D:\courses\signals_and_systems\lab3\. In that case the window will show the files located in the last directory of the path (lab3). The window could look like the following. You don't have to type the whole pathname but instead you can browse the needed directory from the toolbar's Current Directory field (see figure 1).

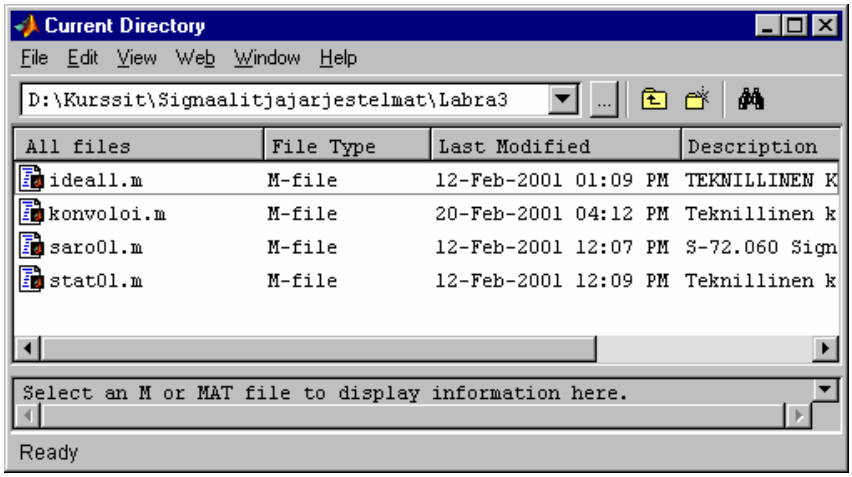

Figure 6. An example of Current Directory -view.

3. M-files

It is not clever to execute long entities (e.g. functions or large amount of consecutive commands) by typing all the commands at the Command Prompt. In such case, it's handy to use m-files (macros). M-file is simply a text file that has .m as file extension. M-files can be used to build up own functions and program listings. After the m-file has been saved, it can be run for example by entering the name of the file in the Command Prompt. In this course you'll have to use some ready-made macros, which can be found in the directory that the assistant gives you. If you have defined the Current Directory to be the one where the macros exist, you can open the macro by double-clicking on it in the Current Directory window. Another possibility is to enter "`edit filename`" at the Command Prompt. Of course there are some other ways to open the files e.g. File-menu – Open... or toolbar's Open shortcut  .

4. Useful hints and tips

1. Use the arrow-keys (up and down) to browse and re-execute the commands (this is explaned more in details in "2 The parts of Matlab window - Command History").

2. With command "Alt + Tab" you can switch between the windows that are currently open in Windows (e.g. Matlab Editor, Plot window, Matlab main window etc.). So keep the Alt-key pressed and push the Tab-key as many times until the "cursor" is pointing the wanted window.

3. When examining the graphs, use the "Zoom"-command. With Zoom you can focus the area of the graph that is interesting and therefore, you get more accurate anrwers and results. In figure 7 there is an example of a plot that has appeared as a result of running an m-file. The Zoom (magnifying glass) is on in the plot. You can zoom the wanted graph by focusing the mouse cursor at one corner of the wanted area (the corner has to be in the white area of the graph) and pushing the left mouse button (keeping it pushed) and draging the cursor to another corner of the area (this corner can be outside of the white area). After that, release the mouse button and the zoom is done. If the zooming operation failed, double-click on the graph so you get back to original situation.

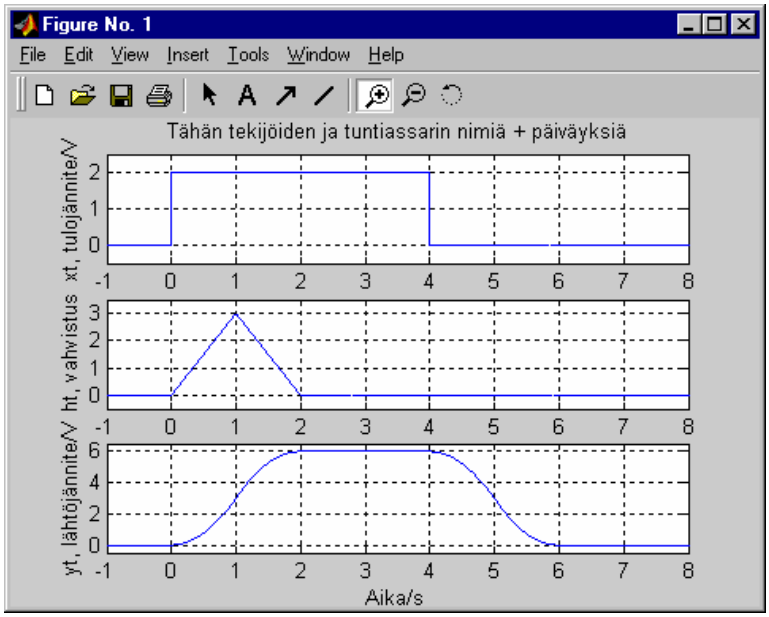

Figure 7. An example of a Zoom situation.

4. If you need information of some Matlab command or function enter "`help command`" in Command Prompt. That tells you about the arguments and use of the command.

5. If an error occurs, the following reasons are noticed to be the most common ones:
   a) comma is typed in decimal number instead of dot
   b) ´ instead of ' used (i.e. the right one is got from the same key as the asterisk character *)
   c) the number of prentheses doesn't match or wrong kind of parentheses used
   d) command in a wrong place
   e) the same parameter or variable is defined again in a lower row
   f) The length of a vector isn't correct. Check the length of the vector `x` by entering "`length(x)`" in the Command Prompt. Call the assistant if the length is somehow wrong.
   g) 1 minute instead of 1 second sound sample is recorded in GoldWave. The format of the sound sample may be stereo even though it should be mono. A suitable value for the sampling rate is 22050 Hz.
   h) Always remember to update the title information of the graphs so that later you can examine the plots reliably. The titles are changed in rows "`title('subject')`" of m-files.
   i) The macro is saved in a wrong place (not to one that assistant gave you).
   j) It is possible that Matlab starts to do something strange or just stops to do anything. In that case restart Matlab.

If these acts doesn't work, ask help from the assistant.

5. Using Help window

The Help window can be opened from Matlab's Help-menu - Matlab Help or by entering "`helpbrowser`" at the Command Prompt. After that the Help window shown in figure 8 appears.

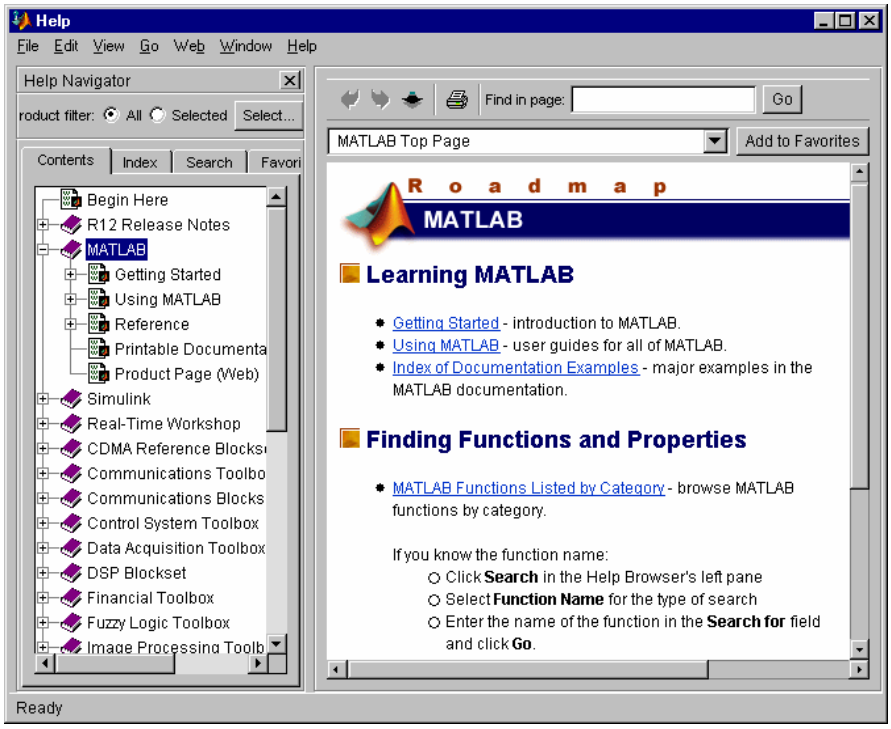

Figure 8. Help window.

If you are seeking some information of a Matlab function, command "`helpwin`" opens the same kind of Help window in which the sub-window on the right shows the different groups of functions (figure 9). If you are looking for information of a certain function, type "`helpwin command`" at the Command Prompt. For example, you want to know what does the command "`square`" do, enter "`helpwin square`" to get the information just about that function.

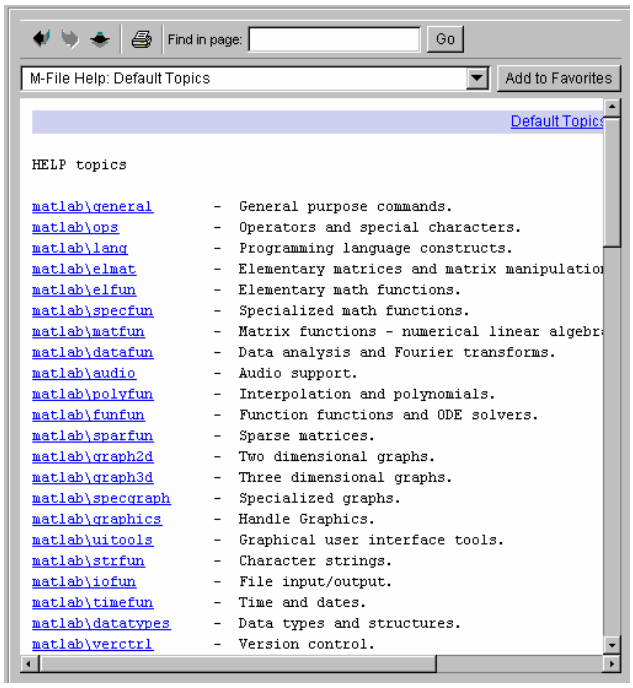

Figure 9. The command "`helpwin`" shows the names of the function groups.

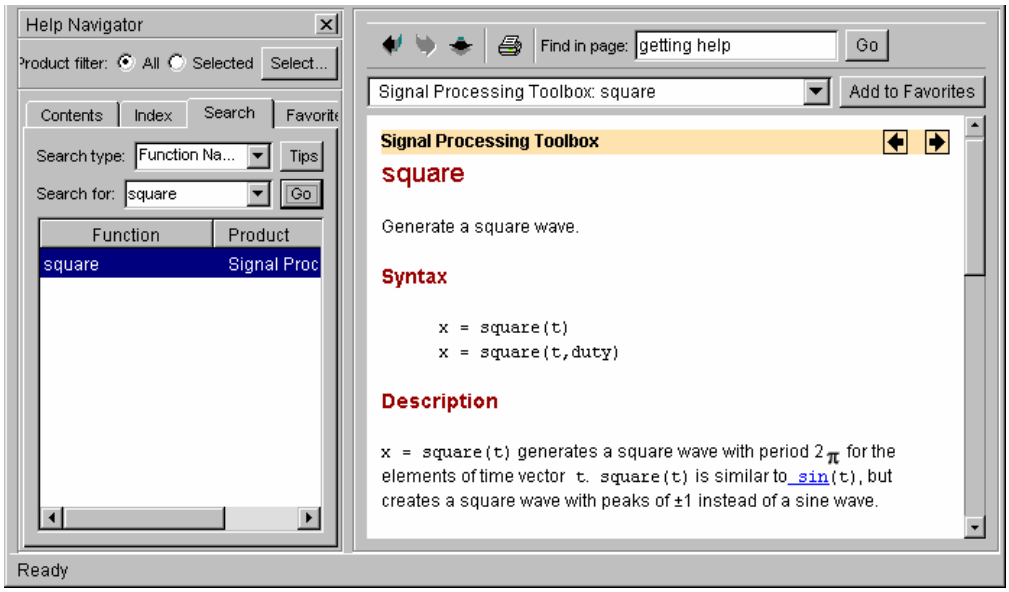

Figure 10. Help Navigator – Search.

On the left side of the Help window in figure 8, there is a sub-window named Help Navigator. With Help Navigator, you can search information from the menu. At Contents-sheet you can browse Help-menu, but finding information may be difficult in here. Easier way might be to choose Search-sheet from the Help Navigator (figure 10). If you write the name of the function in the "Search for" –field and select "Function Name" in the "Search type" –field, available information about the function will appear in the right sub-window. At Index-sheet you are able to enter a free entry. You can add the Help pages of interest to your "Favorites" by clicking "Add to Favorites" –button at such a page. Your Favorites can be found in the Help Navigator's Favorites-sheet.

FINALLY, READ THE FOLLOWING INSTRUCTIONS BEFORE YOU CONTINUE!

Type all the Matlab commands as lowercase letters even though in Matlab Help they are often typed as capital letters. Instead of that the name of a variable can be e.g. HUT, hut or Hut.

Ensure that you don't save the m-files as the already existing functions, e.g. square. Good advice is to save own m-files as a name containing numbers, e.g. square01.m etc.

Sometimes you may leave a variable undefined in an m-file. However, you may use that undefined variable (let's say *f*) in a formula in m-file. In this case you can define the value of *f* at the Command Prompt and the macro will get the value of *f* from Workspace.

The variables of executed macros and commands can be checked at the Command Prompt by typing the name of the variable. The commands entered at the Command Prompt can be browsed using the up and down arrow-keys. These features save you from unnecessary work! With "`clear`" -command you can clear the variables from memory. With `close all` –command all figures are removed.

There is a command named "`function`" in Matlab. Using that command you are able to define you own functions. For example you may make function log2, which computes the logarithm of base 2 of a number. If you will use Matlab even little more than in these Laboratory works, it's strongly recommended to get familiar with this feature (but not right now).

Matlab handles the "analog" signals as discrete (in x- as well as y-dimension). FFT doesn't equal to true Fourier Transform. So there is always some error present. In these Laboratory works the signals are tried to make as accurate as possible so that the phenomena taught on the course can be found.

Matlab and many other simulation and computation tools are extremely useful for engineers. However, the use of these tools requires good knowledge of the things to be simulated so to be sure of the results. If the engineer doesn't master the tool, the consequences might be horrifying. Imagine if an engineer designs using his/her reliable simulation tool the need of GSM base stations of a certain area to be 100 (e.g. à 200 000 €), but the true need is only 10. Not good. An engineer is a person who is able to design the simulation tools itself!!

In science, there are healthy suspects of everything. Because of that, it is useful to have a thought if Matlab computed this (a thing under examination) right or is there something that wasn't taken account. Could the simulation be more reliable or accurate if I repeat the simulation e.g. 10 times. Etc, etc...