Computer Exercise: Matlab introduction

Working in Command Window and using Graphics

This part of the laboratory assignment is about how to work with commands directly in Matlab's Command window. You will work interactively, i.e. you enter a Matlab command and get an immediate response in the Command Window (or in a graphics window if you enter graphics commands).

To do

Open Matlab

1. Open Matlab by double clicking on the Matlab icon. If there are several Matlab versions on your computer, open the most recent version.

2. When the Matlab window comes up, begin with ‘moving Matlab’ to the right folder/directory (where you want save your Matlab files). In the top of the window, you’ll find a box:

   ![Folder Icon]

   Click on the icon ![Folder Icon] and find your way to your folder. You should be able to see the path to the folder in the box.

   It is a common mistake that Matlab files has been saved in one folder on your computer, while Matlab's work folder is a different one. The result is that Matlab can’t find your files. Always begin your Matlab session by ‘moving’ Matlab to the right folder.

Getting help in Matlab

3. It is not possible to remember all Matlab commands and how they are used. Instead, use Matlab’s online help. In the Matlab window you find the box

   ![Search Documentation Icon]

   One way of getting help is to use Search Documentation. Assume you would like to
know how to use the exponential function $e^x$ in Matlab. Try to type the word *exponential* in the *Search Documentation* box. Matlab will now open a window with different options. Pick the one that seems to cover what you are looking for, and read the text describing the Matlab command for $e^x$.

Another common way to get help is to simply prompt in `help` or `doc` followed by the name of the command or function for which you want help. In the Command Window, type

```
    help exp
```

and

```
    doc exp
```

You will now get the same help as before, but in two different ways.

*Always use Matlab’s online help, when you are not sure how to use a command.*

Matlab Windows

The Matlab Window consists of various other windows, including *Command Window*, *Workspace* and *Command History*.

It might look slightly different on your computer as the exact position of the windows, and which of them are open might vary depending on the settings.
You can change the window layout by using the Layout button.

4. Use Layout and make the window Command History visible. Also change Command History to a pop up window. Make it docked again.

5. Sometimes windows end up where you don’t want them, or they might disappear by mistake. In those cases, you can reset it using Layout => Default. Try it!

The Command Window
If you prompt a Matlab command in the Command Window and enter Return key, the command will be executed immediately and Matlab will return the result.

8. Now, calculate \( e^2 \sqrt{3} \sin(0.8) \) in the Command Window by writing

\[
\text{exp}(2) * \text{sqrt}(3) * \sin(0.8).
\]

Note that you have to write the multiplication sign \(*\.

9. The result of the calculation above is automatically stored in the ans variable. Now, calculate the square root of 2, \( \text{sqrt}(2) \) in Matlab. The result will be stored in ans. Note that previous result in ans will disappear (be over-written).

10. If you want to save the result so that you can use it later in the Matlab session, you have to save it in your own variable. You achieve this through an assignment. In Matlab the equal sign ‘=’ is the assignment operator. To calculate \( e^2 \sqrt{3} \sin(0.8) \), and store the result in a variable \( z \), you type

\[
z = \text{exp}(2) * \text{sqrt}(3) * \sin(0.8)
\]

The variable \( z \) is now assigned the value of the right hand side. Note that an assignment always goes from right to left. The variable that is to be assigned a value must be on the left side of the equal sign.

11. Thus, the equal sign ‘=’ in Matlab, does not denote equality, but assignment. For equality, to check whether two values are equal, use ‘==’. Type the following in the command window:

\[
\begin{align*}
a &= 1 \\
b &= 3 \\
c &= a \\
a &= b \\
a &= c
\end{align*}
\]

The first three lines are assignments, i.e. variables \( a, b \) and \( c \) are assigned values. But what about lines 4 and 5? Try to interpret the results.

\[
\begin{align*}
\text{=} \quad \text{(equality sign)} \quad \text{means assignment and is always directed from right to left}
\end{align*}
\]

Equality is denoted by == in Matlab, and it is either true or false

**Workspace and Command History**

12. Look at the window *Workspace*. There you will find the variables that you have worked with in the Matlab session up to now.
   In the *Command window*, give the command `clear z`. What happens?
   In the *Command window*, give the command `clear` or `clear all`. What happens?
   Alternatively, right-click on a specific variable in *Workspace* and choose *delete*.

13. Look at the window *Command History*. There you will find all the commands you have given up to now.
   You can repeat a command by double clicking the command in *Command History*.
   Use that method to repeat the calculation where *z* is assigned a value. Note that *z* will appear in the *Workspace window* again.

   You can also use the mouse and drag a command in *Command History* to *Command Window*, and then press *Return* to execute it. Use that method to calculate *z* again, but replace `exp(2)` with `exp(4)` this time (use left arrow key on keyboard).

   Commands can also be recalled if you, with cursor in *Command Window*, scroll backwards or forward with the *arrow up* or *arrow down* key on the keyboard.
   Try that method and recall the command that calculated the square root of 2.

   **Never write the same command twice in Matlab! Instead, use Command history**

**Presenting results on the screen**

If you don’t want to see the result of a computation, you end the command with semicolon. Semicolon suppresses output.

Assign `z+5` to the variable *y* and end the line with semicolon: `y = z + 5;`

14. If you would like to display the value of *y*, you simply type *y* (without semicolon) and enter ‘Return’ key. Try it!

15. If you have not given any other instruction to Matlab, it will display the result with 4 decimals.
   Give the command `format long` in Command Window. Display the value of *y* again. What happens?

   Give command `help format` or `doc format`, to get more information. Try some of the formats presented in the help text on your own.

   **Note that the different formats do not affect the accuracy of your results. It just changes how the numbers are displayed on the screen.**
Graphics and plotting
To plot mathematical functions or other information you basically perform three steps in Matlab

- Create the x-axis: a vector with x-values stored in a variable (i.e. \( x \))
- Create the y-axis: a vector with the corresponding y-values stored in a variable (i.e. \( y \)). Each x-value must have a corresponding y-value.
- Plot \( y \) versus \( x \) with command \texttt{plot(x,y)}

16. Plot the mathematical function \( f(x) = \sin(x) + 2 \cos(x) \) on the interval \([ -\pi, 2\pi ]\), following the three steps above:

a. Create the x-axis, a vector with x-values from \(-\pi\) to \(2\pi\). The simplest way here is to use the command
   \[
   x = \text{linspace}(-\pi, 2\pi, 10); 
   \]
   Type the command in the Command Window. Try to understand the command by:
   - look at \( x \) (write \( x \) in the Command Window and \( x \) will be displayed),
   - read the help text for the command \texttt{linspace} by typing
     \[
     \text{help linspace}
     \]
     and then by typing
     \[
     \text{doc linspace}
     \]
   Note that you have to write the multiplication sign (*)
   ! Also note that \( \pi \) is built in to Matlab, denoted by \texttt{pi}.

b. When the x-values are there, you create the corresponding y-values. As the function is called \( f(x) \) here, we simply name the y-values \( fx \) in Matlab. The command
   \[
   fx = \sin(x)+2\cos(x);
   \]
   will create the y-values we are looking for. Enter the command, and there should two vectors of the same length stored in variables \( x \) and \( fx \).

c. Plot y-values versus x-values, i.e. \( fx \) versus \( x \):
   \[
   \text{plot(x,fx)};
   \]
   A graphics window will open and the function is plotted.

The graph does not look good, it is a bit jagged. Why, do you think?

17. To make the graph look a bit better we need to divide the interval into more points than 10. Let’s increase the number of points to 200.

a. Recall the \texttt{x=linspace(.,. .)} command by using ’arrow up’ (on keyboard).
   Change from 10 to 200 and press ’Return’
b. Use 'arrow-up' until you find the command where you created the variable \texttt{fx}. Press ‘Return’. As it is a different \texttt{x}, \texttt{fx} will also be recalculated with 200 values.

c. Use 'arrow-up' again until you find the \texttt{plot}-command. Press 'Return’ and the function will be redrawn. Does it look better?

18. Now when the graph looks the way it should, it’s time to add title and labels on the axis. Give commands

\begin{verbatim}
    title('The function f(x)=\sin(x)+2\cos(x)');
    xlabel('x');
    ylabel('f(x)');
\end{verbatim}

Note what happens in the graph when you type the commands. The text inside the quotes (’’) are called text strings. The commands just display the string that is there.

19. Another way of creating tables such as x-axis is to use the principle

\texttt{x3 = 0:0.2:5}

Try it and try to understand the command. What will happen if you replace 0.2 with 0.5?

20. The plot is created in a Figure Window, and you find a menu bar on top of the window. Click \texttt{Edit} in the \texttt{Tools} menu, and the figure is open for editing, e.g. colors and fonts.

\begin{itemize}
  \item Click \texttt{Edit}, and double click on the line graph. Change the line color to red.
  \item Change the line thickness to 2.0
  \item Double click on the text under the x-axis, i.e. on \texttt{x}. Change the text fond to  bold.
  \item Deselect \texttt{Edit} in the \texttt{Tools} menu
\end{itemize}

This is a way of modifying your graph after it is created. It is also possible to do the same changes by modifying your plot commands.