2 MATLAB Input and Output

2.1 Import and export of data

a) Generate a column vector `a` from -3 to 3 with step size 1. Generate a further column vector `b` from 10^{-3} to 10^{3} with the same number of elements as column vector `a` in logarithmic scale.

b) Save variables `a` and `b` in a MAT-File `aandb.mat`.

c) Delete all variables from the workspace.

d) Load variable `a` from MAT-File `aandb.mat`.

e) Save variable `a` in a ASCII-File `aandb.txt`.

f) Delete variable `a` from the workspace.

Check the workspace with Workspace Browser or `who`–command.

g) Load the data from ASCII-File `aandb.txt`. Show all loaded variables.

h) Load the data from MAT-File `aandb.mat`. Show all loaded variables.

i) Save variables `a` and `b` in such a way in a text-file `aandb.txt`, that both variables can be loaded and clearly separated from the text-file.

2.2 Two-dimensional graphics 1

a) Generate the column vector `w` from 0 to 5 with step size 1.

b) Generate the matrix `evonw` with values $e^{-\mathtt{w}}$ in the first column and with values $-e^{-\mathtt{w}}$ in the second column.

c) Now plot matrix `evonw` at the ordinate with abscissa values `w`. Set the title of the figure to "Matrix evonw".

d) Save the workspace data into file `evonw.txt` in ASCII format.

e) Load the data from `evonw.txt`. Why doesn't it work?

f) Open `evonw.txt` with the text editor or type the content of `evonw.txt` at the command window.

g) Generate the matrix `evonw1` with `w` as first column and `evonw` as second and third column.

h) Save the workspace data in the file `evonw1.txt` in ASCII format.

i) Delete als data in the workspace and check deletion.

j) Load als data from ASCII file `evonw1.txt` (ASCII format) in the workspace and show all variables in the workspace.

Show variable `evonw` as well.

k) Generate the same plot as above with the new variable `evonw1`.

l) Finally, the plotted figure should be saved to file `evonw1.eps` with format "Encapsulated Color PostScript".

2.3 Two-dimensional graphics 2

a) Generate the column vector `w` from -2 to 2 with step size .5.

b) Apply the `ceil`-operation to `w`.

c) Show `w` and `ceil(w)` in the command window with `[w ceil(w)]` and compare the row values.

d) Plot `ceil(w)` at the ordinate with abscissa values `w` as red colored line.

e) Plot `ceil` with `fplot` in the range from -2 to 2 as broken line in the same figure.

The former plot should **not be deleted** in the figure!

f) Plot `floor` with `fplot` in the range from -2 to 2 as green dash dotted line in the same figure.

The former plot should **not be deleted** in the figure!

g) Plot a legend in the figure for all three lines with appropriate names.

h) Finally, the plotted figure should be saved to file `ceil.eps` with format "Encapsulated Color PostScript".

2.4 Three-dimensional graphics 1

a) Generate the vector `t` from -6π to 6π with step size 0.1π.

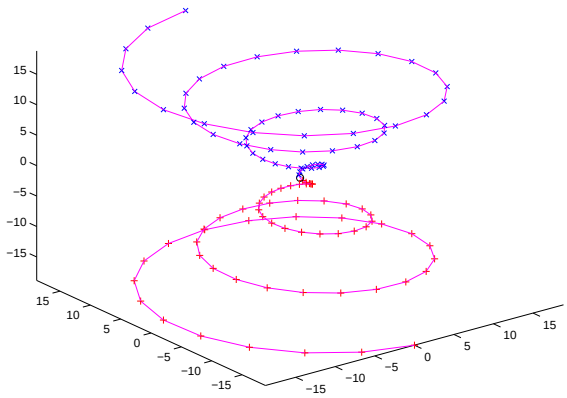
b) Open a new figure as Three-dimensional plot (command `view`).

c) Set the minimal/maximal borders of all three axis to the minimal/maximal values of `t`.

d) Plot the values $x = t \cdot \sin(t)$, $y = t \cdot \cos(t)$ and $z = t$ for each element of `t` with 3D plot command `plot3`.

Each single point should be plotted as red cross for `t` < 0, as blue star for `t` > 0 and as black circle for `t` $= 0$.

Between the plot of subsequent points a pause of 0.05 seconds should be made.



e) Connect the points by a magenta colored line.