Introduction to Programming 2016-2017
Exercise 8

Question 1

A severe sleep disorder affects Unicorns during early adolescence. In an effort to identify the disorder, sleep assessment is performed in sick Unicorns and in a control group using the function '**compActivity**'.

The function receives **two** arguments:

1. **matFileID** (string) **-** Name of a data file containing a cell array **(data**); data contains rows of ID (number), Name (string) and State ('C' or 'S'), see example below.
2. **matFileSleep** (string) **–** Name of the data file containing a matrix **sleepFract** of *n* Unicorns probability of sleep during each hour of the day (*matrix n*24*).

The function returns **two** arguments:

1, 2. **sickStruct, controlStruct** – structures with the following fields relating to the sick/control animals respectively (calculate the values using a common function).

- **numAnimals** (*scalar*) – Number of Unicorns in the group.
- **maxAnimalName** (*string*)– Name of the Unicorn which slept the most in the group.
- **meanHour** (*vector 1*24*) **–** Mean probability of sleep during each hour of Unicorns in the group.

The program should plot two subfigures within a single new figure:

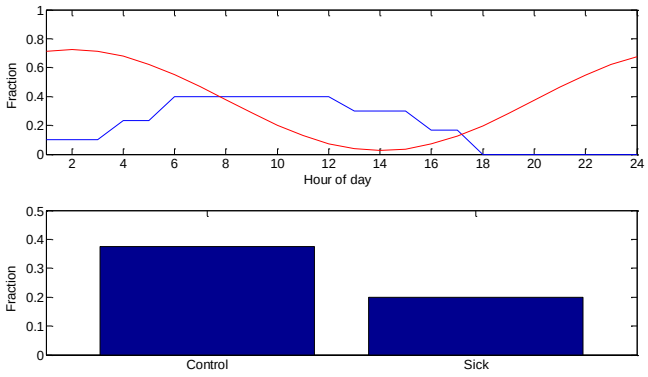
1. (Top) Plot the mean probability of sleep of the sick (blue) and control (red) Unicorns within each hour.
2. (Bottom) Bar graph of the mean total sleep of the two groups side by side with the correct x labels

Example: [sick, control, error] = compActivity('Q1ID.mat', 'Q1Sleep.mat');

Reads the **matFileID** file, which contains a cell array with the values:

ID	Name	State
2	Primara	S
4	Blissia	C
7	Astra	S
5	Yashiana	S
3	Della	C

and plots:



Question 2

a. A sigmoid function can be described by the formula: $y = \frac{a}{1+e^{-bx}}$. Write a function '**sigmaFunc'** that gets three input arguments: a data vector, value for the 'a' coefficient, value for the 'b' coefficient (in this order), and returns one output argument which is a vector of *y* values after applying the sigmoid formula on the data vector (using the specified 'a' and 'b' values).

b. Write a function called '**sigmaPlot**' with no input and no output arguments, which calculates 4 sigmoid functions with different coefficients and plots them on the same graph. The functions should be calculated by calling *sigmaFunc* using the vector [-20: 0.05 :20] as data vector and the appropriate coefficients.

The following coefficients should be used:

- a= 1, b= 1. The function should be plotted in blue solid line.
- a= 3, b= 0.5. The function should be plotted in red dotted line.
- a= 1.5, b= 0.3. The function should be plotted in green dashed line.
- a= 2, b= 3. The function should be plotted in black dash-dot line

Add appropriate title, axes labels and legend to your figure.

c. Write a function called **sigmaPlotB** with no input and no output arguments, which does the following:

- Calculates 3 sigmoid functions with the same coefficients (a=4, b=0.5) but different x-data:
  - i. x = [-50 : 0.5 : 50].
  - ii. x = [-50 : 5 : 50].
  - iii. x = [-50 : 10 : 50].

The sigmoid functions should be calculated by calling the *sigmaFunc* function from section *a*.

- Create a **new figure** with 3 subplots, ordered in one column, and plot the three sigmoid, one on each subplot (in the same order they appear above), as a function of their respective x-data. Make all plots in solid black lines.
- Add for each subplot a relevant x-axis label, y-axis label, and title. The title for each graph should include the increment of the x-data with which the sigmoid was calculated. Change the Y-axis limits (of each subplot) to be from 0 to 6.

Notes for question 2:

- Do **not** use handles of the graphical objects in this question (and do **not** use the 'get' and 'set' commands) - use only the direct graphical commands such as xlim, ylabel, title, etc.
- Exponentials in Matlab can be calculated by the 'exp' command. For example, the Matlab expression *exp(x)* calculates the mathematical expression e^x.

Question 3

The attached function '**ex7Handles**' creates a figure with graphs in it and returns one output argument which is a handle to the line in the top subplot. The function gets no input arguments.
In this question you are required to call the **ex7Handles** function, get its returned handle, and change the appearance of the generated graph by navigating the handles hierarchy based on the returned handle:

Write a function called '**navHandles**'. The function should receive no input arguments and return no output arguments.
Inside the function, call the function '**ex7Handles**' and get its returned value, i.e., the handle to the line object. Use this handle to get handles to other objects in the graph and use them to change the following properties:

Figure properties:

- Name: 'ex 7 figure'.
- Color (background color): [0.2 0.8 1] (RGB).

Subplot 1 (top) properties:

Axes properties:

- Color (background color): [0.6 0.2 0.5] (RGB).
- Font size: 15.
- Axis line color (xColor, yColor): red.
- Ticks of the Y-axis (change the ticks without changing the limit): from 0 to 1 with interval of 1. Then change the label of the ticks to 'min' and 'max', respectively.
- Title: 'sine function'.
    - Color: [1 0.1 0.4]

Line properties:

- Color: black
- Line width: 3.

Text object properties:

- Color: blue
- Font size: 20

Subplot 2 (bottom) properties:

Axes properties:

- Color (background color): [0.2 0.9 0.3] (RGB).
- Y-axis limits: from -1 to 2.
- Ticks of the x-axis (change the ticks without changing the limit): from 0 to 100 with intervals of 25.
- Labels to the x and y axes: 'x data' and 'cosine values', respectively.
    - Font size (of the labels): 14
- Title: 'cosine function'.
    - Font size (of the title): 17
    - Color: yellow

Line properties:

- Color: white
- Line style: dashed
- Line width: 2
- Marker: Hexagram
- Marker size: 10

Notes for question 3:

- Do not use the ‘gca’ or ‘gcf’ functions. Get all your required handles by using the 'parent' and 'children' properties of other objects.
- Do not use direct MATLAB graphics functions such as xlim, ylabel, title, etc. Use only the graphic **handles** to change the **properties** (by the 'get' and 'set' commands).
- Do not change the XData or YData of the plots.

General notes:

1. There is **no need** to print and submit the figures on hard-copy, and no need to submit them with the soft-copy.
2. Submission date: Soft copy should be submitted by Sunday (8.1.17) - 24:00.
3. Hard copy should be submitted in the Tirgul group.

Good luck!