**Homework #2: Data types, algebra, plotting and loops** **Due: 5:00 PM 09/09/16**

Please read the following questions carefully and make sure to answer the problems completely. In your MATLAB script(s), please include the problem numbers with your answers. Then use the *Publish* function in MATLAB to publish your script to a *pdf* document. For more on the *Publish* functionality within MATLAB see `http://www.mathworks.com/help/matlab/matlab_prog/publishing-matlab-code.html`. Upload your *pdf* file to Blackboard under Assignment #2. Your filename should be *GEOS397_HW2_Lastname.pdf*. Hint: You can achieve this automatically by calling your MATLAB script *GEOS397_HW2_Lastname.m*.

### Preliminaries

Use comments to document your code, and distinguish the different *Problems* using "Sections". I suggest that you review two MathWorks, Inc. webpages.

- Improving code readability
- Using 'Sections' in your coding

The goal of this homework set is to familiarize you with some basic MATALB variables and functions that we will use regularly during this course. In the following document, variable names are distinguished in *italics* and MATLAB functions are in **bold**. Remember that you can type **help functionName** or **doc functionName** to get the MATLAB help for a given function (you can also search the MATLAB website or Google). Make sure to answer all parts of the questions. The easiest way to do this is by including answers in your MATLAB script and using the Publish button on the *Publish* tab within MATLAB.

## Problem 1: Variables and time *(20 pts.)*

A) Create a variable *time* using the **clock** function. Answer the following:

- What is the size of *time*? *(Hint: use the **size** function.)*
- Is *time* a row or column vector?
- What does *time* contain? (be specific)
- What variable class is *time*?

B) Use **datestr** to make a new variable called *yearString* that is only the year. *Hint: you may need to read the **help datestr.***

C) Save *time* and *yearString* in a '.mat' file. Call this file whatever you like. *(Hint: Use the **save** function in MATLAB.)*

FYI: **datenum** is another useful MATLAB function that is worth looking into when you have time. For example type **doc datenum** into the command window.

# Problem 2: Plotting sine and cosine waves *(30 pts.)*

A) Create a vector using **linspace** that goes from tMin=0 to tMax=1 second. Call this vector *tArray* and make sure it has 1001 elements. Answer the following:

- What is the sample interval (in seconds) of *tArray*?

B) Make a variable $f$ and set it equal to 5 Hz.

C) How do you convert $f$ to angular frequency? Do this conversion and call the new variable $\omega$.

D) Compute the cosine and sine of $\omega$*tArray* and assign each of these to a variable. *Hint: Make sure you use the built-in MATLAB sine and cosine functions and pay attention to units!*

E) Plot these two curves on the same plot as a function of time (i.e. time on the x-axis). *Hint: you may need to look at the* ***plot*** *and* ***hold*** *commands.* Plot the cosine function in red with a solid line and plot the sine function in blue with a dashed line. *Hint: use* ***help plot***, ***doc plot****, or search online for MATLAB plotting help.* Make sure to label the $x$ and $y$ axes.

F) Use **legend** to add a legend outside of the plot window in the upper-right corner of the figure.

G) Use **axis** to set the x-axis limits to [tMin, tMax] and the y-axis limits to [-1.5, 1.5].

H) Use **grid** to turn on a background grid in the plot.

I) Starting from t=0, determine if you did in fact compute 5 Hz sine and cosine waves. Discuss your observations of the differences and similarities between the sine and cosine waves.

J) What is the amplitude of the waves?

K) How could you change the amplitude?

# Part 3: Population growth *(50 pts.)*

## Preliminaries

A typical computer program follows this logical sequence:

1. Define parameters and constants
2. Create the model space (includes time if necessary)
3. Compute calculations (using flow control and loops if necessary)
4. Plot results

In this problem, the model will be of rabbit population growth based on a growth factor per unit time. The easiest way to visualize a population growth model is to assume that some proportion of the population reproduces during each time step and some proportion dies. The difference between the two is the growth factor. The growth factor therefore must be a number between 0 and 1. If the factor is 0 then the population does not grow, as the number of rabbits born equals the number of rabbits that die in a time step. If the factor is 1 then the population doubles at every time step (i.e. 100% survival and no deaths). Think of the time step in this model as one gestation period.

The differential equation representing this system is

$$\frac{dR}{dt} = b * R_o. \tag{1}$$

This says that the change in rabbits with time ($dR/dt$) is equal to the growth rate ($b$) times the initial population ($R_o$). We can solve this differential equation for the total population at a given time as follows:

$$dR = b * R_o * dt \tag{2}$$
$$dR = R_n - R_o = b * R_o * dt \tag{3}$$
$$R_n = R_o + b * R_o * dt. \tag{4}$$

$R_n$ is the total number of rabbits (i.e. population) given the number of rabbits initially ($R_o$) for a time step ($dt$) and a growth factor ($b$). A simple description of this model is *Have = Had + Change*. Our goal in this problem is to implement this model in a MATLAB script and compute the rabbit population. We want to create the script in such a way that it is simply a matter of changing $b$, $dt$, and $R_o$ and then rerunning the script to get the new rabbit population $R$.

A) Create a variable called *initialRabbits* and set it equal to 2. (We need at least two rabbits to start the model.) The unit of time in this problem will be a "gestation unit". Create a variable called $dt$ and set it equal to 1. This means our time step for the computation will be 1 gestation unit.

B) Create a variable that represents the maximum number of gestation units you want run your model for. *Suggestion: Keep it small to start with; perhaps less than 10.* Create a growth rate variable and give this a value.

C) Use a **while** loop to run your model from time zero to your maximum number of gestation periods. In the loop you need to keep track of the total number of gestation units and the total number of rabbits; also remember that during each iteration of the loop you need to reset/update $R_o$, the initial rabbit population.

*Hint: To keep track of the population and gestation periods, the easiest approach is to add a new element to an array. At the end of the loop, you should have two variables (time and rabbits) that you can then plot*

*against each other to see how the population grows with each gestation period. (Remember "gestation period = time" in this model.)*

D) Plot the results: use the **subplot** command to make a plot with two columns. In the left column plot the normal rabbit population. In the right column plot the $\log_{10}$ of the rabbit population. Make sure to label the axes (search Google/MATLAB for how to do this). Use the **grid** command to turn on the background grid lines.

E) Vary the value of the growth rate and rerun the model a few times. Describe your observations of the population change with varying growth rates.

F) Look up the gestation period for rabbits; make sure to cite your source. Based on the value you found for rabbit gestation period, how many gestation periods are there in one calendar year? Given the number of gestation units in a calendar year and starting with two rabbits and a growth rate of 1, how big will the population of rabbits be after 1 year.