Math 451 ASSIGNMENT 2 **Due at 3 PM Wednesday, April 2nd**

Here are the outcomes for this assignment:

1. Understand the mathematical notation and *MATLAB* syntax for sequences.
2. Create a sequence defined by a recursion formula iteratively, using a for loop.
3. Plot the terms of a sequence.
4. Create a document using the *Publish* feature of *MATLAB*.

For this assignment you will be working with the Fibonacci sequence. A sequence is just a (possibly infinite) list of numbers, like

$$1, 2, 4, 8, 16, 32, ... \tag{1}$$

The numbers in a sequence are called the **terms**, and we often denote all of them with the letter a, using subscripts to tell them apart. A general sequence is then a list of the form

$$a_0, a_1, a_2, a_3, a_4, ... \qquad \text{or} \qquad a_1, a_2, a_3, a_4, ... \tag{2}$$

Each subscript is called an **index** (the plural is *indices*). Using the first of these, for the sequence in (1) we would have $a_0 = 1$, $a_1 = 2$, $a_2 = 4$, and so on. We can see that $a_n = 2^n$ for $n = 0, 1, 2, ...,$ which is what we call an **explicit** formula for determining the terms of the sequence. We can also see that each term of the sequence is twice the previous term, so we could also generate the terms of the sequence using the **recursion** formulas $a_0 = 1$, $a_k = 2a_{k-1}$ or $a_0 = 1$, $a_{k+1} = 2a_k$.

Because of how *MATLAB* works, any sequences we create will be like the second sequence in (2), with the first index being one. We won't subscript, but will use function notation instead, with a sequence like the second one in (2) being given by

a(1), a(2), a(3), a(4), ...

Another difference between the mathematical notation given in (2) and the *MATLAB* syntax is that we will often use a full word or several letters to name each term of a sequence in *MATLAB*, rather than just one letter. You will see an example of this as you work on the assignment.

The particular sequence we will be working with is called the **Fibonacci sequence:**

$$1, 1, 2, 3, 5, 8, 13, 21, 34, ...$$

Can you see the pattern here? That's right - each term is the sum of the previous two terms. We can write this as $a_k = a_{k-1} + a_{k-2}$. Of course this makes no sense for any value of k less than three, and for $k = 3$ we need to have been told what a_2 and a_1 are. A full recursive description of the sequence is

$$a_1 = 1, \qquad a_2 = 1, \qquad a_k = a_{k-1} + a_{k-2} \qquad \text{(3) for } k \geq 3$$

You will generate this sequence, using *MATLAB*, as what we call a **(row) vector**, which is basically just a finite list of numbers. We'll call the vector fib, so if we had the first six terms of the sequence it would look like

fib = [1 1 2 3 5 8]

If we asked *MATLAB* for fib(4) it would tell us that ans=3 because the 4th term of the sequence is three.

OK, it's time to get started!

1. In the command window, generate the first four terms of the sequence one term at a time by typing fib(1)=1, fib(2)=1, fib(3)=2, fib(4)=3 at the command prompts. After entering these four, type fib at the command prompt and see how *MATLAB* returns the entire row vector whose **components** are the first four terms of the sequence.
2. You will need to think of generating the terms of the sequence one at a time, but notice that if you type fib=[1 1 2 3 5 8 13 21 34] at the command prompt, it will return the entire vector.
3. Open a new script file and make the first line clear all; *You should start all script files with this command from now on.* **Initialize** the sequence with the commands fib(1)=1 and fib(2)=1, each followed by a semicolon to suppress output.

4. Generate terms three through ten of the sequence using a **for loop**. The command for this loop is `for k=3:10` After that you will need to enter the command for calculating the kth term from the two previous terms. You will then need to end the loop with an `end` command.

5. The objective is to have this file generate the first ten terms of the Fibonacci sequence and display them all just once. Save your file as `fib1` and try running it. Fix it as needed so that it does what it is supposed to when you run it. (To run it, you just type the name of the file at the command prompt and hit *Enter.*)

6. We now want to plot a graph of the first ten terms of the sequence. To see how to create a plot, type `help plot` at the command prompt in the command window. This will give you a description of how to create a plot. Add to your script file a line that plots the terms of the sequence as red "stars" (asterisks).

7. Save your file again, then rename it *yourname_assn2* and send it to me

8. As the course goes on, you will be asked to generate reports relating to various problems we are trying to solve. *MATLAB* has a very nice *Publish* feature designed for doing this. Here you will see how to do this.

   (a) Add some comments to key parts of your code, telling what is going on with each. There are three or four obvious places for this, to my mind.

   (b) Add this line at the top of your code: `%% Fibonacci Sequence 1`. After that, add a comment line with your name and another line with the assignment number. Then add two percent signs right before the start of your code, then add another two percent signs before the line that creates the graph.

   (c) Now click the publish tab at the top of the editor window, and then click the green *Publish* arrow at the upper right. Look carefully at the result it gives you, to see what the publisher does.

   (d) Add something like *Script File* after the second double percents, and an appropriate title after the double percents that are before the line that causes the plot. Then add another double percent At the end of your file, followed by *Discussion.* Run publish again to see what happens.

   (e) Add blank comment lines anywhere you like to give more pleasing spacing, if needed.

   (f) Instead of clicking the green *Publish* arrow, click the downarrow below it. Select *Edit Publishing Options...* and then click on `html` to the right of `Output file format`. Select `doc` and publish again. This will create a *Word* document that you can then edit in *Word.* Save this file as *yourname_assn2* `-` the `.doc` extension will distinguish it from the `.m` file.

9. Now go to the *Discussion* section of your *Word* file and add a few sentences about your experience doing this assignment. Include

   - how difficult the assignment was for you
   - something (if there was anything) that was particularly difficult or challenging to do or understand
   - something you learned that you think might be helpful on future assignments

   Then e-mail me the file.