Phys234, 2018, Problem set #2: Due Monday February 5, 8pm

Question 1:

Write a function m-file `function A = tridiagv(a,b,n)` that uses the built-in Matlab function `diag` to create a symmetric tridiagonal n by n matrix that looks like

$$\begin{bmatrix} a_1 & b_1 & 0 & 0 & \cdots & 0 \\ b_2 & a_2 & b_2 & 0 & \cdots & 0 \\ 0 & b_3 & a_3 & b_3 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & 0 & \cdots & b_{n-1} & a_{n-1} & b_{n-1} \\ 0 & 0 & \cdots & 0 & b_n & a_n \end{bmatrix}$$

if a and b are vectors with n elements (with elements identified by the subscripts), but produces the matrix

$$\begin{bmatrix} a & b & 0 & 0 & \cdots & 0 \\ b & a & b & 0 & \cdots & 0 \\ 0 & b & a & b & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & 0 & \cdots & b & a & b \\ 0 & 0 & \cdots & 0 & b & a \end{bmatrix}$$

if a and b are scalar numbers. (*Hint:* check the size of a and b to see if input values are vectors or scalars.) To test your function, write a function file `ps2q1.m` that calls `tridiagv` with the following arguments

a) `tridiagv(2,-1,5)`

b) `a=linspace(1,6,6); b=linspace(11,16,6); tridiagv(a,b,6)`

Question 2, 3 4: will be given at the start of the lab.

Question 5:

In a function m-file `ps2q5.m`, first define the vector `x = [21 22 23 24]` and the matrix `B = ones(3,3)`. Then, write the one-line vectorized operation that has the same effect as the following scalar loop:

```
k = 0;
for i=2:3
   for j=1:2
     k = k + 1;
     B(i,j) = x(k);
   end
end
```

(*Hint*: The built-in `reshape` function will be helpful.)

Question 6:

Write the code for the `evenChecker` function that meets the description of the following prologue:

```
function A = evenChecker(m,n)
% evenChecker Create a checkerboard matrix of ones and zeros
%
% Syntax:  A = evenChecker(m,n)
%
% Input:  m,n = number of rows and columns in matrix
%
% Output:  A = an m-by-n matrix having ones in all elements with even
% values of i+j and zeros in all elements with odd values
% of i+j, where i and j are the row and column indices,
% respectively
```

Test your `evenChecker` function with a `ps2q6.m` function that contains the following calls

```
A=evenChecker(7,7)
B=evenChecker(6,4)
C=evenChecker(4,6)
```

Question 7:

The data in `xy.dat` (that you can find in the data folder on eclass) were produced by adding noise to $t = \pi(\sin(x) + \cos(x))$, for $0 \leq x \leq 5\pi/2$ (i.e., $y = t$+noise). Starting from the function `myplotData` that I showed in class, create a modified version `plotDataAve` that plots the (x, y) and (x, t) data sets, along with a horizontal line that corresponds to the average value of y. The (x, y) data should be plotted with open circles, the (x, t) should be plotted with a solid line, and the average y value should be indicated with a dashed line. Use the `legend` function to label the three curves. Also make sure to label your x, y axes. Test your function `plotDataAve` by calling it from a function m-file `ps2q7.m`.