EE322 Fall 2012: Lesson 5/PS05: Solutions

Introduction to MATLAB, Chapter 5: Control Structures and Chapter 6: Matrix Computations.

1. Section 2.4.3 in Chapter 2 introduces you to allowing users to input values into your program using the *input* function. Use that section as a guide to perform the following:

   In communication receivers, noise plays an important role in how well the system works. In a receiver, noise power can be calculated as:

   $$N = kTB$$

   Where N is noise power in Watts, k is Boltzmann's constant (1.38×10^{-23} Joules/°K), T is the noise temperature in °K, and B is the receiver bandwidth in Hz).

   Write a program that will prompt a user to input the noise temperature in °K and the receiver bandwidth in Hz, and the program will compute noise power in Watts and display "xxx watts", where xxx is the computed value of power. Use your program to compute noise power if noise temperature is 297 °K and bandwidth = 140 kHz.

   Noise power = _____ W

```
T=input('\nInput noise temp in deg K: ')
B=input('Input bandwidth in Hz: ')
k=1.38e-23;
N=k*T*B;
fprintf('Noise power is %e Watts\n\n',N);

% for T=297, B=140e3, Noise power is 5.738e-16 Watts
```

2. Section 3.6 in Chapter 3 deals with writing functions in MATLAB. Write a function called *MaxDistance* that will compute the maximum radio frequency (RF) communication distance between two antennas. The max communication distance (in miles) between an antenna with height h_1 feet and another antenna with height h_2 feet is given by:

   $$d = \sqrt{2h_1} + \sqrt{2h_2}$$

   Your m-file called *MaxDistance.m* should have comments after the *function* line to let the user know how to use the function (an example was provided in the EE322 Course Policy Statement).

   Use your function to compute the max transmission distance between an aircraft carrier (antenna height 148') and a submerged submarine (antenna height 17').

   Distance = _____ miles

```
% see MaxDistance function code below. That code is in a file called "MaxDistance.m"
%  MaxDistance(148,17) = 23.0356 miles
-----------------------------------
function d=MaxDistance(h1,h2)
% function d=MaxDistance(h1,h2)
%   This function computes max comms distance between two antennas, one at
```

```
%   h1 feet high, the other at h2 feet high. Returns max distance in miles.
%   Author: R.W. Ives, USNA, 20 Aug 2012
d=sqrt(2*h1)+sqrt(2*h2);
-----------------------------------
```

Note: The following computes distance over a range of antenna heights (10' to 30'), and you were only asked for an antenna height of 15'. The answer to this problem is

3. Use a *for* loop (section 5.3.1) to print out a table of max communication distances between the aircraft carrier from problem 2 and antennas that vary from 10 ft to 30 ft, in 2 ft increments. Print out this table line by using the *fprintf* function (introduced in Section 2.3.3). These values will be printed in the command window, with the first column being height and the 2nd column being distance.

```
h=10:2:30;

for k=1:length(h)
    d(k)=MaxDistance(148,h(k));
end
fprintf('Height\t\tDistance\n')
fprintf('______\t\t________\n')
fprintf('%6d\t\t%7.2f\n',[h;d])

% results in:
% Height        Distance
% ______        ________
%     10          21.79
%     12          22.22
%     14          22.61
%     16          22.98
%     18          23.32
%     20          23.65
%     22          23.95
%     24          24.25
%     26          24.53
%     28          24.80
%     30          25.07
```

4. Work Problem 6.14. For part (b), you can download the *elements.dat* file from the course website.

```
% Part (a)
mw_oxygen = 15.9994;
mw_carbon = 12.001;
mw_nitrogen = 14.00674;
mw_sulfer = 32.066;
mw_hydrogen = 1.00794;
molecular_wts = [mw_oxygen mw_carbon mw_nitrogen mw_sulfer mw_hydrogen];
No=input('Number of oxygen atoms: ');
Nc=input('Number of carbon atoms: ');
Nn=input('Number of nitrogen atoms: ');
Ns=input('Number of sulfer atoms: ');
Nh=input('Number of hydrogen atoms: ');
molecules = [No Nc Nn Ns Nh];
molecular_wt = molecular_wts * molecules';
% note: the ' gives the transpose of molecules, so we are multiplying a 1x5
% vector times a 5x1 vector, which gives a 1x1 result (i.e., a single
```

```
% value)
fprintf('Your molecular weight is %5.2f\n',molecular_wt);

% Part (b)
load elements.dat;
MW=elements * molecular_wts'  % elements is a 20 row-by-5 column matrix
                              % transpose molecular_wts to make it 5 x 1,
                              % result will be a 20 x 1 matrix of the
                              % answers.

% This command yields:

MW =

   89.0641
  175.1509
  132.0792
  132.0560
  121.1301
  146.0729
  146.0961
   75.0472
  156.1044
  131.1148
  131.1148
  147.1374
  149.1639
  165.1019
  116.0899
  105.0635
  119.0804
  203.1106
  469.6953
  117.0979
```