Matrices & Vectors: (Please save in a file called m_v.m)

1. a) Generate a vector a that starts at 0, has increments of 0.3 and ends at 2.1
1. b) Generate a vector b, with uniformly spaced elements, that starts at 0 ends at 5 having **exactly** 100 elements.

2.

Store the two matrices a follows: $c = \begin{matrix} 2 & -4 & 5 \\ -3 & 5 & 7 \\ 5 & 3 & -8 \end{matrix}$ $d = \begin{matrix} 36 & 80 \\ 7 & 120 \\ -31 & 0 \end{matrix}$

a. Calculate c*d, c.*d where for the second multiplication use only the first two columns of c. Explain the difference between the multiplications. Do the second multiplication by extracting the columns… you needto find out how to do this!
b. Calculate exp(c).
c. Solve the matrix equation $c*x = \begin{matrix} 36 \\ 7 \\ -31 \end{matrix}$ using the \ operator (which will be discussed in class today) and column extraction on d.
d. Calculate the dot product of row 2 of c and column 2 of d and then dividing by the element in row 1 column 3 of c, again by extracting the value.

Plotting: (Please save in a file called plot1.m, plot2.m and plot3.m)

1: Plot cosine as a function of the degree from -360 degrees to 360 degrees:

i. Use grid on feature
ii. Label the x axis with the title Degrees
iii. Label the y axis with the title Cos(x)
iv. Caution!!! Be careful of what cos takes as input

2. Plot y1=ln(x) and y2=ln (2x) for [-1, 4] together on one graph with:

i. Y1 as a solid line
ii. Y2 as a dotted line
iii. Label x axis as x
iv. Label y axis a magnitude
v. Use grid on
vi. Place o title of Natural Logarithm Plot
vii. Have legends showing which graph is which

3. Plot the discrete-time function y=sin (2pi*n/11) for n from 0 to 11 using stem(y) and grid on.

4. Explain in detail each part of the following command; act as if you needed to explain what this command does to someone who has never seen a computer.

fprintf('x(%i)=%20.16e\n',i,x)

this line is from Newton.m

Loops and Structures: ***__Caution__ be sure to properly initialize a values properly where need be, namely before entering any loop…

Create a function named QuadRoots(a,b,c) that takes in the 3 coefficients of ax^2+bx+c and uses the if, else if structure to display a message stating the type of roots as well as calculating the roots and showing them. *** For bonus provide a plot with all labels and find a way to label the roots right where the root is located.

2. Create a function BuildVect(n,k,N) that "builds a vector" of the form

[n^2, n^3, (n+k)^2, (n+k)^3, (n+2k)^2, (n+2k)^3,…. ….,(n+Nk)^2, (n+Nk)^3]

For the values n=2, k=2, N=3.
** Note: you will need to learn how to augment vectors

3. Generate a script that determine the smallest non-negative integer n that satisfies the equation 2^n>=2300.

i. Create a function called PowerOf2(N) that returns the value.
ii. Use a while loop