Table of Contents

PREAMBLE

DO NOT REMOVE THE LINE BELOW

```
clear;
```

QUESTION 1: TEAM NAME! (Change it on Slack)

======================= Imma Firin' My Phazer

QUESTION 2: NAMES AND INTERESTS!

=======================

```
% Tom Stowell EE - Does not know what he expects from signals and
 systems. Wants to lead into real time DSP.
% Sarah CE - Senior one of last classes to take. Wants to learn the
 fundamentals.
% Rachel Romaine CE - Not sure what she expects for the class but is
 excited to revisit the math.
% Isabells Perlmutter EE - Softmore. Also wants to learn the
 fundamentals and a basis in machine learning
% Corinne Meyers EE - Wants to delve into bioelectric after this so
 wants an understanding of the basics.
```

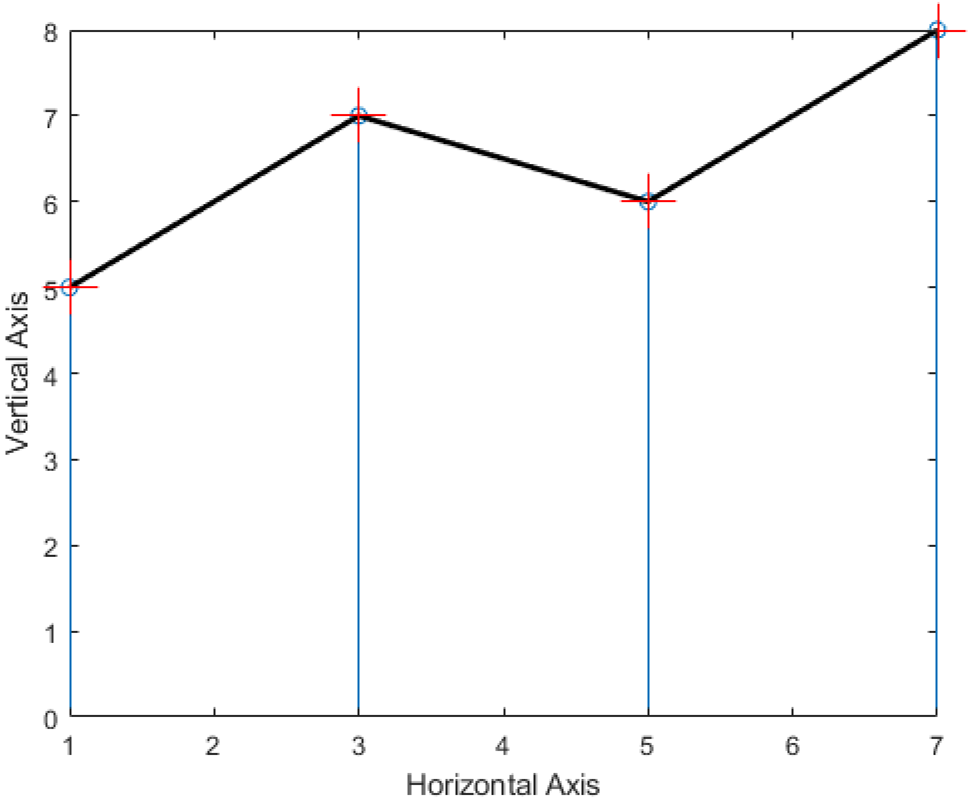
QUESTION 3: COMMENTING

=======================

```
% Copy and comment every line of the following MATLAB script. Say what
% each line is doing in your comment. Explain each MATLAB line by
 using
% no more than one comment line, as done in the first line below. Run
 and
% publish the script:
a=zeros(1,5) % Generate and print a 1x5 row vector of zeros
b=ones(3,2) % Generate and print a 3x2 row matrix of ones
c=size(a); % Returns the size of vector a
abs([-5.2 , 3]) % Rreturns the absolute value of each element in
 array.
floor(3.6) % Rounds 3.6 to the nearest integer less than or equal to
 that element, 3
d=[1:-3.5:-9]; % Creates a regularly-spaced vector from 1 to -9 using
 -3.5 as the increment between elements
f=d(2); % Returnss element 2 in the array d which is -2.5
g=sin(pi/2); % Calulates sin(pi/2) which equals 1
K=[1.4, 2.3; 5.1, 7.8]; % Generates a 2x2 array whith the
 corresponding numbers with the semicolon sperating the rows
m=K(1,2); % Returns the element that is in row 1 column 2 of K
n=K(:,2); % Returns all elements that are in row 1 column 2 of K - all
 of column 2
comp = 3+4i; % Assigns comp to the 3+4i
real(comp) % Returns 3 which is the "real" value of comp and prints
imag(comp) % Returns 4 which is the "imginary/complex value of comp
 and prints
abs(comp) % Returns the magnitude of comp and prints
angle(comp) % Returns the phase angle of comp and prints
disp('haha, MATLAB is fun'); % Prints or displays "haha, MATLAB is
 fun"
3^2 % Calculates 3 to the power of 2 and prints
4==4 % Booleans checks of the two values 4 and 4 are eqaul and prints
[2==8 3~=5] % Boolean checks the vector if 2 is equal to 8 and 3 is
 not equal to 5
x=[1:2:8]; % Creates a regularly-spaced vector from 1 to 8 using 2 as
 the increment between elements
y=[5 7 6 8]; % Generates a row vector of the corresponsing elements

q = zeros(10,1); % Generates a column vector of 10 zeros
for ii = 1:10 % Initiates a for loop beginning at 1 and ending at 10
 (inclusive)
    q(ii) = ii^2; % Sets the value of q(ii) - from 1-10 - to power of
 ii
end % Ends the for loop
figure(1021); % Displays figure 1021
stem(x,y) % Plots the data sequence as stems that extend from a
 baseline along the x-axis
hold on; % Retains plots s so that new plots added to the axes do not
 delete existing plots
plot(x,y, 'k', 'linewidth', 2) % Creates a 2-D line plot of the data
 in Y versus X with linewidth of 2 and linestyle of a black line
plot(x,y,'+r', 'markersize', 20); % Creates a 2-D line plot of the
 data in Y versus X with markersize of 20 and linestyle red crosses
hold off; % Turns of retaining the current plot, new plots will be
 plotted on their own
```

```
xlabel('Horizontal Axis') % Adds a label for the horizontal axis
ylabel('Vertical Axis') % Adds a label for the vertical axis
```

```

a =

     0     0     0     0     0


b =

     1     1
     1     1
     1     1


ans =

    5.2000    3.0000


ans =

     3


ans =

     3


ans =

     4


ans =

     5


ans =

    0.9273

haha, MATLAB is fun

ans =

     9


ans =
  logical

   1


ans =

  1×2 logical array

   0   1
```



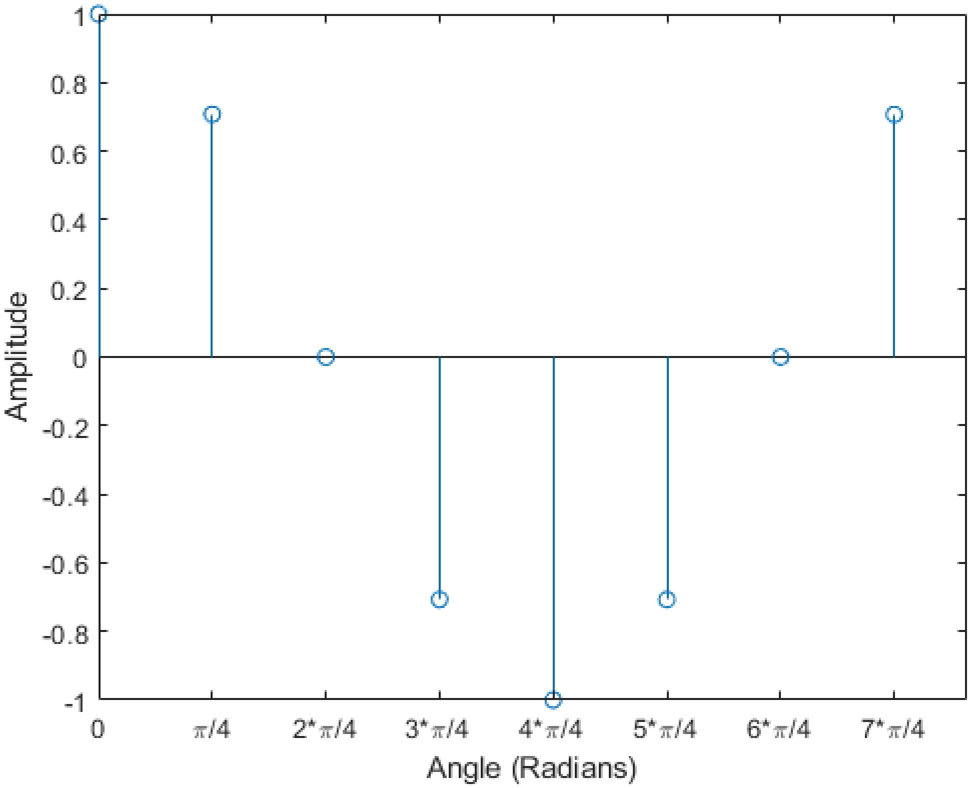
QUESTION 4: PLOTTING

```
======================
```

4(a) PLOT RESULT

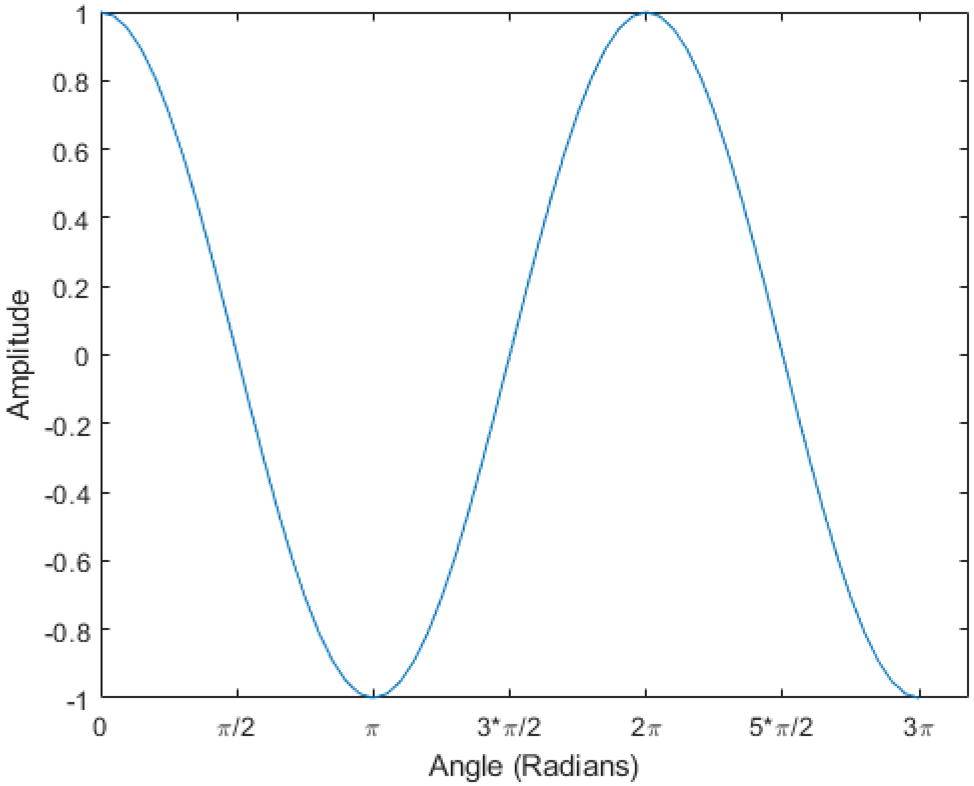
```
vect1 = [0 pi/4 2*pi/4 3*pi/4 4*pi/4 5*pi/4 6*pi/4 7*pi/4];
vect2 = cos(vect1); % Sets y equal to vector cos(vect1)
stem(vect1, vect2); % Plots the defined y vs x using stem
xticks(vect1); % Re-defines the x ticks in increments of pi/4
xticklabels({'0' '\pi/4' '2*\pi/4' '3*\pi/4' '4*\pi/4' '5*\pi/4' '6*
\pi/4' '7*\pi/4'}); % Re-lables the x ticks to match
ylabel('Amplitude'); % Labels the y axis Amplitide
```

```
xlabel('Angle (Radians)');
```



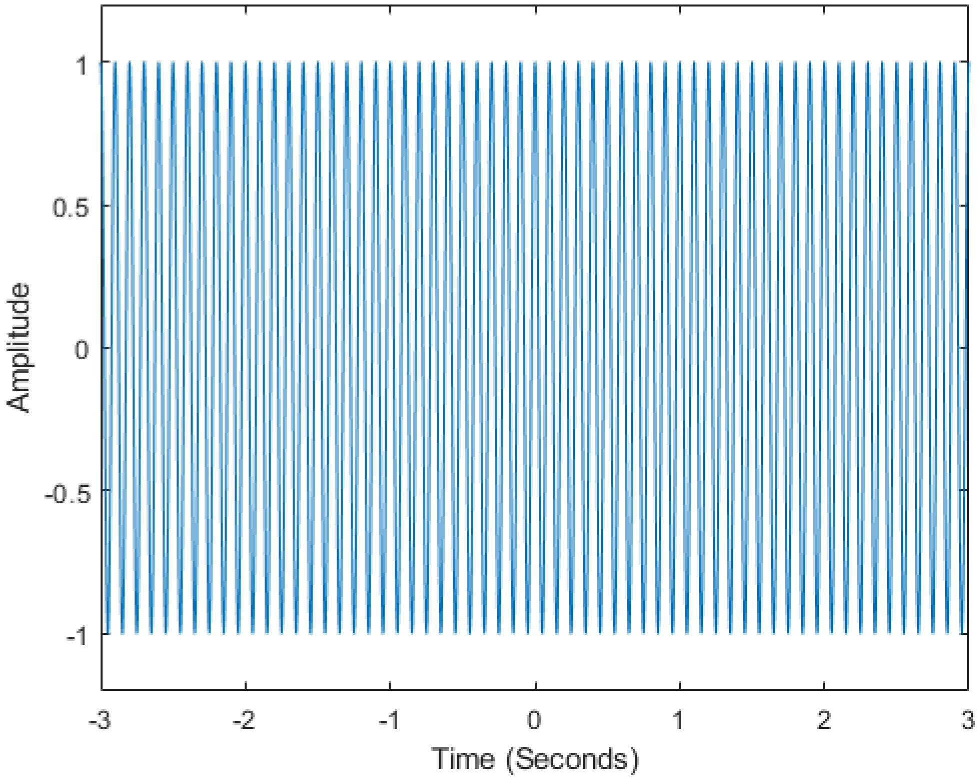
4(b) PLOT RESULT

```
theta = 0:pi/20:3*pi; % Creates a regularly-spaced vector from 0 to
 3pi using 1/5000 as the increment between elements
y = cos(theta); % Sets y equal to vector cos(theta)
plot(theta, y); % Plots the defined y vs x
xticks([0 pi/2 pi 3*pi/2 2*pi 5*pi/2 3*pi]); % Re-defines the x ticks
 in increments of pi/2
xticklabels({'0', '\pi/2','\pi', '3*\pi/2', '2\pi', '5*
\pi/2','3\pi'}); % Re-lables the x ticks to match
ylabel('Amplitude'); % Labels the y axis Amplitide
xlabel('Angle (Radians)'); % Labels the x axis Time (Seconds)
```



4(c) PLOT RESULT

```
x = -3:(1/5000):3; % Creates a regularly-spaced vector from 0 to 3
 using 1/5000 as the increment between elements
y = cos(20*pi*x); % Sets y equal to vector cos(20*pi*x)
plot(x,y); % Plots the defined y vs x
ylabel('Amplitude'); % Labels the y axis Amplitide
xlabel('Time (Seconds)'); % Labels the x axis Time (Seconds)
ylim([-1.2, 1.2]); % Sets the limits of the y axis to -1.2 and 1.2 for
 better clarity
```



QUESTION 5: COMPLEX ROOTS

```
======================
```

5(a) WRITE FUNCTION IN SEPARATE FILE (TEMPLATE PROVIDED)

```
type('myroots.m') % Prints out the entire function


function r = myroots(n, a)
% myroots: Find all the nth roots of the complex number a
%
% Input Args:
%   n: a positive integer specifying the nth roots
%   a: a complex number whose nth roots are to be returned
%
% Output:
%   r: 1xn vector containing all the nth roots of a

% n = input("Enter a positive integer specifying the nth root: ");
% a = input("Enter a complex number whose nth roots are to be
 returned");
```

```
poly = -1; % Defines variable -1
poly(n+1) = a; % Puts the complex number into the n+1 place of the
 poly matrix
r = roots(poly); % Find the roots of the poly matrix
end
```

5(b) ANSWER QUESTION

```
help("myroots") % Uses the first comment block as the description of
 what the function does and prints for the user
```

```
  myroots: Find all the nth roots of the complex number a

  Input Args:
    n: a positive integer specifying the nth roots
    a: a complex number whose nth roots are to be returned

  Output:
    r: 1xn vector containing all the nth roots of a
```

5(c) OUTPUT RESULTS

```
myroots(9,2) % Calculates the 9th roots of 2
myroots(23, -1i) % Calculates the 23rd roots of -j
```

```

ans =

  -1.0149 + 0.3694i
  -1.0149 - 0.3694i
  -0.5400 + 0.9354i
  -0.5400 - 0.9354i
   0.1876 + 1.0637i
   0.1876 - 1.0637i
   0.8274 + 0.6942i
   0.8274 - 0.6942i
   1.0801 + 0.0000i


ans =

  -0.1362 - 0.9907i
  -0.3984 - 0.9172i
  -0.6311 - 0.7757i
  -0.8170 - 0.5767i
  -0.9423 - 0.3349i
  -0.9977 - 0.0682i
  -0.9791 + 0.2035i
  -0.8879 + 0.4601i
  -0.7308 + 0.6826i
  -0.5196 + 0.8544i
  -0.2698 + 0.9629i
```

```
   0.0000 + 1.0000i
   0.1362 - 0.9907i
   0.3984 - 0.9172i
   0.2698 + 0.9629i
   0.5196 + 0.8544i
   0.6311 - 0.7757i
   0.7308 + 0.6826i
   0.8170 - 0.5767i
   0.8879 + 0.4601i
   0.9423 - 0.3349i
   0.9791 + 0.2035i
   0.9977 - 0.0682i
```

Published with MATLAB® R2020a