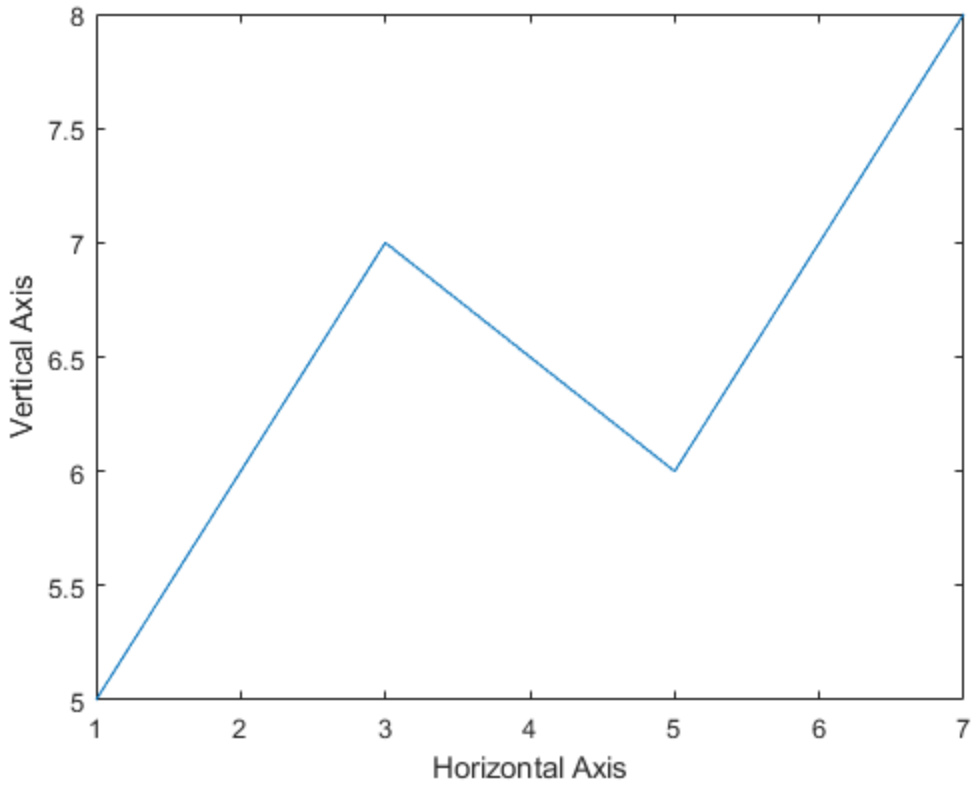
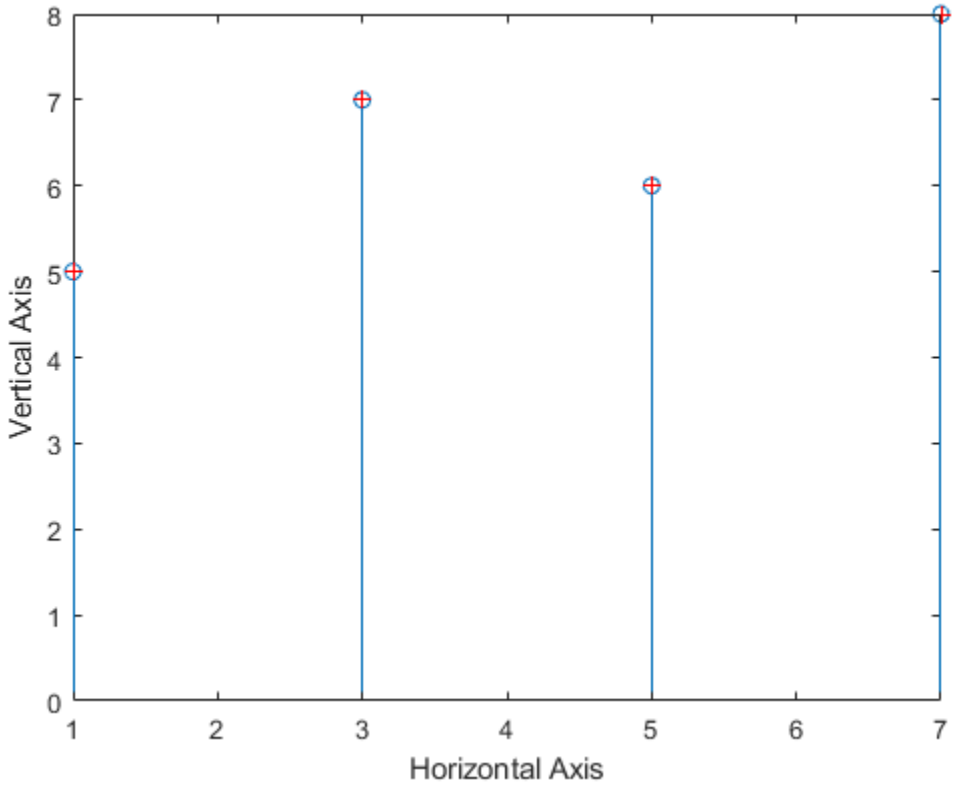
Exercise 1.1 : Koby Miller

```
a = zeros(1 ,5)    % Generate and print a 1x5 row vector of zeros
a = zeros(1 ,5);   % Generate a 1x5 row vector of zeros.
b = ones(3 ,2)     % Generate and print a 3x2 matrix filled with 1's
c = size( a );     % Saves dimensions of matrix a into vector c
abs ([ -5.2 , 3]); % Takes a vector and makes every value positive
floor (3.6);       % Rounds the decimal down
d =[1: -3.5: -9];  % Generates vector from 1 to -9 counting by -3.5
f = d(2);          % Generates the second value in the vector d
g = sin( pi /2) ;  % Generates the equivalent of sin(pi/2)
K =[1.4 , 2.3; 5.1 , 7.8];% Generates a matrix with those values
m = K(1 ,2);     % Generates value in first row, second column of K
n = K(: ,2);       % Generates all of the values in the second row
comp = 3+4i;       % Generates complex number and saves it in comp
real(comp)         % Prints real part of complex number we saved
imag(comp)         % Prints imaginary part of complex number we saved
abs (comp)         % Prints complex modulus of our complex number
angle (comp)       % Prints the phase angle of our complex number
disp ('haha , MATLAB is fun');% Prints phrase 'haha , MATLAB is fun'
3^2                % Prints 9. The value of 3^2
4==4               % Prints 1. Logical true because 4 does equal 4
2==8               % Prints 0. Logical false because 2 does equal 8
3~=5               % Prints 1. Logical true because 3 doesn't equal 5
x =[1:2:8];        % Generates a vector from 1 to 8 counting by 2
y =[5 7 6 8];      % Generates the vector [5 6 7 8]
figure;            % Creates a new figure window
plot(x,y);% Generates a plot on figure window using vectors x and y
xlabel('Horizontal Axis') % Lables the x axis
ylabel('Vertical Axis')   % Lables the y axis
figure(1021);     % Creates a new figure window named 'Figure 1021'
stem(x,y)% Generates stem plot on Figure 1021 using vectors x and y
hold on; % Sort of pauses the current plot and its properties
plot (x ,y , '+r')  % This does nothing. The Figure stays the same
hold off;          % Un-pauses the plot
xlabel('Horizontal Axis') % Lables the x axis on Figure 1021
ylabel('Vertical Axis')   % Lables the y axis on Figure 1021
```

```
a =

     0     0     0     0     0


b =

     1     1
     1     1
     1     1


ans =
```

```
     3

ans =

     4

ans =

     5

ans =

    0.9273

haha , MATLAB is fun

ans =

     9

ans =

  logical

   1

ans =

  logical

   0

ans =

  logical

   1
```


Vertical Axis
Horizontal Axis
8
7.5
7
6.5
6
5.5
5
1
2
3
4
5
6
7


Vertical Axis
Horizontal Axis
8
7
6
5
4
3
2
1
0
1
2
3
4
5
6
7

Exercise 1.2

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

A = abs(a);     % A is magnitude of a
phi = angle(a); % phi is phase angle of a
r1 = (A^(1/n)) * exp(j*phi/n); % find r1
k = [0:1:n-1];  % make vector for each k we need to find each root
r2 = exp(j * 2 * pi * k / n ); % find r2
r = r1 * r2;    % multiply r1 by every value in r2 to get all roots.


help myroots

test1 = myroots(9, 2)
check1 = test1 .^ 9

%I did make this testing code, but there were problems that I
%didn't know how to deal with because I am new to MATLAB. Like
%even if a vector shows all of the values as 2.000, I guess that
%doesn't == 2, or it could have decimals further down than it
%shows. I originally didn't have the real(), and I messed with
%things to make the testing easier but couldn't figure
%it out. So I just run test1 .^ 9 and visually check the results

%check1 = 2 * ones(1,9);
%if(real(test1 .^ 9) == check1)
%    disp('Correct results!')
%else
%    disp('Incorrect results...')
%end

test2 = myroots(23, -j)
check2 = test2 .^ 23

%Same as above comments
%check2 = -1 * ones(1,23);
%if(imag(test2 .^ 23) == check2)
%    disp('Correct results!')
%else
%    disp('Incorrect results...')
%end
```

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


test1 =

  Columns 1 through 4

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

  Columns 5 through 8

  -1.0149 + 0.3694i  -1.0149 - 0.3694i  -0.5400 - 0.9354i   0.1876 -
 1.0637i

  Column 9

   0.8274 - 0.6942i


check1 =

  Columns 1 through 4

   2.0000 + 0.0000i   2.0000 - 0.0000i   2.0000 - 0.0000i   2.0000 -
 0.0000i

  Columns 5 through 8

   2.0000 - 0.0000i   2.0000 - 0.0000i   2.0000 - 0.0000i   2.0000 -
 0.0000i

  Column 9

   2.0000 - 0.0000i


test2 =

  Columns 1 through 4

   0.9977 - 0.0682i   0.9791 + 0.2035i   0.8879 + 0.4601i   0.7308 +
 0.6826i

  Columns 5 through 8

   0.5196 + 0.8544i   0.2698 + 0.9629i   0.0000 + 1.0000i  -0.2698 +
 0.9629i
```

```
  Columns 9 through 12

  -0.5196 + 0.8544i  -0.7308 + 0.6826i  -0.8879 + 0.4601i  -0.9791 +
 0.2035i

  Columns 13 through 16

  -0.9977 - 0.0682i  -0.9423 - 0.3349i  -0.8170 - 0.5767i  -0.6311 -
 0.7757i

  Columns 17 through 20

  -0.3984 - 0.9172i  -0.1362 - 0.9907i   0.1362 - 0.9907i   0.3984 -
 0.9172i

  Columns 21 through 23

   0.6311 - 0.7757i   0.8170 - 0.5767i   0.9423 - 0.3349i


check2 =

  Columns 1 through 4

  -0.0000 - 1.0000i  -0.0000 - 1.0000i   0.0000 - 1.0000i  -0.0000 -
 1.0000i

  Columns 5 through 8

  -0.0000 - 1.0000i   0.0000 - 1.0000i  -0.0000 - 1.0000i   0.0000 -
 1.0000i

  Columns 9 through 12

  -0.0000 - 1.0000i  -0.0000 - 1.0000i   0.0000 - 1.0000i  -0.0000 -
 1.0000i

  Columns 13 through 16

  -0.0000 - 1.0000i   0.0000 - 1.0000i  -0.0000 - 1.0000i  -0.0000 -
 1.0000i

  Columns 17 through 20

  -0.0000 - 1.0000i   0.0000 - 1.0000i  -0.0000 - 1.0000i  -0.0000 -
 1.0000i

  Columns 21 through 23

   0.0000 - 1.0000i   0.0000 - 1.0000i  -0.0000 - 1.0000i
```

Published with MATLAB® R2020a