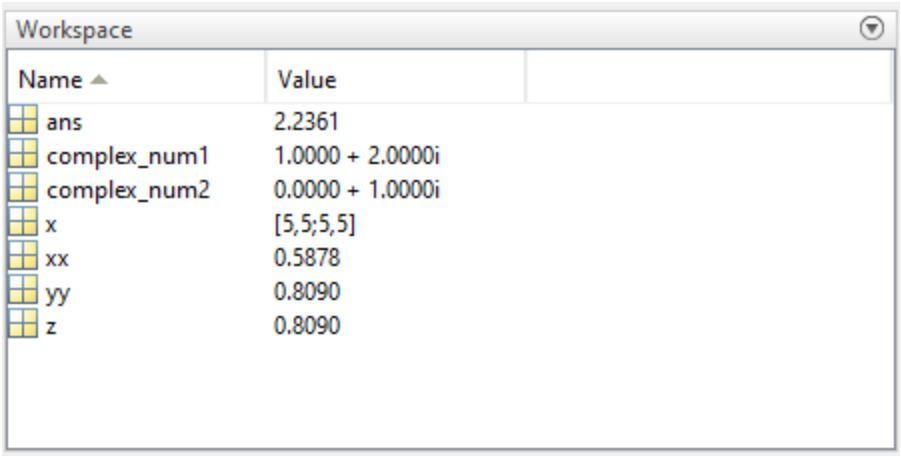
Lab Report

Week 2

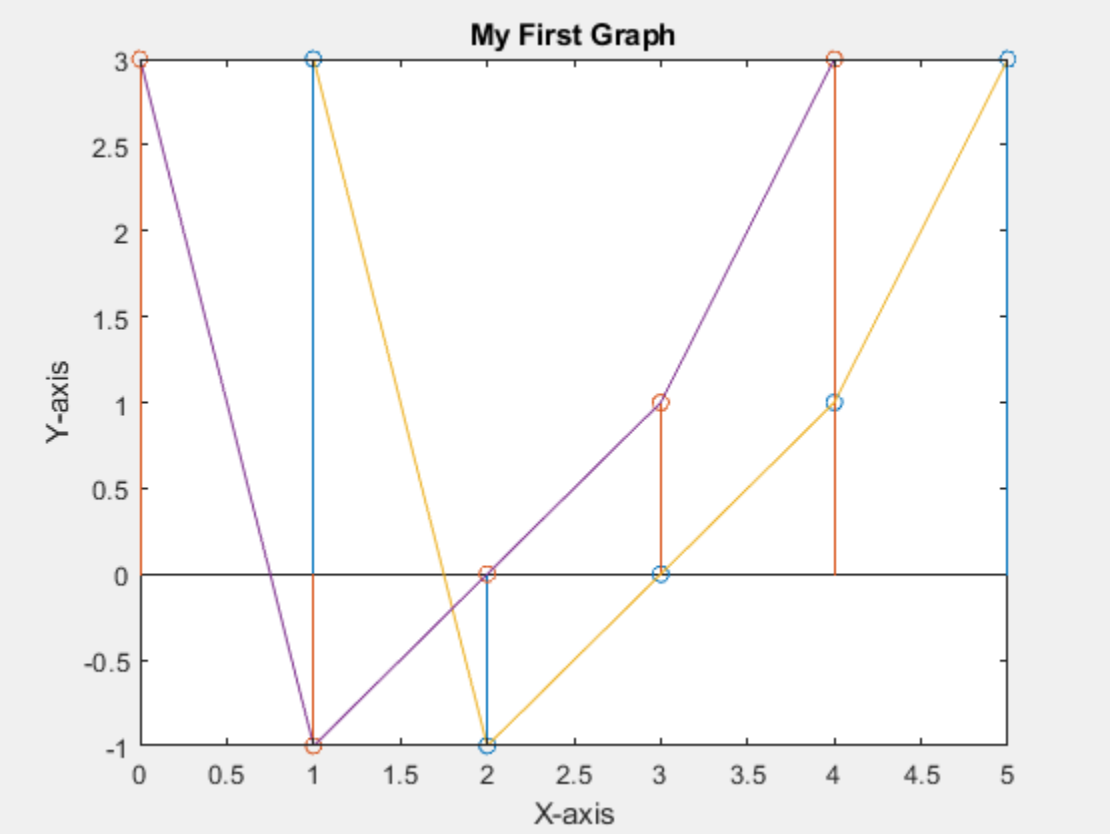
```
>> pi*pi - 10

ans =

   -0.1304

>> sin(pi/4)

ans =

    0.7071

>> ans^2

ans =

    0.5000

% "ans" stands for "answer".

>> abs(-2)

ans =

     2

>> xx = sin(pi/5)

xx =

    0.5878

>> z = cos(pi/5)

z =

    0.8090

>> yy = sqrt(1 - xx*xx)

yy =

    0.8090

>> yy = sqrt(1 - xx^2)

yy =
```

```
    0.8090

>> x = [2,3;4,5] + [3,2;1,0]

x =

     5     5
     5     5

>> complex_num1 = 1 + 2*i

complex_num1 =

   1.0000 + 2.0000i

>> complex_num2 = sqrt(-1)

complex_num2 =

   0.0000 + 1.0000i

>> complex_num1*complex_num2

ans =

  -2.0000 + 1.0000i

>> complex_num1 + complex_num2

ans =

   1.0000 + 3.0000i

>> real(complex_num1)

ans =

     1

>> imag(complex_num1)

ans =

     2

>> abs(complex_num1)

ans =

    2.2361

>> save workspace
>> clear all
>> load('myfirstdatafile.mat')
```




```
>> z = [12,3,-7,5,120]

z =

    12     3    -7     5   120

>> y = [12;3;-7;5;120]

y =

    12
     3
    -7
     5
   120

>> disp(x)
     5     5
     5     5

>> size(z)

ans =

     1     5

>> size(y)

ans =

     5     1

>> x'

ans =

     5     5
     5     5

% Nothing happens because of the values and dimensions of the matrix.
```

```
>> size(x')

ans =

     2     2


% "size(x')" gives you the dimensions of the matrix x transposed. The first
value is the number of rows. The second value is the number of columns. In
this case the two values are equal.

>> x = [3,-1,0,1,3]

x =

     3    -1     0     1     3

>> hold on
>> stem(x)
>> n = [0,1,2,3,4];
>> stem(n,x)
>> plot(x)
>> plot(n,x)
>> title('My First Graph')
>> xlabel('X-axis')
>> ylabel('Y-axis')
```




```
>> n = [0:0.01:4]

n =
```

```
Columns 1 through 5

         0    0.0100    0.0200    0.0300    0.0400

Columns 6 through 10

    0.0500    0.0600    0.0700    0.0800    0.0900

Columns 11 through 15

    0.1000    0.1100    0.1200    0.1300    0.1400

Columns 16 through 20

    0.1500    0.1600    0.1700    0.1800    0.1900

Columns 21 through 25

    0.2000    0.2100    0.2200    0.2300    0.2400

Columns 26 through 30

    0.2500    0.2600    0.2700    0.2800    0.2900

Columns 31 through 35

    0.3000    0.3100    0.3200    0.3300    0.3400

Columns 36 through 40

    0.3500    0.3600    0.3700    0.3800    0.3900

Columns 41 through 45

    0.4000    0.4100    0.4200    0.4300    0.4400

Columns 46 through 50

    0.4500    0.4600    0.4700    0.4800    0.4900

Columns 51 through 55

    0.5000    0.5100    0.5200    0.5300    0.5400

Columns 56 through 60

    0.5500    0.5600    0.5700    0.5800    0.5900

Columns 61 through 65

    0.6000    0.6100    0.6200    0.6300    0.6400

Columns 66 through 70

    0.6500    0.6600    0.6700    0.6800    0.6900
```

```
Columns 71 through 75

  0.7000    0.7100    0.7200    0.7300    0.7400

Columns 76 through 80

  0.7500    0.7600    0.7700    0.7800    0.7900

Columns 81 through 85

  0.8000    0.8100    0.8200    0.8300    0.8400

Columns 86 through 90

  0.8500    0.8600    0.8700    0.8800    0.8900

Columns 91 through 95

  0.9000    0.9100    0.9200    0.9300    0.9400

Columns 96 through 100

  0.9500    0.9600    0.9700    0.9800    0.9900

Columns 101 through 105

  1.0000    1.0100    1.0200    1.0300    1.0400

Columns 106 through 110

  1.0500    1.0600    1.0700    1.0800    1.0900

Columns 111 through 115

  1.1000    1.1100    1.1200    1.1300    1.1400

Columns 116 through 120

  1.1500    1.1600    1.1700    1.1800    1.1900

Columns 121 through 125

  1.2000    1.2100    1.2200    1.2300    1.2400

Columns 126 through 130

  1.2500    1.2600    1.2700    1.2800    1.2900

Columns 131 through 135

  1.3000    1.3100    1.3200    1.3300    1.3400

Columns 136 through 140

  1.3500    1.3600    1.3700    1.3800    1.3900

Columns 141 through 145
```

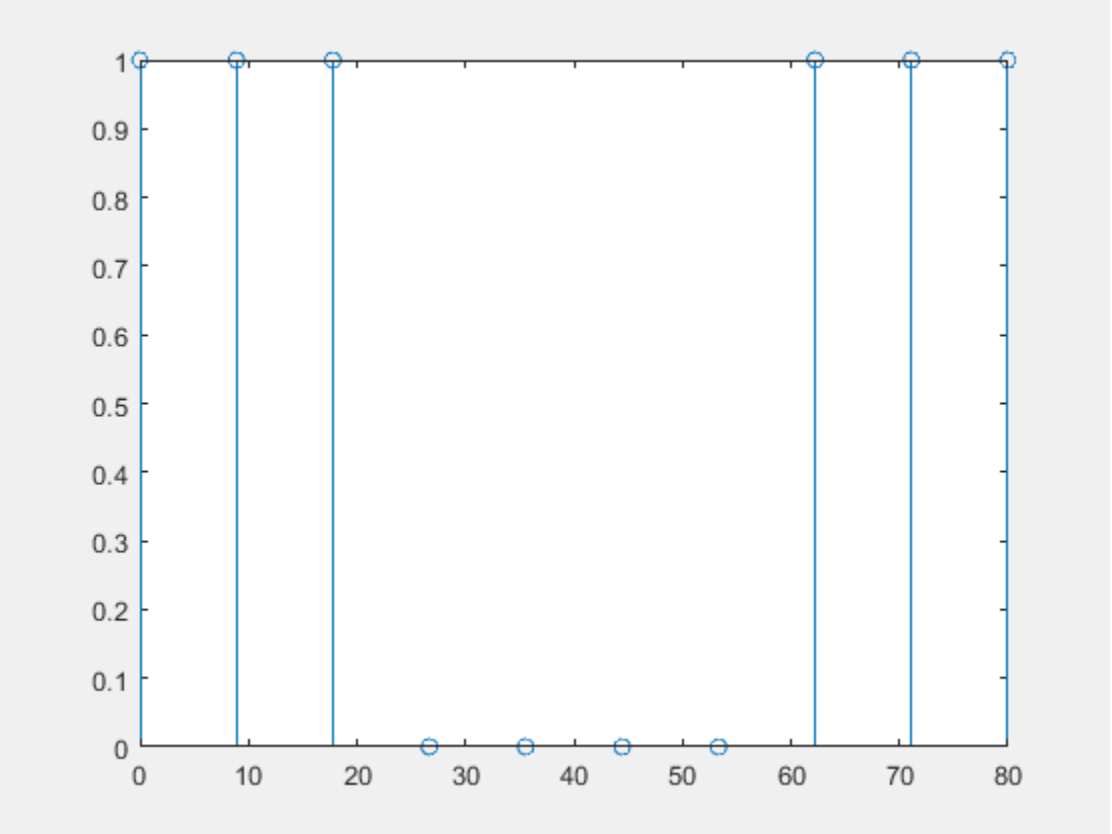
```
    1.4000    1.4100    1.4200    1.4300    1.4400

  Columns 146 through 150

    1.4500    1.4600    1.4700    1.4800    1.4900

  Columns 151 through 155

    1.5000    1.5100    1.5200    1.5300    1.5400

  Columns 156 through 160

    1.5500    1.5600    1.5700    1.5800    1.5900

  Columns 161 through 165

    1.6000    1.6100    1.6200    1.6300    1.6400

  Columns 166 through 170

    1.6500    1.6600    1.6700    1.6800    1.6900

  Columns 171 through 175

    1.7000    1.7100    1.7200    1.7300    1.7400

  Columns 176 through 180

    1.7500    1.7600    1.7700    1.7800    1.7900

  Columns 181 through 185

    1.8000    1.8100    1.8200    1.8300    1.8400

  Columns 186 through 190

    1.8500    1.8600    1.8700    1.8800    1.8900

  Columns 191 through 195

    1.9000    1.9100    1.9200    1.9300    1.9400

  Columns 196 through 200

    1.9500    1.9600    1.9700    1.9800    1.9900

  Columns 201 through 205

    2.0000    2.0100    2.0200    2.0300    2.0400

  Columns 206 through 210

    2.0500    2.0600    2.0700    2.0800    2.0900

  Columns 211 through 215
```

```
    2.1000    2.1100    2.1200    2.1300    2.1400

  Columns 216 through 220

    2.1500    2.1600    2.1700    2.1800    2.1900

  Columns 221 through 225

    2.2000    2.2100    2.2200    2.2300    2.2400

  Columns 226 through 230

    2.2500    2.2600    2.2700    2.2800    2.2900

  Columns 231 through 235

    2.3000    2.3100    2.3200    2.3300    2.3400

  Columns 236 through 240

    2.3500    2.3600    2.3700    2.3800    2.3900

  Columns 241 through 245

    2.4000    2.4100    2.4200    2.4300    2.4400

  Columns 246 through 250

    2.4500    2.4600    2.4700    2.4800    2.4900

  Columns 251 through 255

    2.5000    2.5100    2.5200    2.5300    2.5400

  Columns 256 through 260

    2.5500    2.5600    2.5700    2.5800    2.5900

  Columns 261 through 265

    2.6000    2.6100    2.6200    2.6300    2.6400

  Columns 266 through 270

    2.6500    2.6600    2.6700    2.6800    2.6900

  Columns 271 through 275

    2.7000    2.7100    2.7200    2.7300    2.7400

  Columns 276 through 280

    2.7500    2.7600    2.7700    2.7800    2.7900

  Columns 281 through 285

    2.8000    2.8100    2.8200    2.8300    2.8400
```

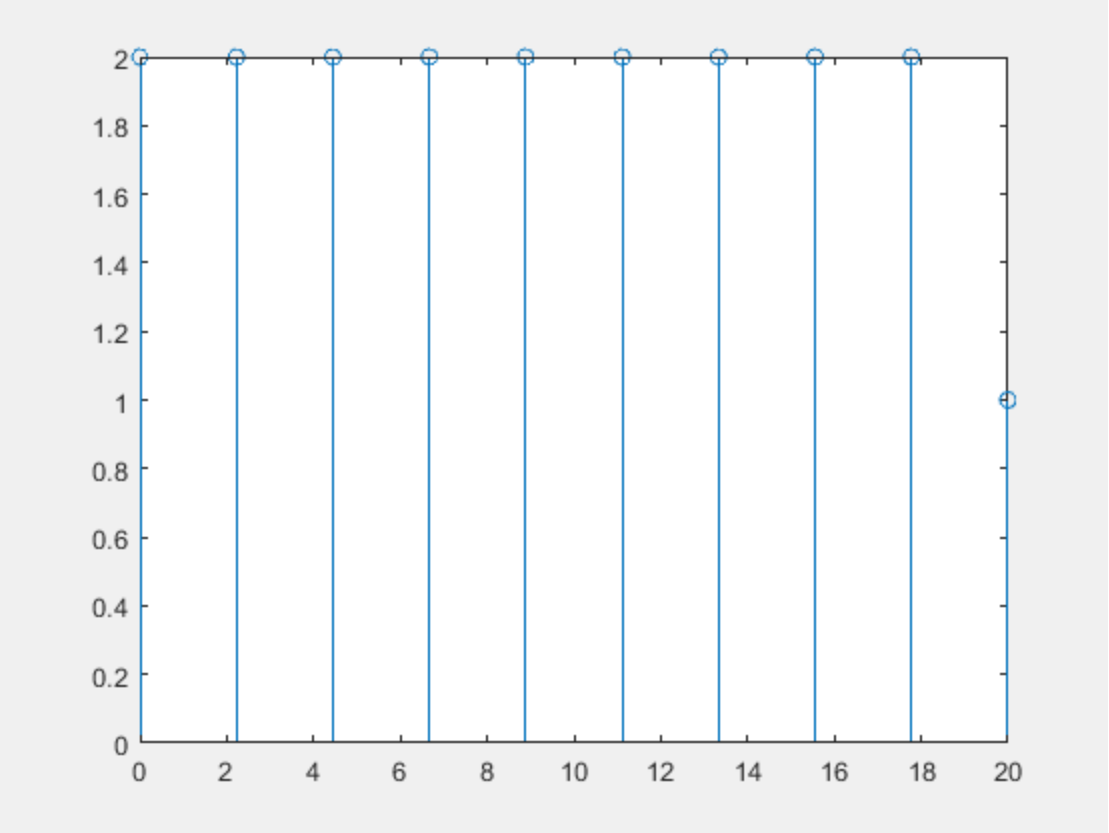
```
Columns 286 through 290

  2.8500    2.8600    2.8700    2.8800    2.8900

Columns 291 through 295

  2.9000    2.9100    2.9200    2.9300    2.9400

Columns 296 through 300

  2.9500    2.9600    2.9700    2.9800    2.9900

Columns 301 through 305

  3.0000    3.0100    3.0200    3.0300    3.0400

Columns 306 through 310

  3.0500    3.0600    3.0700    3.0800    3.0900

Columns 311 through 315

  3.1000    3.1100    3.1200    3.1300    3.1400

Columns 316 through 320

  3.1500    3.1600    3.1700    3.1800    3.1900

Columns 321 through 325

  3.2000    3.2100    3.2200    3.2300    3.2400

Columns 326 through 330

  3.2500    3.2600    3.2700    3.2800    3.2900

Columns 331 through 335

  3.3000    3.3100    3.3200    3.3300    3.3400

Columns 336 through 340

  3.3500    3.3600    3.3700    3.3800    3.3900

Columns 341 through 345

  3.4000    3.4100    3.4200    3.4300    3.4400

Columns 346 through 350

  3.4500    3.4600    3.4700    3.4800    3.4900

Columns 351 through 355

  3.5000    3.5100    3.5200    3.5300    3.5400
```

```
  Columns 356 through 360

    3.5500    3.5600    3.5700    3.5800    3.5900

  Columns 361 through 365

    3.6000    3.6100    3.6200    3.6300    3.6400

  Columns 366 through 370

    3.6500    3.6600    3.6700    3.6800    3.6900

  Columns 371 through 375

    3.7000    3.7100    3.7200    3.7300    3.7400

  Columns 376 through 380

    3.7500    3.7600    3.7700    3.7800    3.7900

  Columns 381 through 385

    3.8000    3.8100    3.8200    3.8300    3.8400

  Columns 386 through 390

    3.8500    3.8600    3.8700    3.8800    3.8900

  Columns 391 through 395

    3.9000    3.9100    3.9200    3.9300    3.9400

  Columns 396 through 400

    3.9500    3.9600    3.9700    3.9800    3.9900

  Column 401

    4.0000

>> n = linspace(-5,5,10)

n =

  Columns 1 through 5

   -5.0000   -3.8889   -2.7778   -1.6667   -0.5556

  Columns 6 through 10

    0.5556    1.6667    2.7778    3.8889    5.0000

>> n = [0,1,2,3,4]

n =

     0     1     2     3     4
```

```
>> n = [0:4]

n =

     0     1     2     3     4

>> T = 10;
>> D = 5;
>> N = 8;
>> n = linspace(0,T*N,10);
>> d = [0:T:T*N];
>> y = pulstran(n,d,'rectpuls',D);
>> stem(n,y)
```




```
>> T = 5;
>> D = 10;
>> N = 4;
>> n = linspace(0,T*N,10);
>> d = [0:T:T*N];
>> y = pulstran(n,d,'rectpuls',D);
>> stem(n,y)
```



Week 3

```
>> xk = cos(pi*(0:11)/4)

xk =

  Columns 1 through 5

    1.0000    0.7071    0.0000   -0.7071   -1.0000

  Columns 6 through 10

   -0.7071   -0.0000    0.7071    1.0000    0.7071

  Columns 11 through 12

    0.0000   -0.7071
```

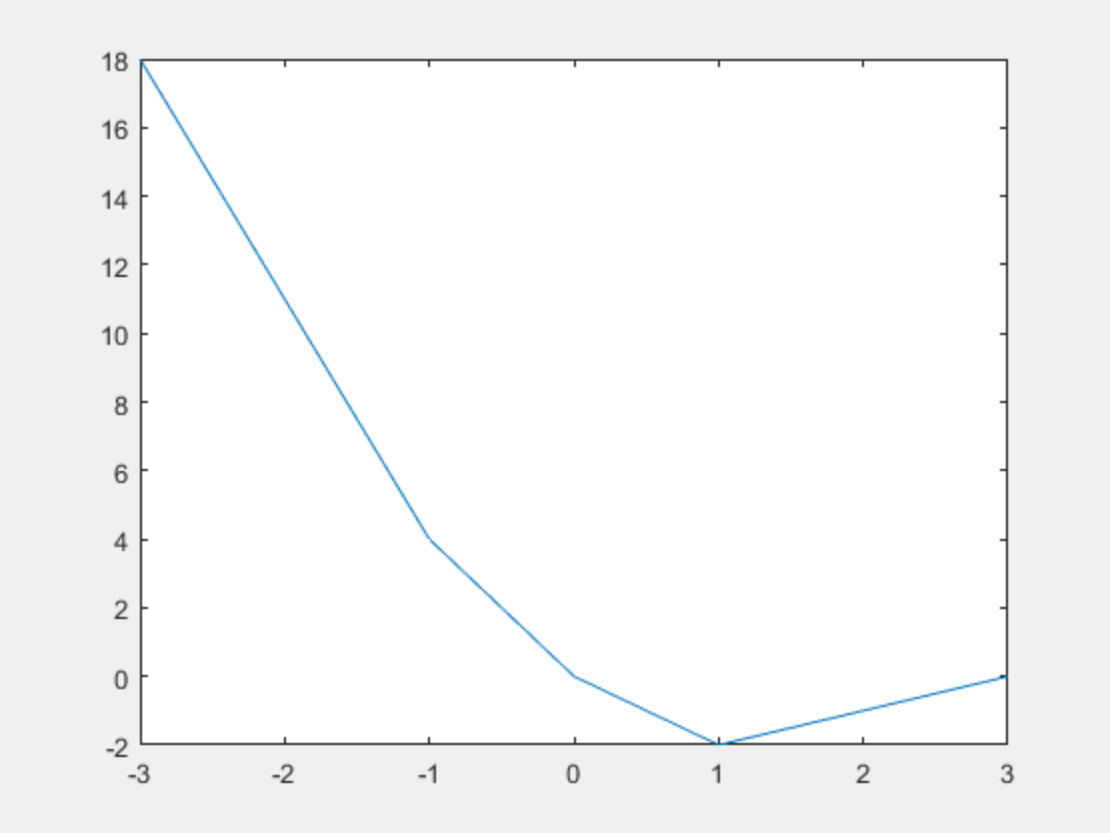
```
% Each value of the cosine function is stored across various columns in the
vector xk. The columns are labelled starting from 1. xk(1) is 1.0000. xk(0)
is not defined.
```

```
>> yy = cos((-5:5)*pi/3)

yy =

  Columns 1 through 5
```

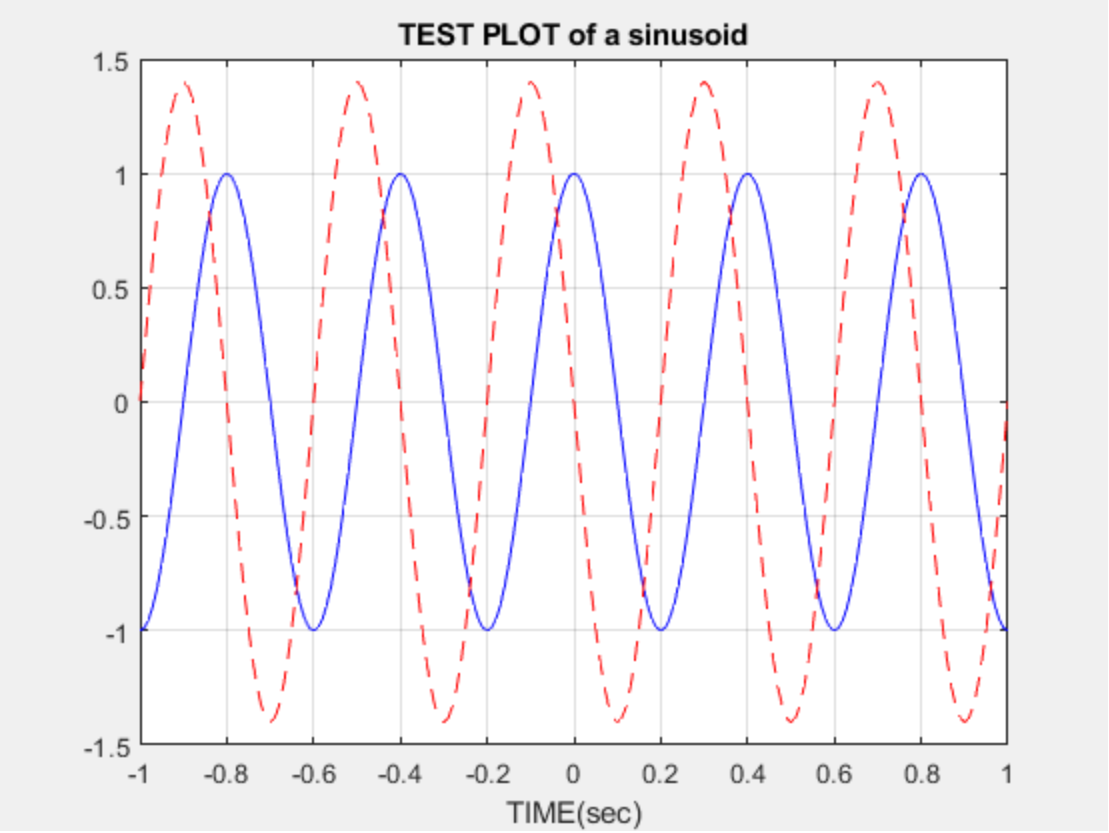
```
    0.5000   -0.5000   -1.0000   -0.5000    0.5000

  Columns 6 through 10

    1.0000    0.5000   -0.5000   -1.0000   -0.5000

  Column 11

    0.5000
```

```
% It is necessary to use yy(k + 6) in the loop because the indices in the
vector must start from 1. If yy(k) is used, then the first number will be
stored in an invalid index.
```

```
>> x = [-3,-1,0,1,3];
>> y = x.*x - 3*x;
>> plot(x,y)
>> z = x + y*sqrt(-1)

z =

  Columns 1 through 2

  -3.0000 +18.0000i  -1.0000 + 4.0000i

  Columns 3 through 4

   0.0000 + 0.0000i   1.0000 - 2.0000i

  Column 5

   3.0000 + 0.0000i

>> plot(z)
```

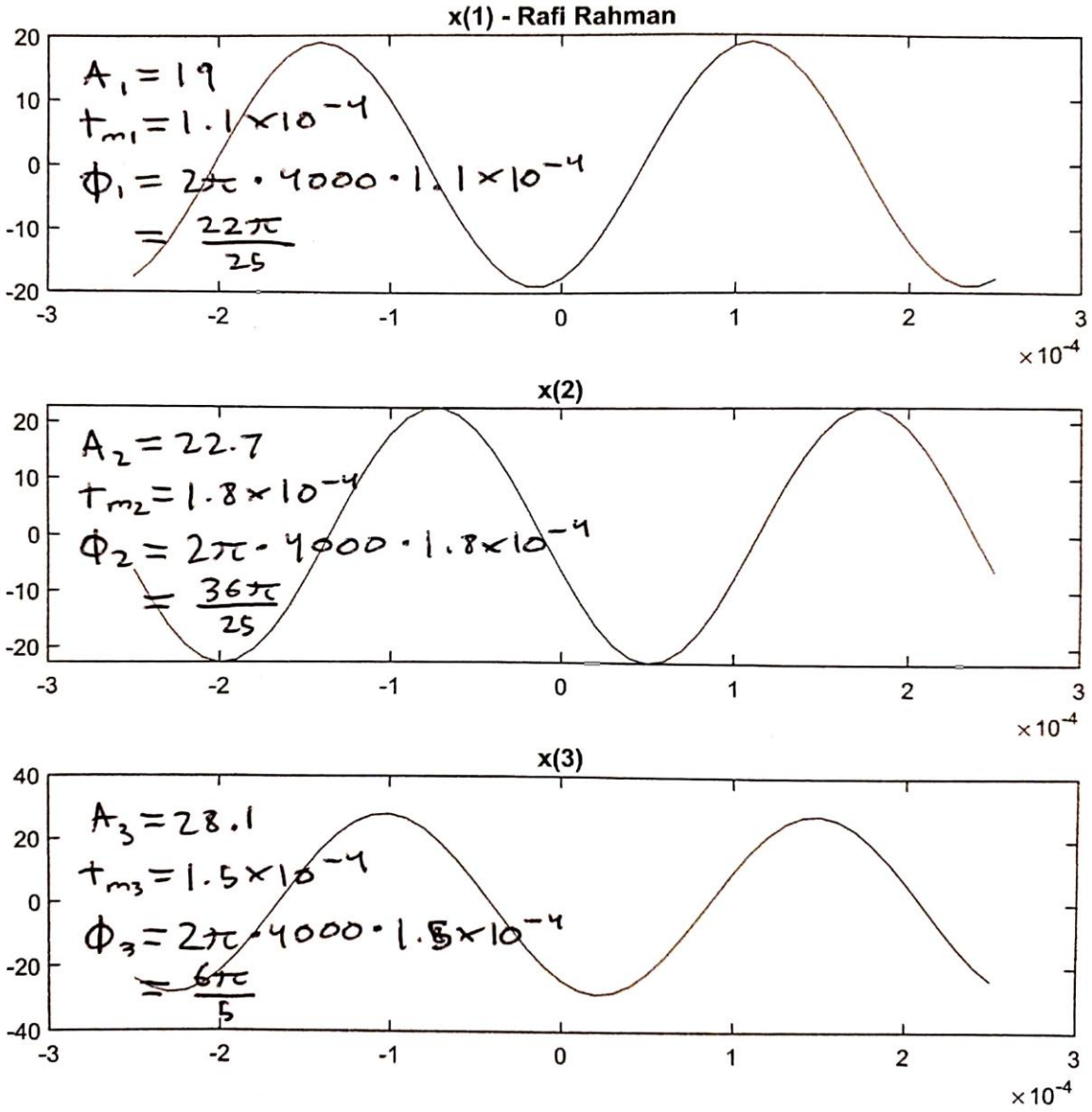



```
% xx.*xx gives us the dot product of xx and itself. Dot product works by
multiplying corresponding elements of two vectors of the same length and
summing the products, unlike matrix multiplication, which multiplies the
elements of two matrices and gives us a matrix as the result.

tt = -1:0.01:1;
xx = cos(5*pi*tt);
zz = 1.4*exp(j*pi/2)*exp(j*5*pi*tt);
plot(tt,xx,'b-',tt,real(zz),'r--'), grid on
title('TEST PLOT of a sinusoid')
xlabel('TIME(sec)')

% The plot of real(zz) is a sinusoid thanks to Euler's Identity. The
amplitude of the wave is 1.4 and the phase is pi/2.

>> mylab1
```



Individual Work

```
t = 1/4000;
tt = -t:t/25:t;
a1 = 19;
a2 = 1.2*a1;
tm1 = (37.2/5)*t;
tm2 = -(41.3/18)*t;
x1 = a1*cos(2*pi*4000*(tt - tm1));
subplot(3,1,1)
plot(tt,x1)
title('x(1) - Rafi Rahman')
x2 = a2*cos(2*pi*4000*(tt - tm2));
subplot(3,1,2)
plot(tt,x2)
title('x(2)')
x3 = x1 + x2;
subplot(3,1,3)
plot(tt,x3)
title('x(3)')
```



Week 4

```
>> r = roots([1,4,3]);
>> disp(['Case (k = 3): roots = [',num2str(r.'),']']);
Case (k = 3): roots = [-3 -1]
>> r = roots([1,4,4]);
>> disp(['Case k = 4: roots = [',num2str(r.'),']']);
Case k = 4: roots = [-2 -2]
>> r = roots([1,4,40]);
>> disp(['Case (k = 40): roots = [',num2str(r.',' %0.5g'),']']);
Case (k = 40): roots = [-2 +6i -2 -6i]
>> y_0 = dsolve('D2y + 4*Dy + 3*y = 0','y(0) = 3','Dy(0) = -7','t');
>> disp(['(a) k = 3: y_0 = ',char(y_0)])
(a) k = 3: y_0 = exp(-t) + 2*exp(-3*t)
>> y_0 = dsolve('D2y + 4*Dy + 4*y = 0','y(0) = 3','Dy(0) = -7','t');
>> disp(['(b) k = 4: y_0 = ',char(y_0)])
```

```
(b) k = 4: y_0 = 3*exp(-2*t) - t*exp(-2*t)
>> y_0 = dsolve('D2y + 4*Dy + 40*y = 0','y(0) = 3','Dy(0) = -7','t');
>> disp(['(c) k = 40; y_0 = ',char(y_0)])
(c) k = 40; y_0 = 3*cos(6*t)*exp(-2*t) - (sin(6*t)*exp(-2*t))/6
>> y = dsolve('D2y + 3*Dy + 2*y = 5*t + 3','y(0) = 2','Dy(0) = 3','t');
>> disp(['y(t) = (',char(y),')u(t) ']);
y(t) = ((5*t)/2 + 9*exp(-t) - (19*exp(-2*t))/4 - 9/4)u(t)
```

```
R = [1e4,1e4,1e4];
C = [1e-6,1e-6];
a0 = 1;
a1 = (1/R(1) + 1/R(2) + 1/R(3))/C(2);
a2 = 1/R(1)*R(2)*C(1)*C(2);
A = [a0,a1,a2];
lambda = roots(A);
```

```
>> mylab3
```

Name	Value
A	[1,300.0000,1.0000e-12]
a0	1
a1	300.0000
a2	1.0000e-12
C	[1.0000e-06,1.0000e-06]
lambda	[-300.0000;-3.3333e-15]
R	[10000,10000,10000]

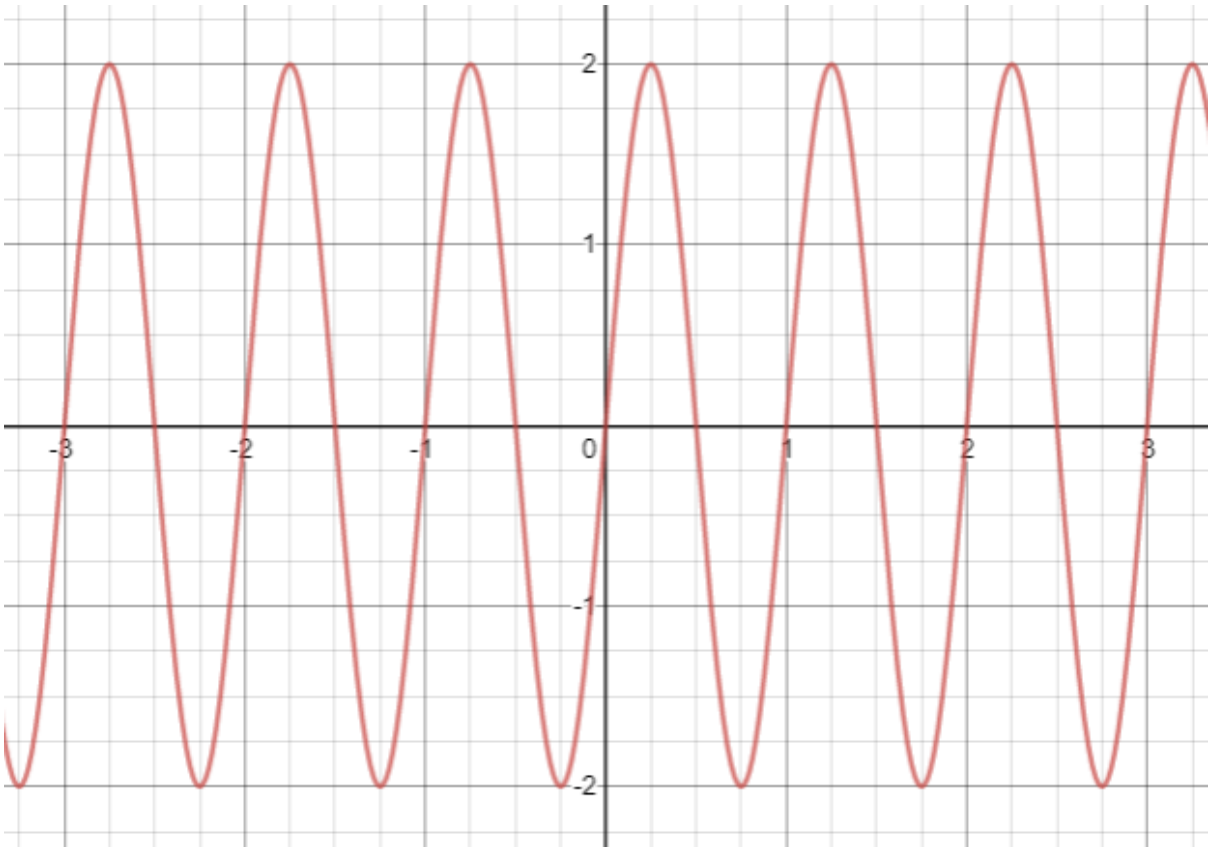
```
R = [1e4,1e4,1e4];
C = [1e-9,1e-6];
a0 = 1;
a1 = (1/R(1) + 1/R(2) + 1/R(3))/C(2);
a2 = 1/R(1)*R(2)*C(1)*C(2);
A = [a0,a1,a2];
lambda = roots(A);
```

```
>> mylab3
```

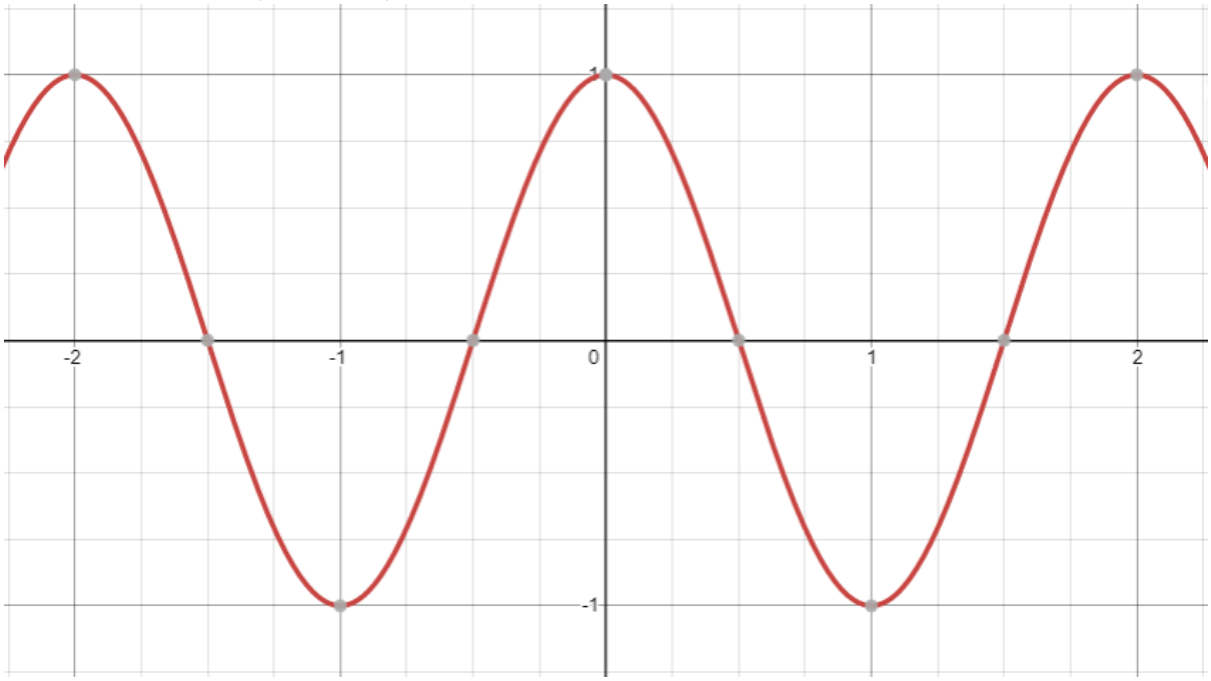
Name	Value
A	[1,300.0000,1.0000e-15]
a0	1
a1	300.0000
a2	1.0000e-15
C	[1.0000e-09,1.0000e-06]
lambda	[-300.0000;-3.3333e-18]
R	[10000,10000,10000]

Lab Assessable Exercises

1. Non-causal, Periodic (1 second), Odd



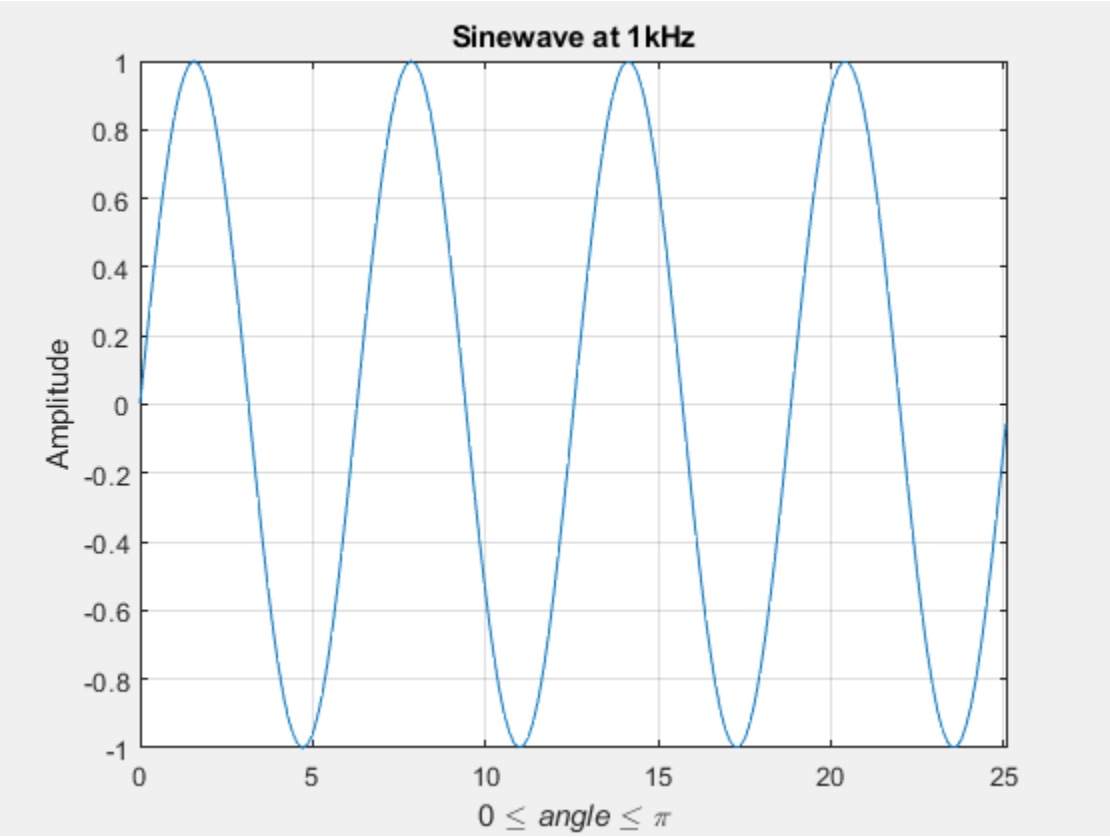
2. Non-causal, Periodic (2 seconds), Even



3. The system is linear because it follows the rules of additivity, homogeneity, and superposition, due to the equal sign that relates the output *y* to input *x*. It is causal because it only depends on present and past inputs. It is a time-invariant system as the output is not affected by changes in time.

Individual Work

```
function [t,sinewave] = sinegen(fsig,fsamp,ncycle)
    delta_angle = 2*pi*fsig/fsamp;
    t = 0:delta_angle:4*2*pi;
    sinewave = sin(t);

function [noise] = noisegen(rms,nsamp)
    noise = rms*randn(nsamp);

fs = 44100;
f = 1000;
rms = 0.1;
[t,sinewave] = sinegen(f,fs,4);
plot(t,sinewave);
grid on;
axis([0,8*pi,-1,1]);
xlabel('0 \leq \itangle \leq \pi');
ylabel('Amplitude');
title('Sinewave at 1kHz');
noise = noisegen(rms,fs);
hold on;
plot(noise);
```




```
Error using randn
Requested 44100x44100 (14.5GB) array exceeds maximum array size preference.
Creation of arrays greater than
this limit may take a long time and cause MATLAB to become unresponsive.
See array size limit or preference
panel for more information.
```

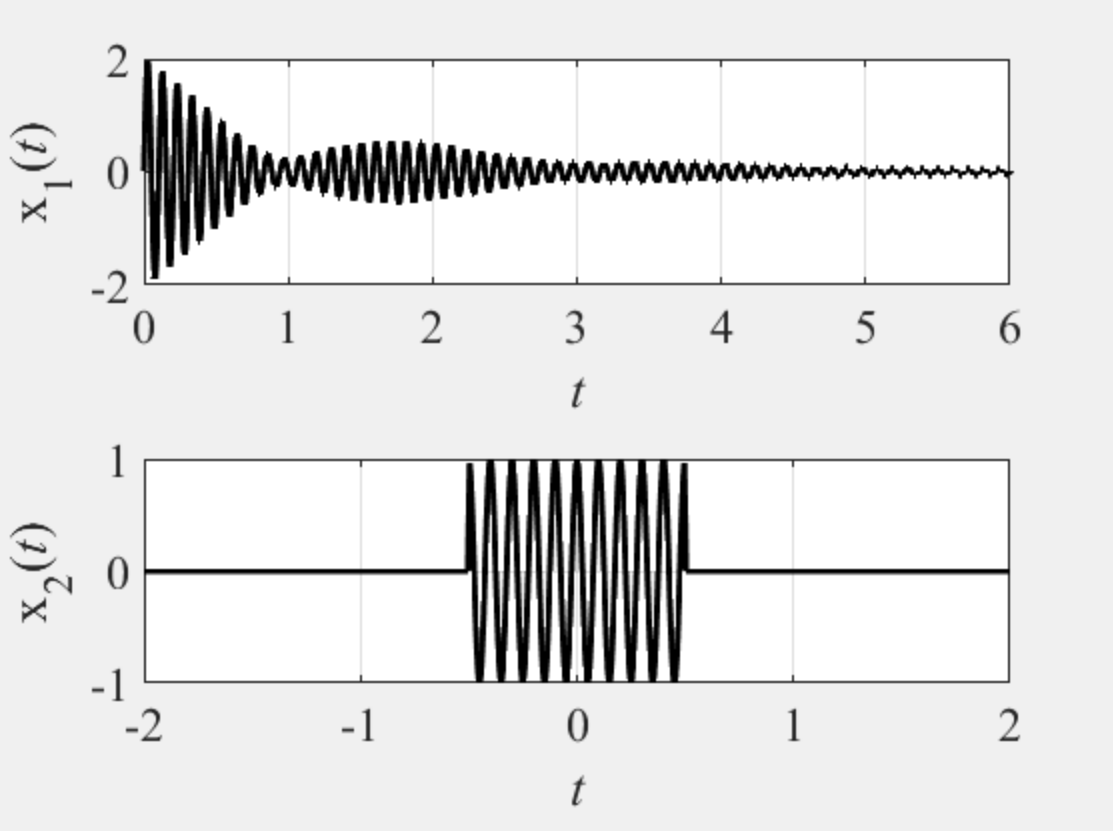
Week 5

```
function c = hyp(a,b)
    c = sqrt(a^2 + b^2);

>> hyp(3,4)

ans =

     5

function y = us(x)
    y = (sign(x) + 1)/2;

>> us(1)

ans =

     1

>> us(0)

ans =

    0.5000

>> us(-1)

ans =

     0

function y = ramp(x)
    y = x.*us(x);

>> ramp(0)

ans =

     0

>> ramp(0.5)

ans =

    0.5000

>> ramp(1)

ans =

     1

function y = rect(x)
    y = us(x + 0.5) - us(x - 0.5);

>> rect(-1)
```

```
ans =

     0

>> rect(-0.5)

ans =

    0.5000

>> rect(0)

ans =

     1

>> rect(0.5)

ans =

    0.5000

>> rect(1)

ans =

     0
```

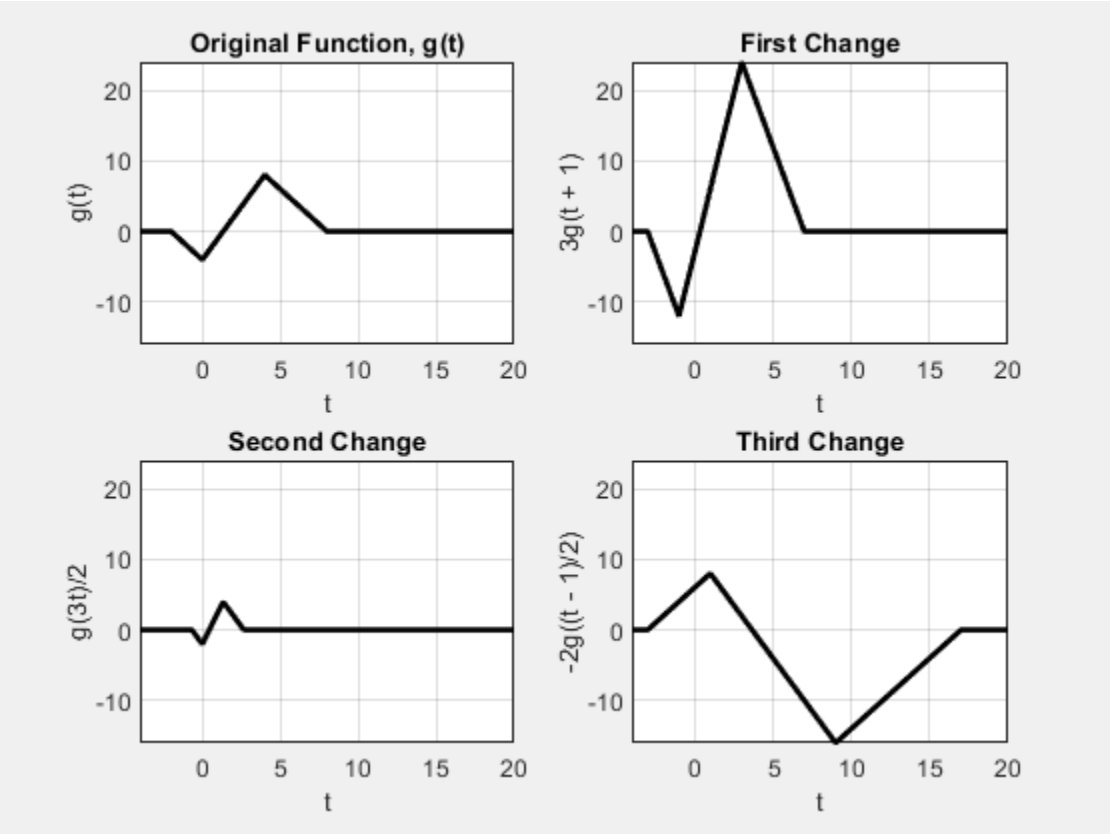
```
t = 0:1/240:6;
x1 = exp(-t).*sin(20*pi*t) + exp(-t/2).*sin(19*pi*t);
subplot(2,1,1);
p = plot(t,x1,'k');
set(p,'LineWidth',2);
xlabel('\itt','FontName','Times','FontSize',24);
ylabel('x_1({\itt})','FontName','Times','FontSize',24);
set(gca,'FontName','Times','FontSize',18);
grid on;
t = -2:1/240:2;
x2 = rect(t).*cos(20*pi*t);
subplot(2,1,2);
p = plot(t,x2,'k');
set(p,'LineWidth',2);
xlabel('\itt','FontName','Times','FontSize',24);
ylabel('x_2({\itt})','FontName','Times','FontSize',24);
set(gca,'FontName','Times','FontSize',18);
grid on;
```

```
>> mylab4
```




```
tmin = -4; tmax = 20;
dt = 0.1;
t = tmin:dt:tmax;
g0 = g(t);
g1 = 3*g(t + 1);
g2 = g(3*t)/2;
g3 = -2*g((t - 1)/2);
gmax = max([max(g0),max(g1),max(g2),max(g3)]);
gmin = min([mix(g0),min(g1),min(g2),min(g3)]);
subplot(2,2,1);
p = plot(t,g0,'k');
set(p,'LineWidth',2);
xlabel('t');
ylabel('g(t)');
title('Original Function, g(t)');
axis([tmin,tmax,gmin,gmax]);
grid;
subplot(2,2,2);
p = plot(t,g1,'k');
set(p,'LineWidth',2);
xlabel('t');
ylabel('3g(t + 1)');
title('First Change');
axis([tmin,tmax,gmin,gmax]);
grid;
subplot(2,2,3);
p = plot(t,g2,'k');
set(p,'LineWidth',2);
xlabel('t');
ylabel('g(3t)/2');
title('Second Change');
axis([tmin,tmax,gmin,gmax]);
```

```
grid;
subplot(2,2,4);
p = plot(t,g3,'k');
set(p,'LineWidth',2);
xlabel('t');
ylabel('-2g((t - 1)/2)');
title('Third Change');
axis([tmin,tmax,gmin,gmax]);
grid;

>> mylab5
```

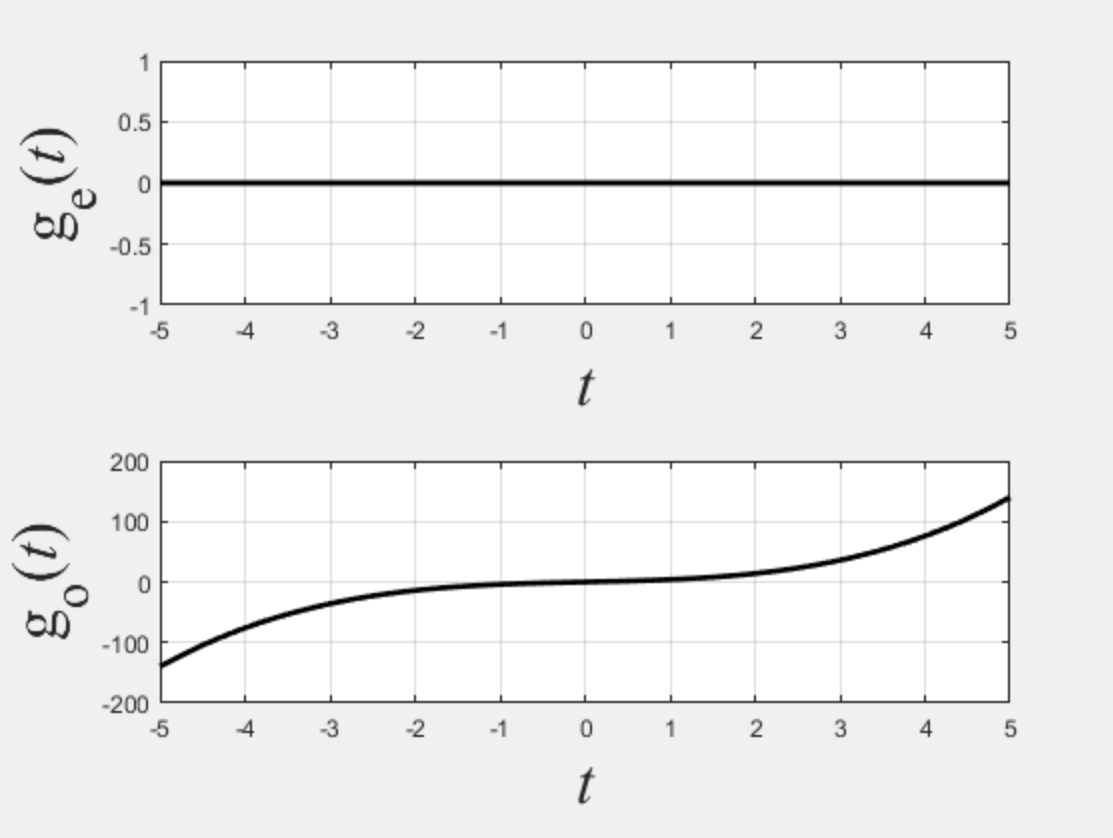



```
>> x = sym('x');
>> diff(sin(x^2))

ans =

2*x*cos(x^2)

>> sym('x');
>> int(1/(1 + x^2))

ans =

atan(x)

>> cumsum(1:5)

ans =

     1     3     6    10    15
```

```
>> dx = pi/16;
>> x = 0:dx:pi/4;
>> y = sin(x)

y =

         0    0.1951    0.3827    0.5556    0.7071

>> cumsum(y)*dx

ans =

         0    0.0383    0.1134    0.2225    0.3614
```

```
function GraphEvenAndOdd
    t = -5:0.1:5;
    ge = (g(t) + g(-t))/2;
    go = (g(t) - g(-t))/2;
    subplot(2,1,1);
    ptr = plot(t,ge,'k');
    set(ptr,'LineWidth',2);
    grid on;
    xlabel('\itt','FontName','Times','FontSize',24);
    ylabel('g_e({\itt})','FontName','Times','FontSize',24);
    subplot(2,1,2);
    ptr = plot(t,go,'k');
    set(ptr,'LineWidth',2);
    grid on;
    xlabel('\itt','FontName','Times','FontSize',24);
    ylabel('g_o({\itt})','FontName','Times','FontSize',24);

function y = g(x)
    y = x.*(x.^2 + 3);
```

```
>> GraphEvenAndOdd
```

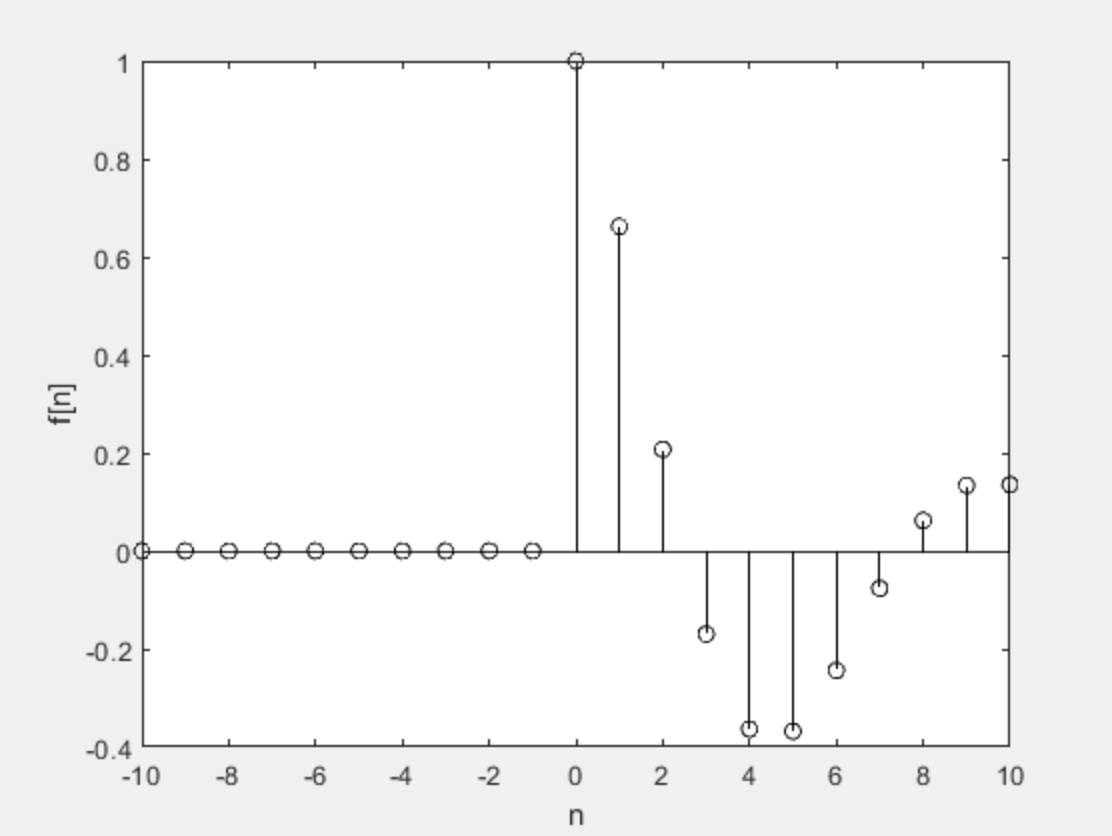



```
dt = 0.1;
t = -7:dt:13;
x = 4*exp(-t/10).*rect((t - 4)/3);
xsq = x.^2;
Ex = trapz(t,xsq);
disp(['(a) Ex = ',num2str(Ex)]);
T0 = 10;
dt = 0.1;
t = -5:dt:5;
x = -3*t;
xsq = x.^2;
Px = trapz(t,xsq)/T0;
disp(['(b) Px = ',num2str(Px)]);
```

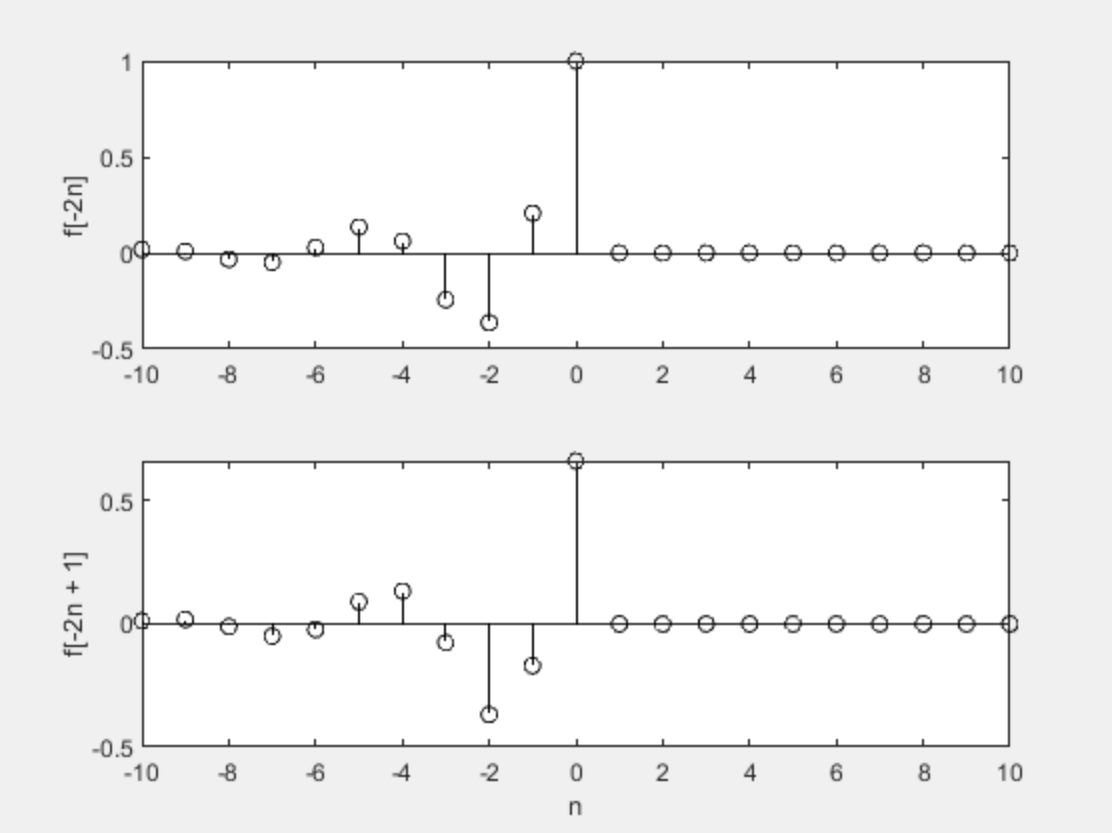
```
>> mylab6
(a) Ex = 21.5177
(b) Px = 75.015
```

Week 6

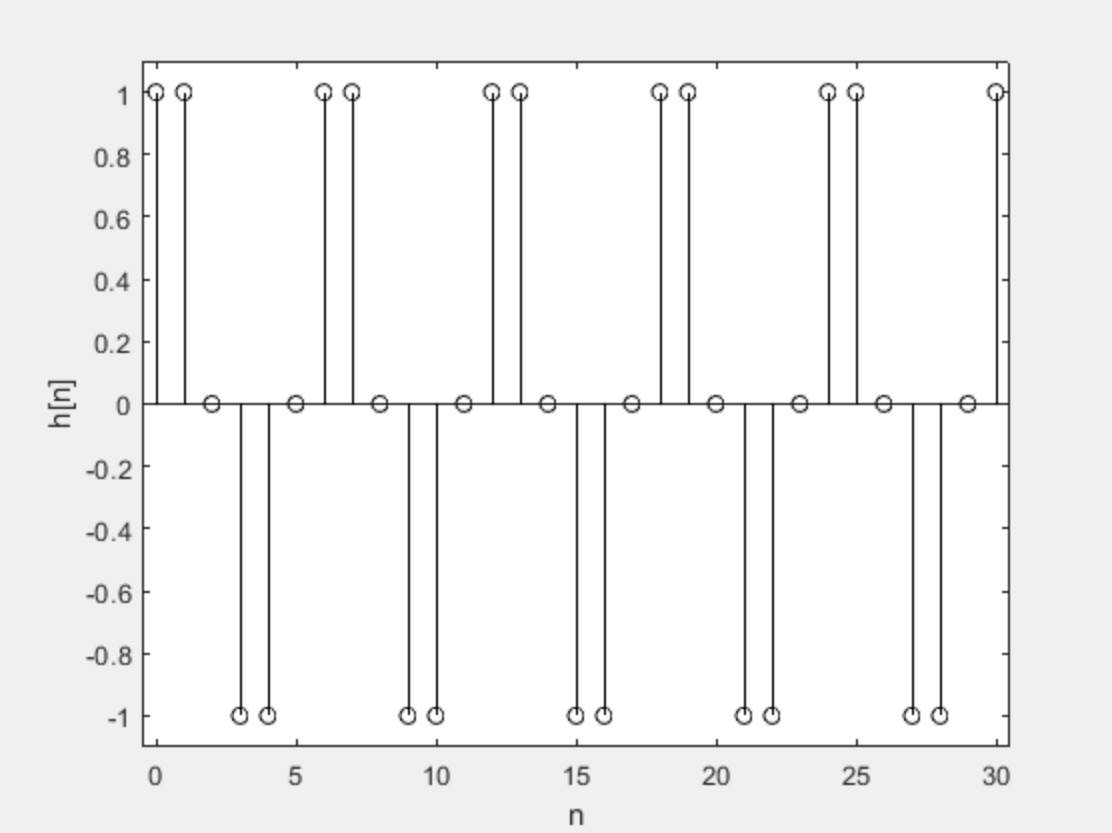
```
>> f = inline('exp(-n/5).*cos(pi*n/5).*(n>=0)','n');
>> n = (-10:10)';
>> stem(n,f(n),'k');
>> xlabel('n');
>> ylabel('f[n]');
```



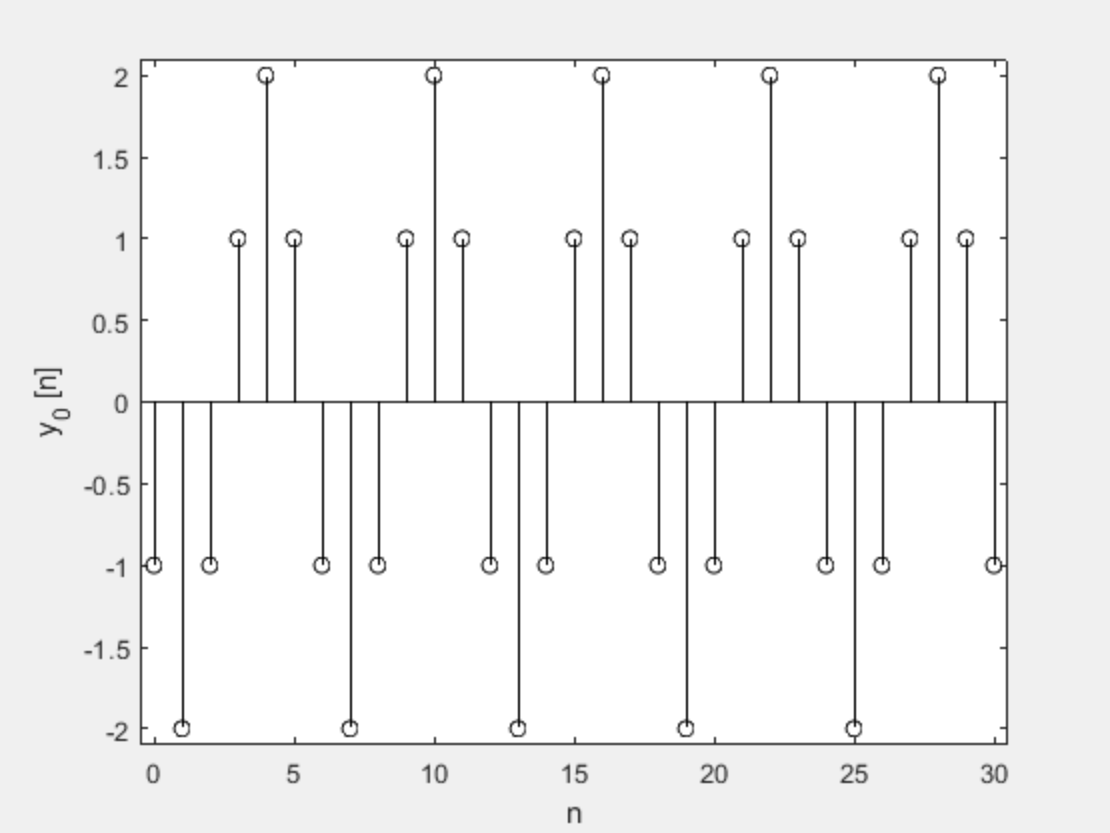

```
>> subplot(2,1,1);
>> stem(n,f(-2*n),'k');
>> ylabel('f[-2n]');
>> subplot(2,1,2);
>> stem(n,f(-2*n + 1),'k');
>> ylabel('f[-2n + 1]');
>> xlabel('n');
```




```
>> b = [1,0,0];
>> a = [1,-1,1];
>> n = (0:30)';
>> delta = inline('n==0','n');
>> h = filter(b,a,delta(n));
>> stem(n,h,'k');
>> axis([-0.5,30.5,-1.1,1.1]);
>> xlabel('n');
>> ylabel('h[n]');
```

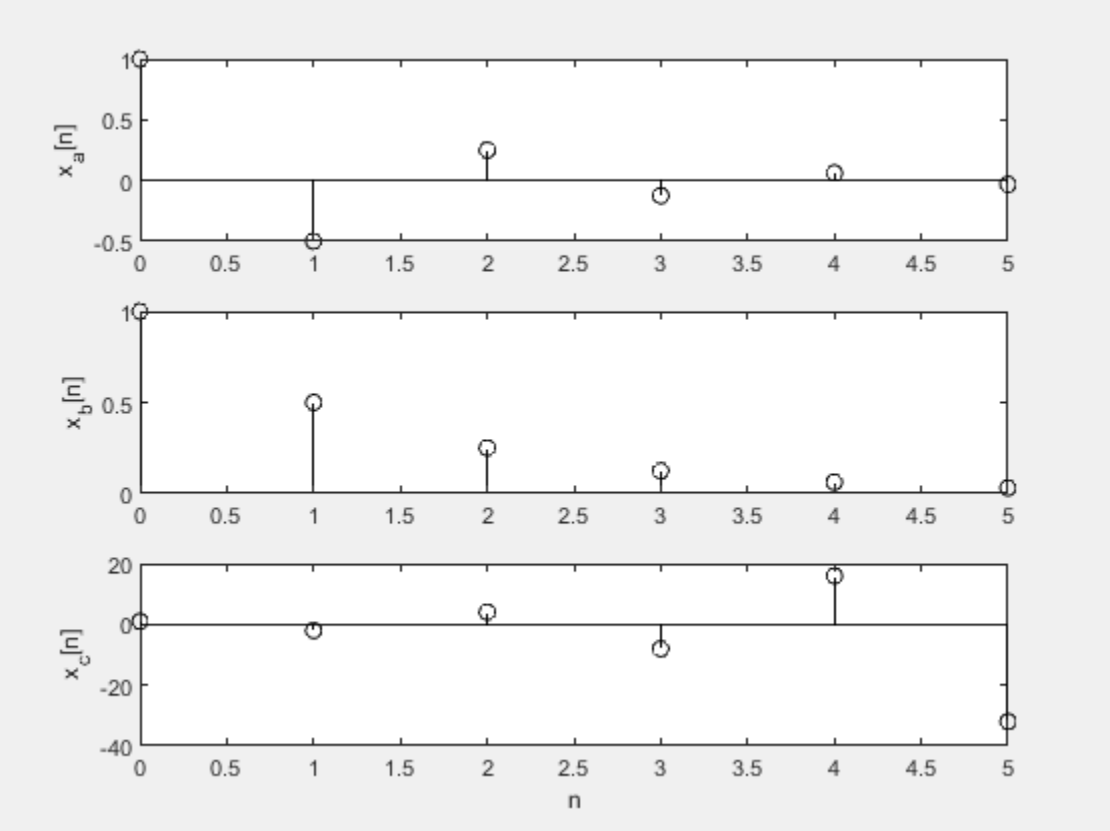



```
>> z_i = filtic(b,a,[1,2]);
>> y_0 = filter(b,a,zeros(size(n)),z_i);
>> stem(n,y_0,'k');
>> xlabel('n');
>> ylabel('y_{0} [n]');
>> axis([-0.5,30.5,-2.1,2.1]);
```

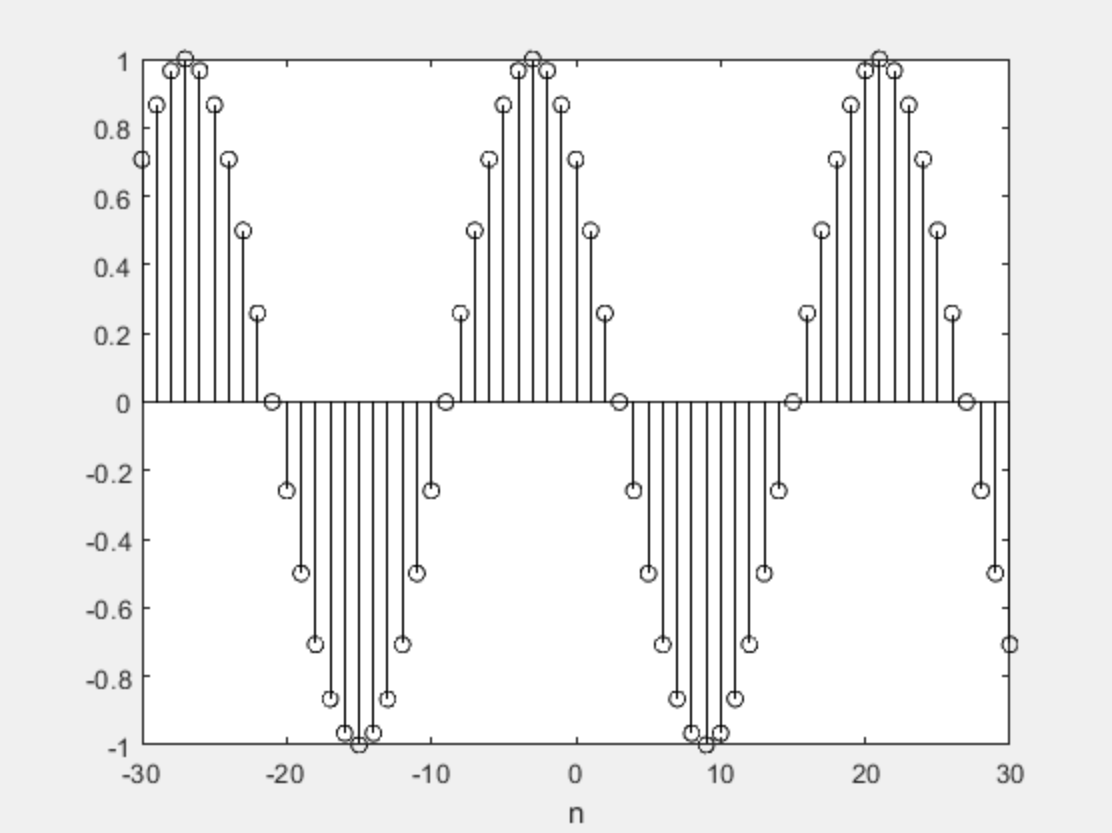



```
>> y_total = filter(b,a,x(n),z_i);
>> sum(abs(y_total - (y + y_0)))

ans =

   1.8430e-14

>> conv([1,1,1,1],[1,1,1,1])

ans =

     1     2     3     4     3     2     1

>> conv([1,1,1,1],[1,1,1,1])

ans =

     1     2     3     4     3     2     1

>> n = (0:5);
>> x_a = (-0.5).^n;
>> x_b = 2.^(-n);
>> x_c = (-2).^n;
>> subplot(3,1,1);
>> stem(n,x_a,'k');
>> ylabel('x_a[n]');
>> subplot(3,1,2);
>> stem(n,x_b,'k');
>> ylabel('x_b[n]');
>> subplot(3,1,3);
```

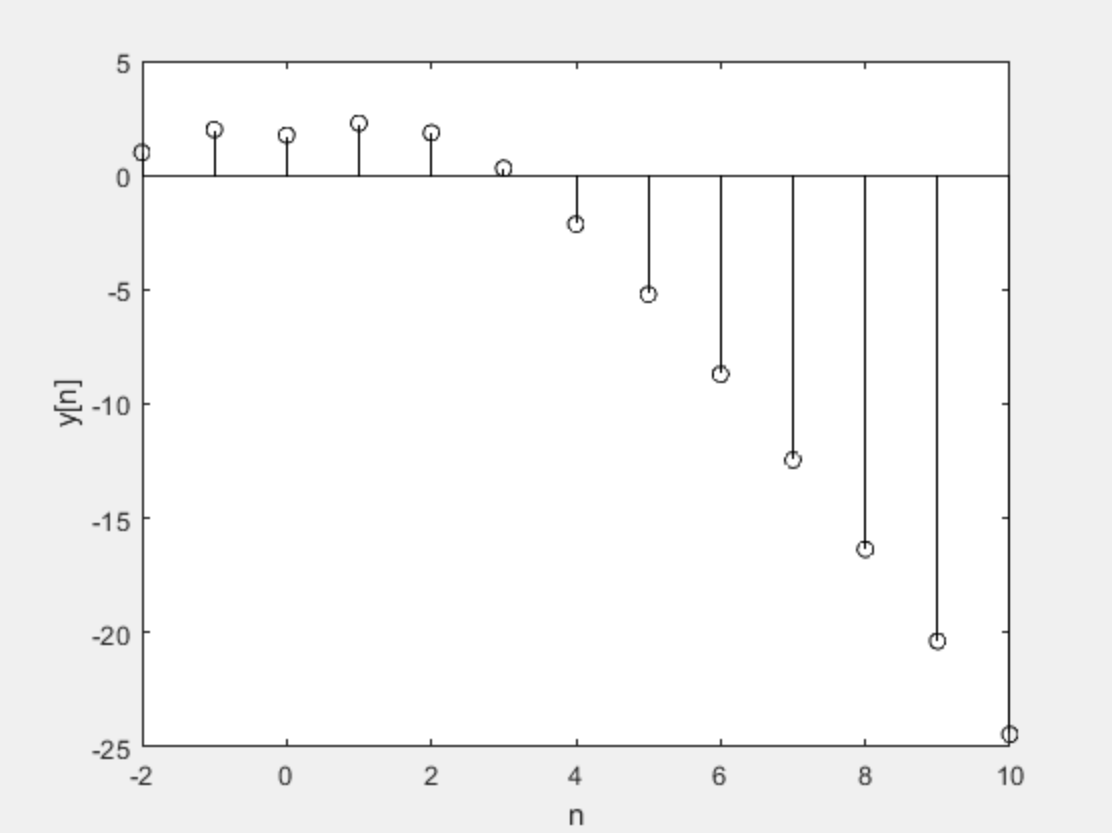
```
>> stem(n,x_c,'k');
>> ylabel('x_c[n]');
>> xlabel('n');
```



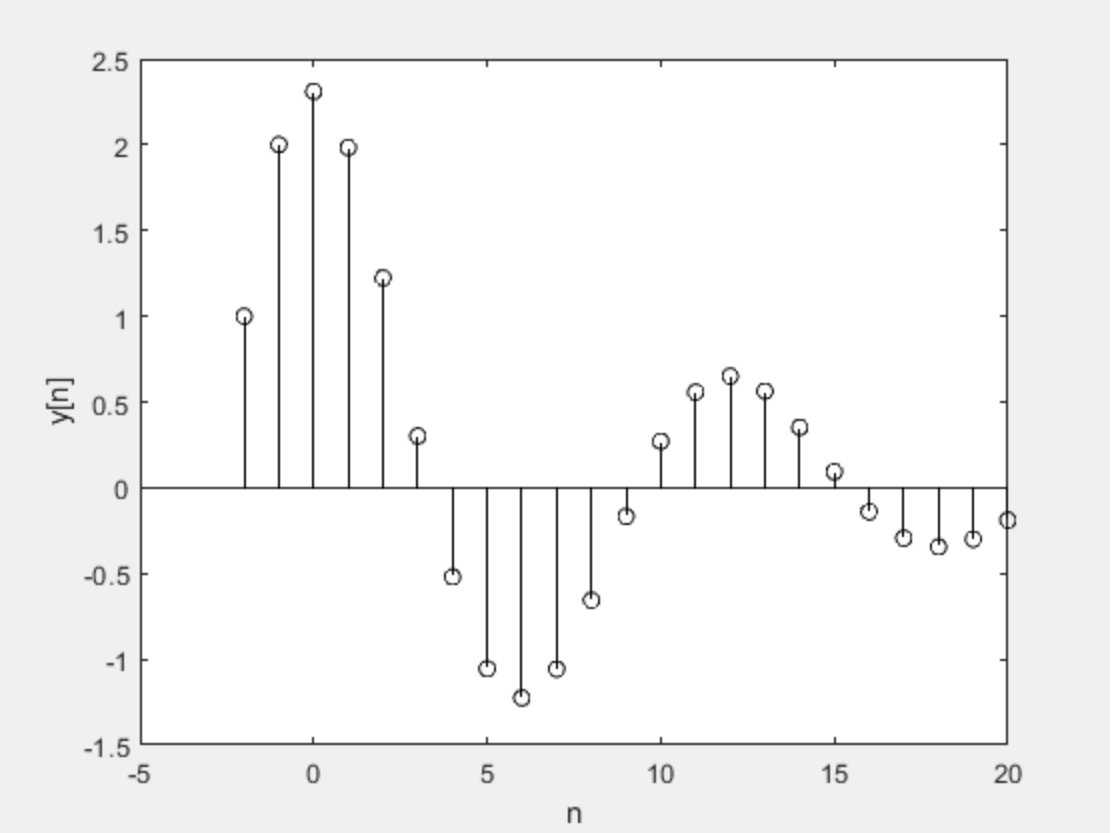

```
>> n = (-30:30);
>> x = cos(n*pi/12 + pi/4);
>> clf;
>> stem(n,x,'k');
>> xlabel('n');
```



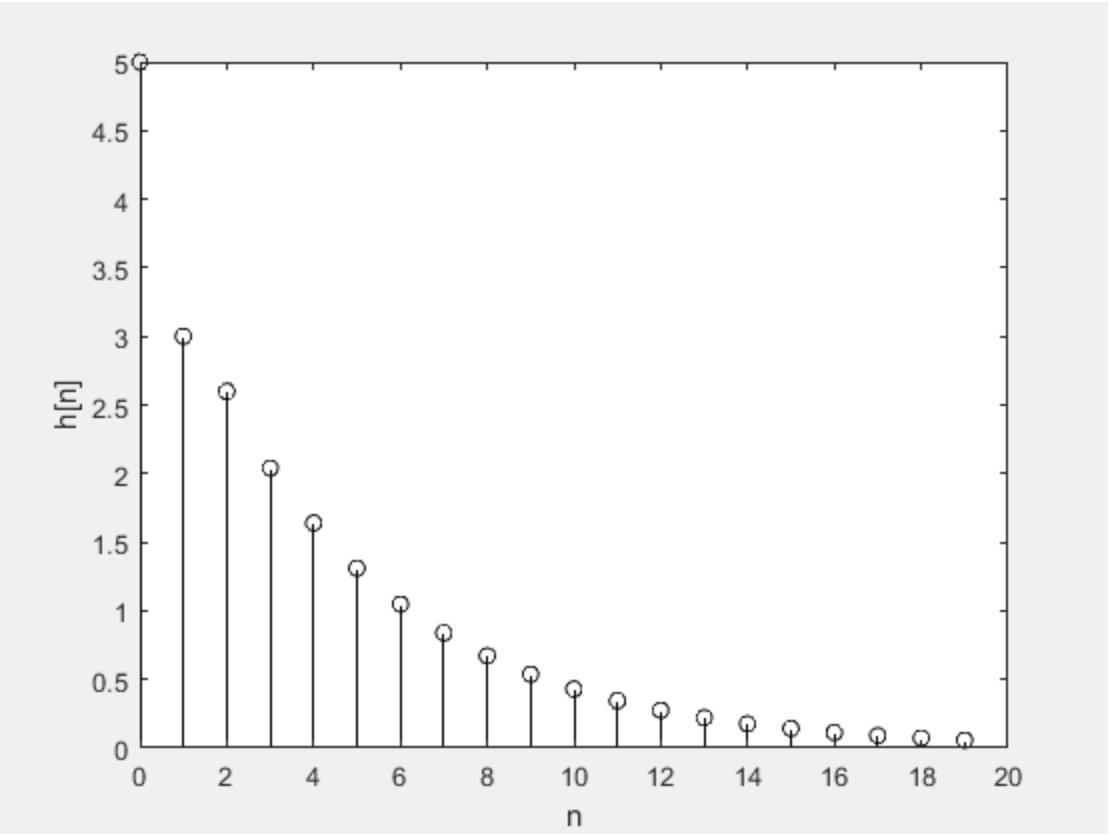

```
>> n = (-2:10)';
>> y = [1;2;zeros(length(n) - 2,1)];
>> x = [0;0;n(3:end)];
>> for k = 1:length(n) - 2,
y(k + 2) = y(k + 1) - 0.24*y(k) + x(k + 2) - 2*x(k + 1);
end;
>> clf;
>> stem(n,y,'k');
>> xlabel('n');
>> ylabel('y[n]');
```



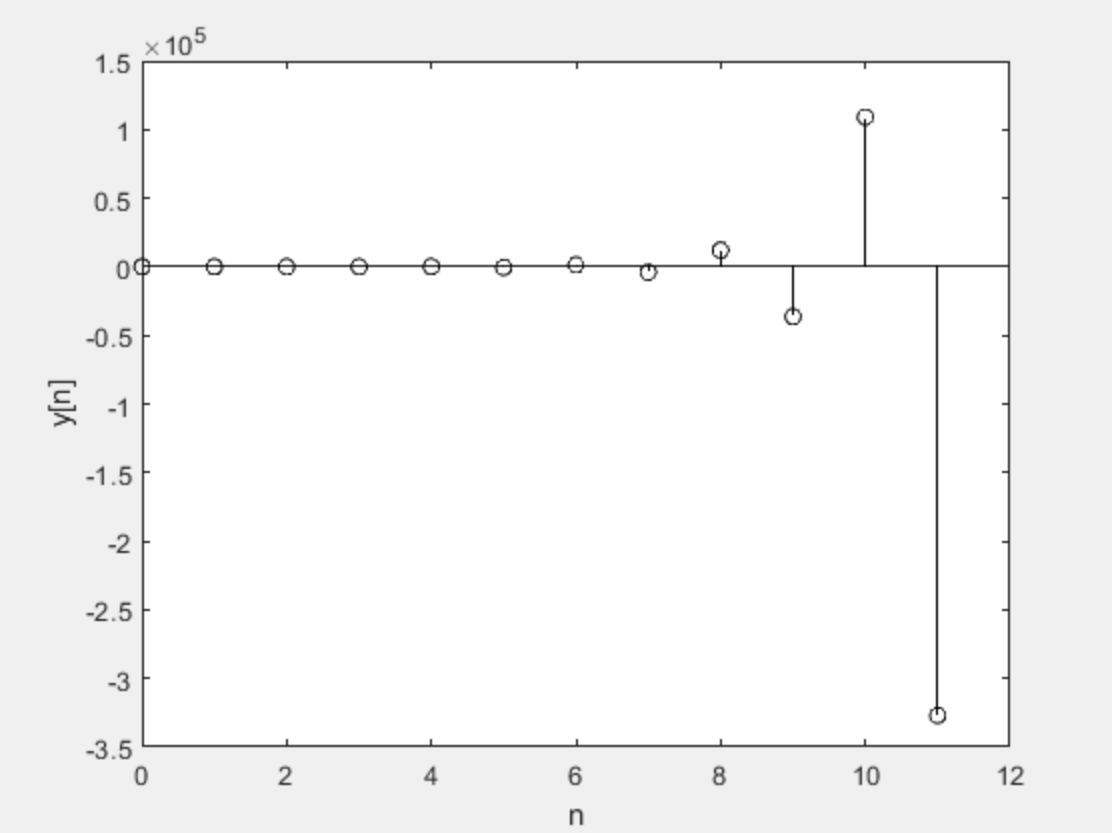

```
>> disp('n                  y'); disp([num2str([n,y])]);
n                  y
-2             1
-1             2
 0          1.76
 1          2.28
 2        1.8576
 3        0.3104
 4      -2.13542
 5      -5.20992
 6      -8.69742
 7       -12.447
 8      -16.3597
 9      -20.3724
10      -24.4461
>> n = (-2:20)';
>> y = [1;2;zeros(length(n) - 2,1)];
>> for k = 1:length(n) - 2,
y(k + 2) = 1.56*y(k + 1) - 0.81*y(k);
end
>> clf;
>> stem(n,y,'k');
>> xlabel('n');
>> ylabel('y[n]');
```



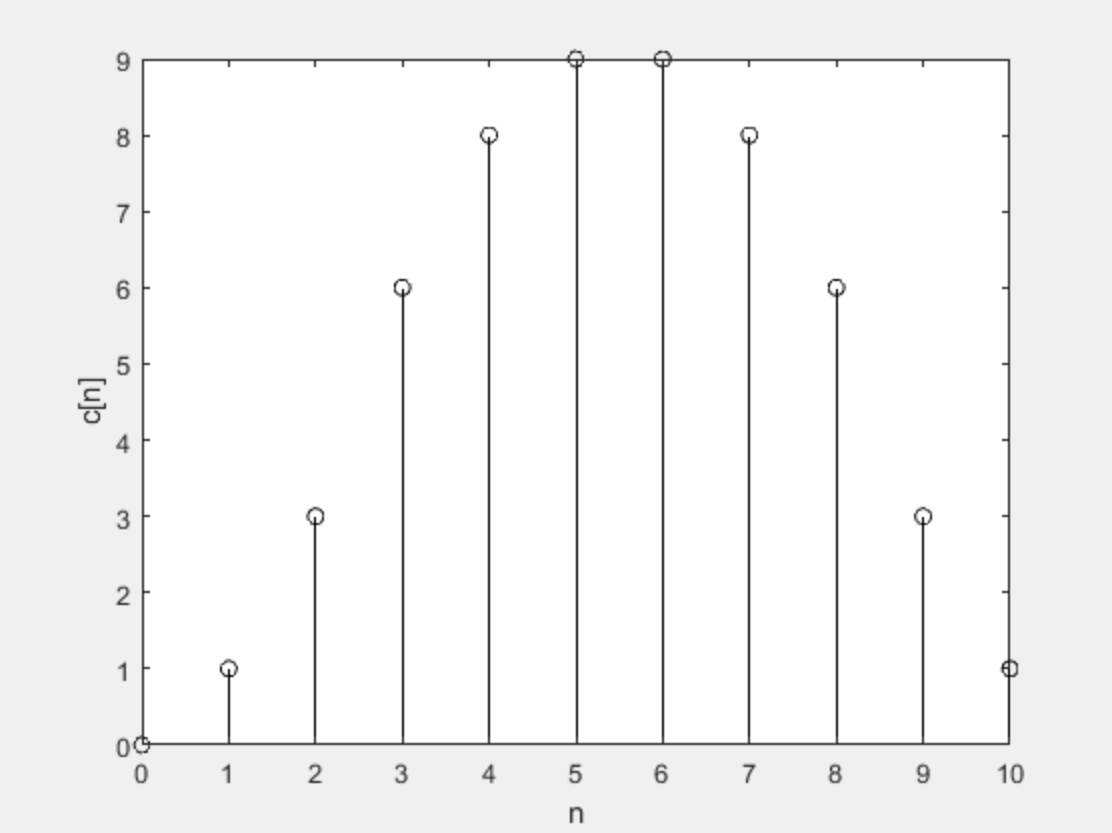

```
>> n = (0:19);
>> x = inline('n == 0');
>> a = [1,-0.6,-0.16];
>> b = [5,0,0];
>> h = filter(b,a,x(n));
>> clf;
>> stem(n,h,'k');
>> xlabel('n');
>> ylabel('h[n]');
```




```
>> n = (0:11);
>> x = inline('(4.^(-n)).*(n >= 0)');
>> a = [1,6,9];
>> b = [2,6,0];
>> y = filter(b,a,x(n));
>> clf;
>> stem(n,y,'k');
>> xlabel('n');
>> ylabel('y[n]');
```




```
>> x = [0,1,2,3,2,1];
>> g = [1,1,1,1,1,1];
>> n = (0:1:length(x) + length(g) - 2);
>> c = conv(x,g);
>> clf;
>> stem(n,c,'k');
>> xlabel('n');
>> ylabel('c[n]');
```


c[n]
n
0
1
2
3
4
5
6
7
8
9
10