

# ECE 3793

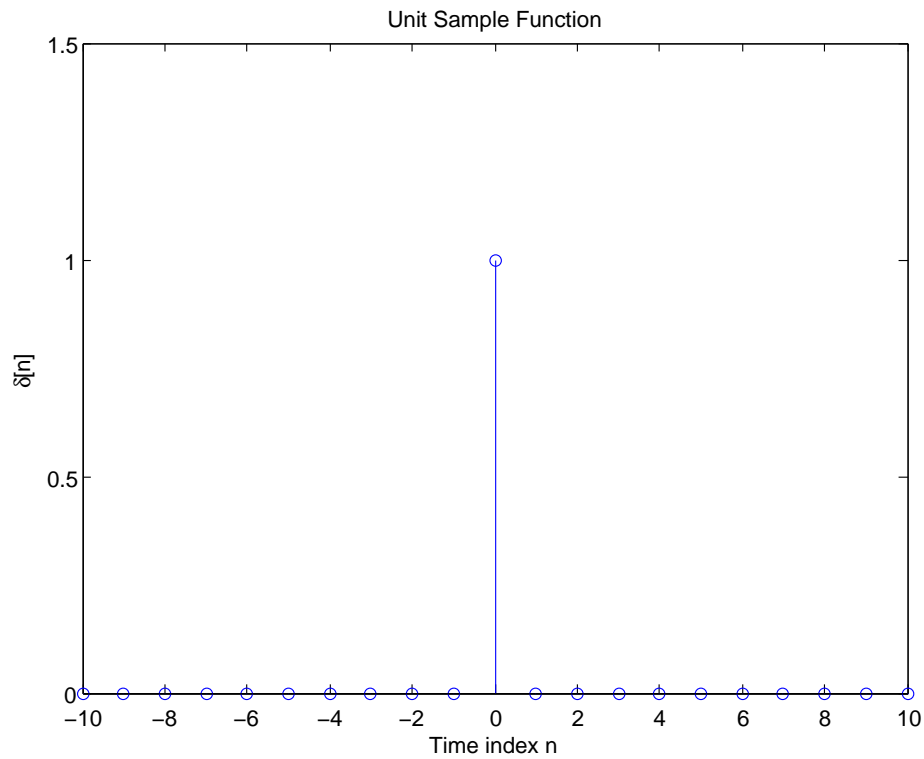
## Matlab Project 1 Solution

Spring 2017

Dr. Havlicek

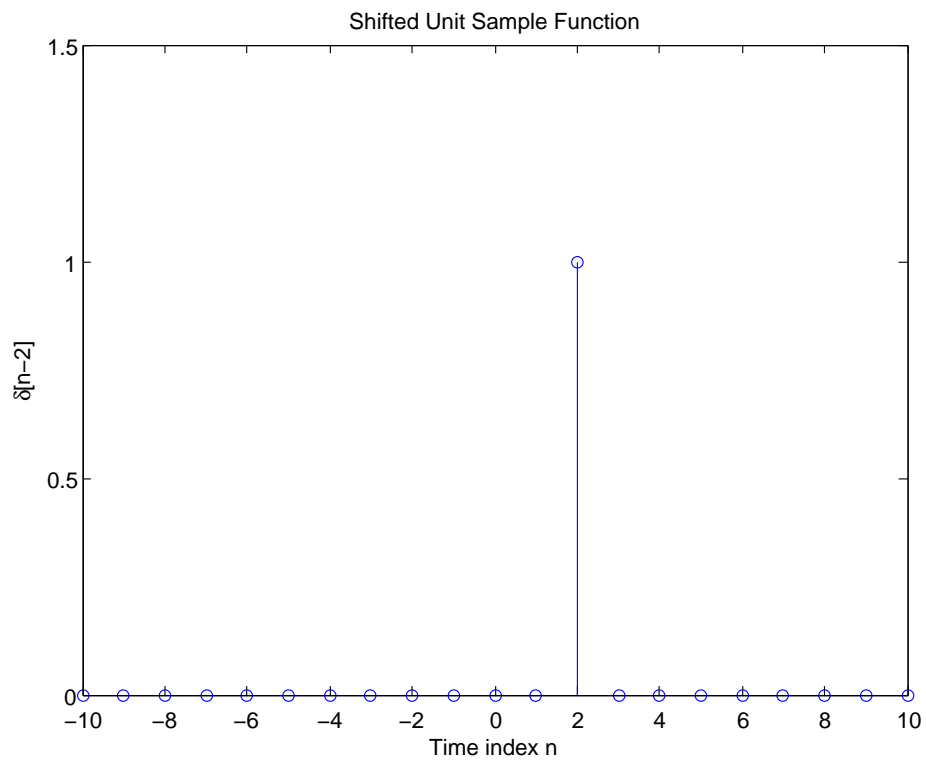
1. (a) Matlab code:

```
%-----  
% P1a  
%  
% generate the signal \delta[n] and plot it  
%  
n = -10:10;          % values of the time variable  
delta_n = [zeros(1,10) 1 zeros(1,10)];  
stem(n,delta_n);  
axis([-10 10 0 1.5]);  
title('Unit Sample Function');  
xlabel('Time index n');  
ylabel('\delta[n]');
```



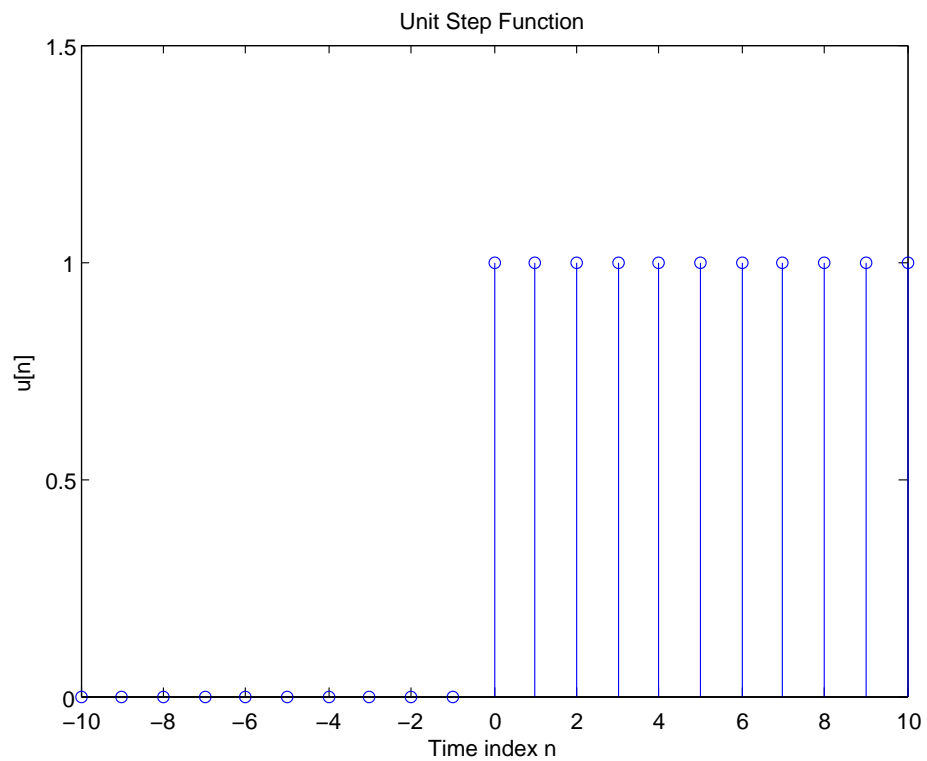
(b)

```
%-----  
% P1b  
%  
% generate the signal \delta[n-2] and plot it  
%  
n = -10:10;          % values of the time variable  
delta_nm2 = [zeros(1,12) 1 zeros(1,8)];  
stem(n,delta_nm2);  
axis([-10 10 0 1.5]);  
title('Shifted Unit Sample Function');  
xlabel('Time index n');  
ylabel('\delta[n-2]');
```



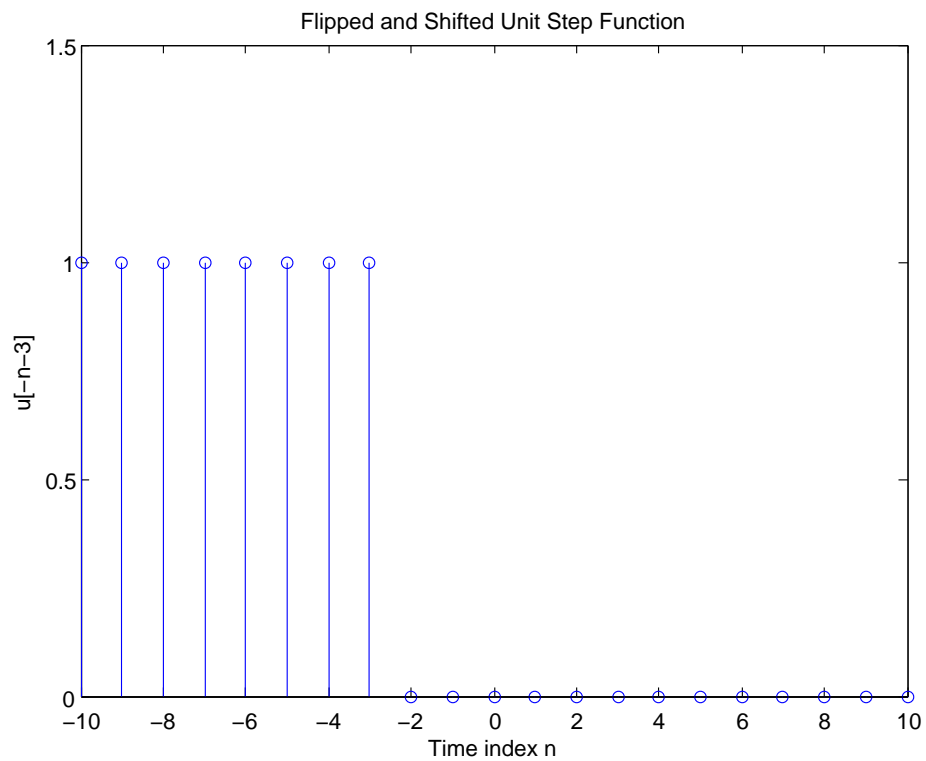
(c)

```
%-----  
% P1c  
%  
% generate the signal u[n] and plot it  
%  
n = -10:10;          % values of the time variable  
un = [zeros(1,10) ones(1,11)];  
stem(n,un);  
axis([-10 10 0 1.5]);  
title('Unit Step Function');  
xlabel('Time index n');  
ylabel('u[n]');
```



(d)

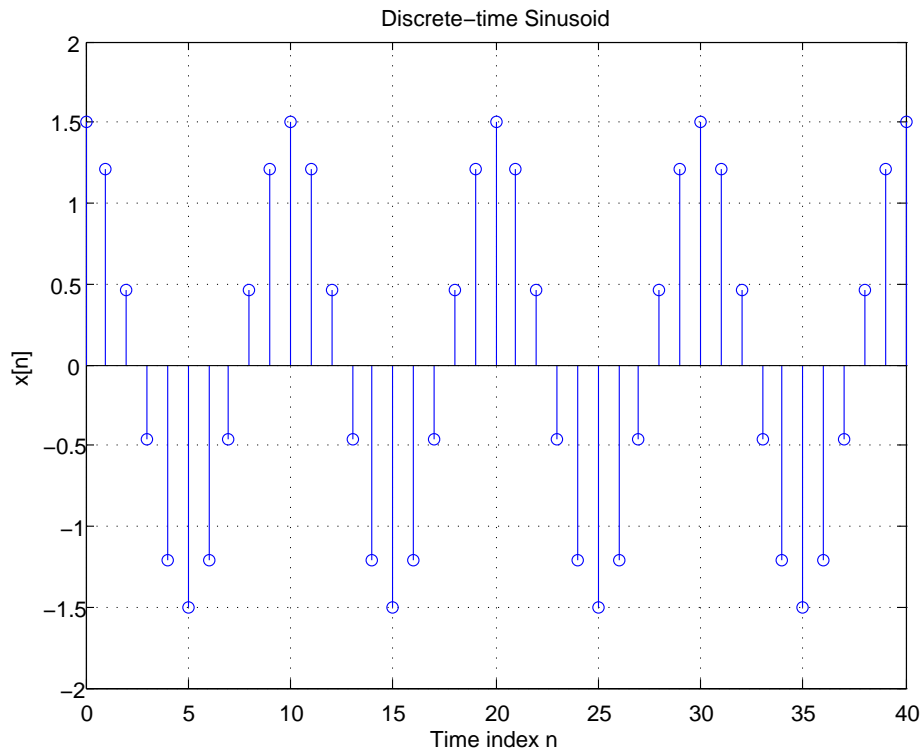
```
%-----  
% P1d  
%  
% generate the signal u[-n-3] and plot it  
%  
n = -10:10;          % values of the time variable  
u_mnm3 = [ones(1,8) zeros(1,13)];  
stem(n,u_mnm3);  
axis([-10 10 0 1.5]);  
title('Flipped and Shifted Unit Step Function');  
xlabel('Time index n');  
ylabel('u[-n-3]');
```



2.

(a)

```
%-----  
% P2a  
%  
% generate and plot a discrete-time cosine signal  
%  
n = 0:40;           % values of the time variable  
w = 0.1*2*pi;      % frequency of the sinusoid.  
phi = 0;           % phase offset.  
A = 1.5;           % amplitude.  
xn = A * cos(w*n - phi);  
stem(n,xn);  
axis([0 40 -2 2]);  
grid;  
title('Discrete-time Sinusoid');  
xlabel('Time index n');  
ylabel('x[n]');
```



(b) The signal has values from  $n = 0$  to  $n = 40$ , so the length is 41.

(c) We have:

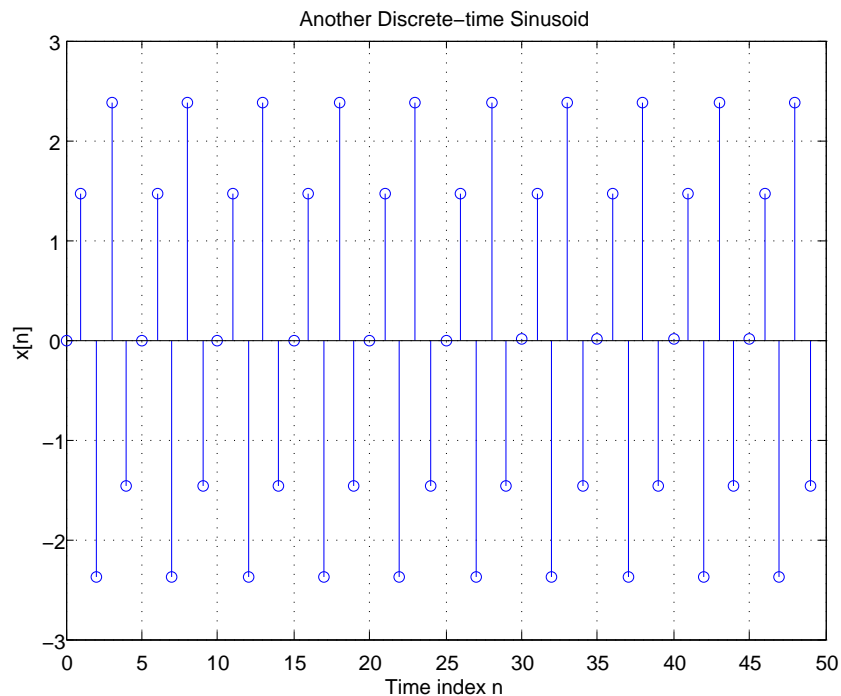
$$\frac{\omega}{2\pi} = \frac{0.2\pi}{2\pi} = \frac{1}{10} = \frac{m}{N}.$$

So the fundamental period is  $N = 10$  and each period of the discrete-time signal looks like  $m = 1$  period of the continuous-time function.

(d) The `grid` command draws grid lines on the graph.

(e)

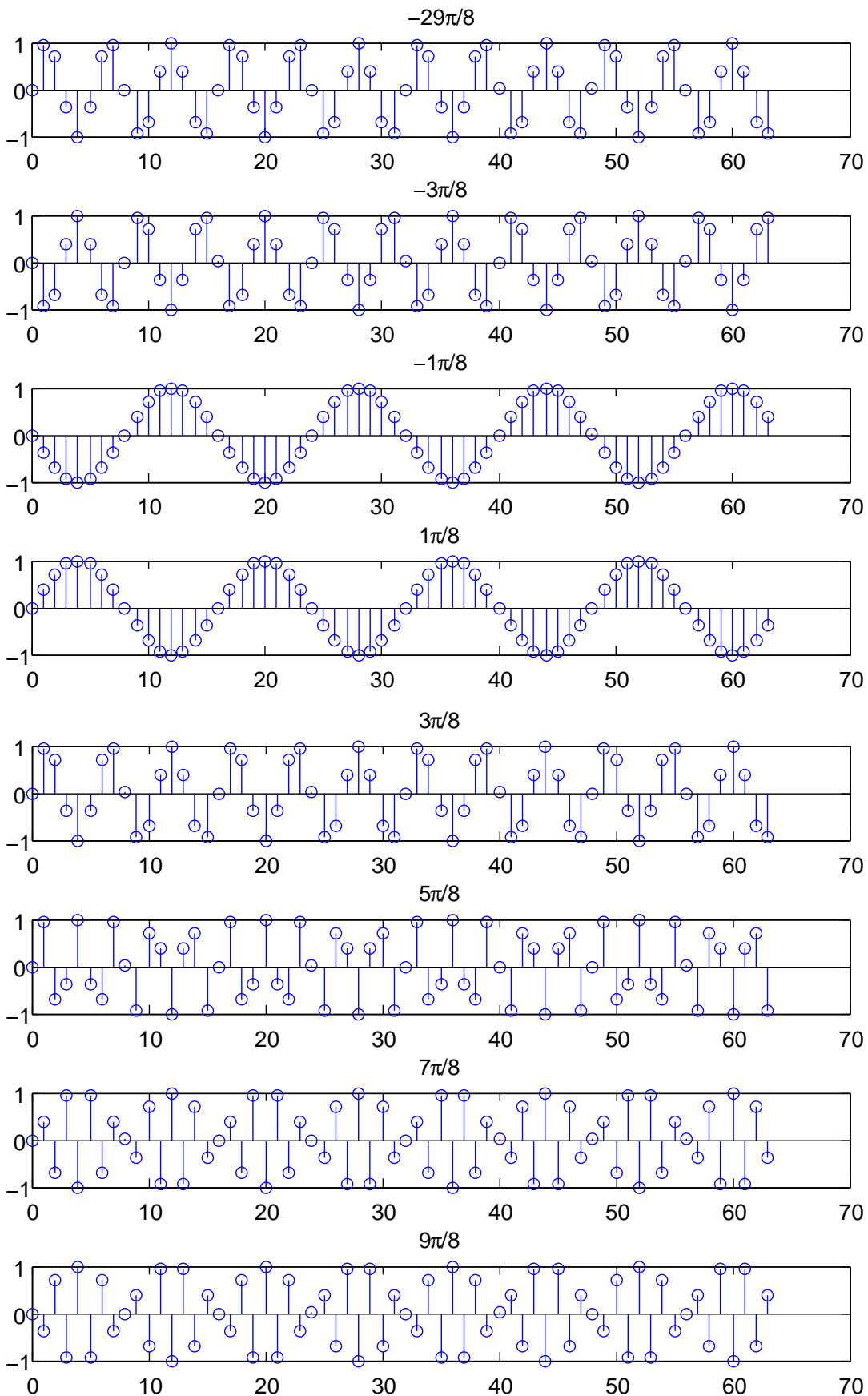
```
%-----  
% P2b  
%  
% generate and plot another discrete-time cosine signal  
%  
n = 0:49;           % values of the time variable  
w = 0.4*2*pi;      % frequency of the sinusoid.  
phi = pi/2;        % phase offset.  
A = 2.5;           % amplitude.  
xn = A * cos(w*n - phi);  
stem(n,xn);  
axis([0 50 -3 3]);  
grid;  
title('Another Discrete-time Sinusoid');  
xlabel('Time index n');  
ylabel('x[n]');
```



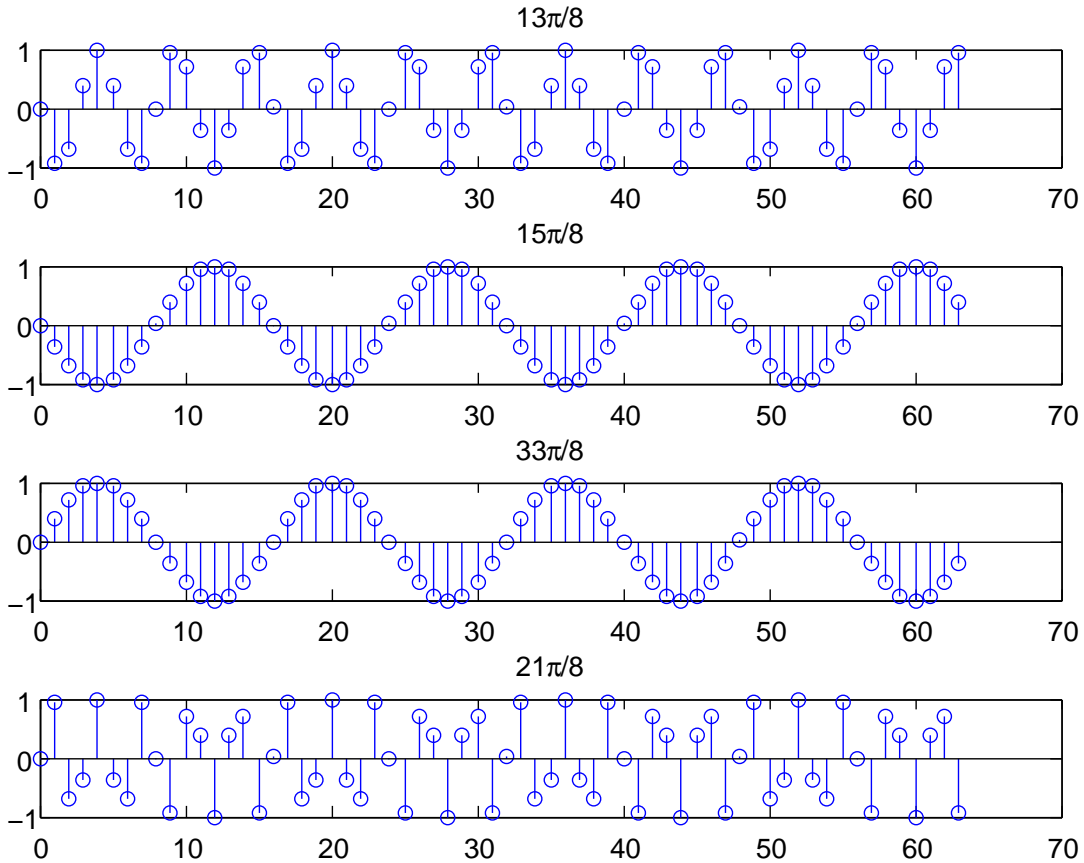
3.

(a)

```
%-----  
% P3a  
%  
% plot a bunch of discrete-time sine signals  
%  
  
% The frequency will be  $w = k\pi/8$ .  
% Load up the k's into a vector:  
%  
kvals = [-29 -3 -1 1 3 5 7 9 13 15 33 21];  
  
% make a counter to index the "next" k to use:  
next_k = 1;  
  
n = 0:63;    % the time variable  
  
% There are 12 k values. We will plot four per  
% figure. So we will need three figures all  
% together. Loop on figures.  
  
for Fig_num=1:3  
    figure(Fig_num);    % selects the "current" figure  
  
    % each time through this loop, we are going to do  
    % 4 of the k's. Loop on Sub Figure number:  
  
    for SubFig_num = 1:4  
        k = kvals(next_k);  
        next_k = next_k + 1;  
        w = k * pi/8;        % the frequency  
        xn = sin(w*n);  
        subplot(4,1,SubFig_num);  
        stem(n,xn);  
        title(sprintf('%d%s',k,'\pi/8'));  
    end    % for SubFig_num  
end    % for Fig_num
```







- (b) For  $\omega_0 = -\frac{3\pi}{8}, -\frac{\pi}{8}, \frac{\pi}{8}, \frac{3\pi}{8}, \frac{5\pi}{8},$  and  $\frac{7\pi}{8}$ , we have that  $-\pi \leq \omega_0 \leq \pi$ . So these discrete sinusoids are not aliased.

The rest of the frequencies  $\omega_0$  are outside the range  $[-\pi, \pi]$ , so these signals will be aliased. Remember, when  $\omega_0$  is outside the range  $[-\pi, \pi]$ , then the graph of  $\sin(\omega_0 n)$  is the same as the graph of  $\sin(\omega_1 n)$  for another frequency  $\omega_1$  that is **inside** the range  $[-\pi, \pi]$  and differs from  $\omega_0$  by an integer multiple of  $2\pi$ .

- Since  $\sin(\omega_0 n) = -\sin(-\omega_0 n)$ , the graph of  $\sin(-\frac{3\pi}{8}n)$  is the negative of the graph of  $\sin(\frac{3\pi}{8}n)$ .
- Likewise, the graph of  $\sin(-\frac{\pi}{8}n)$  is the negative of the graph of  $\sin(\frac{\pi}{8}n)$ .
- For the rest of the frequencies,  $|\omega_0| > \pi$ , so each of these signals is just a *different name* for a discrete sinusoid whose frequency *is* in the range  $[-\pi, \pi]$ .
- $\frac{13\pi}{8} - 2\pi = -\frac{3\pi}{8}$ , so the graph of  $\sin(\frac{13\pi}{8}n)$  is the same as the graph of  $\sin(-\frac{3\pi}{8}n)$ .
- Likewise,  $\frac{15\pi}{8} - 2\pi = -\frac{\pi}{8}$ , so the graph of  $\sin(\frac{15\pi}{8}n)$  is the same as the graph of  $\sin(-\frac{\pi}{8}n)$ .
- $\frac{33\pi}{8} - 2\pi = \frac{17\pi}{8}$  and  $\frac{17\pi}{8} - 2\pi = \frac{\pi}{8}$ . So  $\sin(\frac{33\pi}{8}n)$  and  $\sin(\frac{\pi}{8}n)$  are the same signal. Their graphs are the same.
- $-\frac{29\pi}{8} + 4\pi = \frac{3\pi}{8}$ , so  $\sin(-\frac{29\pi}{8}n)$  and  $\sin(\frac{3\pi}{8}n)$  are the same signal; they have the same graph.

- $\frac{21\pi}{8} - 2\pi = \frac{5\pi}{8}$ , so  $\sin\left(\frac{21\pi}{8}n\right)$  and  $\sin\left(\frac{5\pi}{8}n\right)$  have the same graph.
- (c)  $\frac{9\pi}{8} - 2\pi = -\frac{7\pi}{8}$ , so  $\sin\left(\frac{9\pi}{8}n\right) = -\sin\left(\frac{7\pi}{8}n\right)$ . The graphs of  $\sin\left(\frac{9\pi}{8}n\right)$  and  $\sin\left(\frac{7\pi}{8}n\right)$  are therefore negatives of one another.

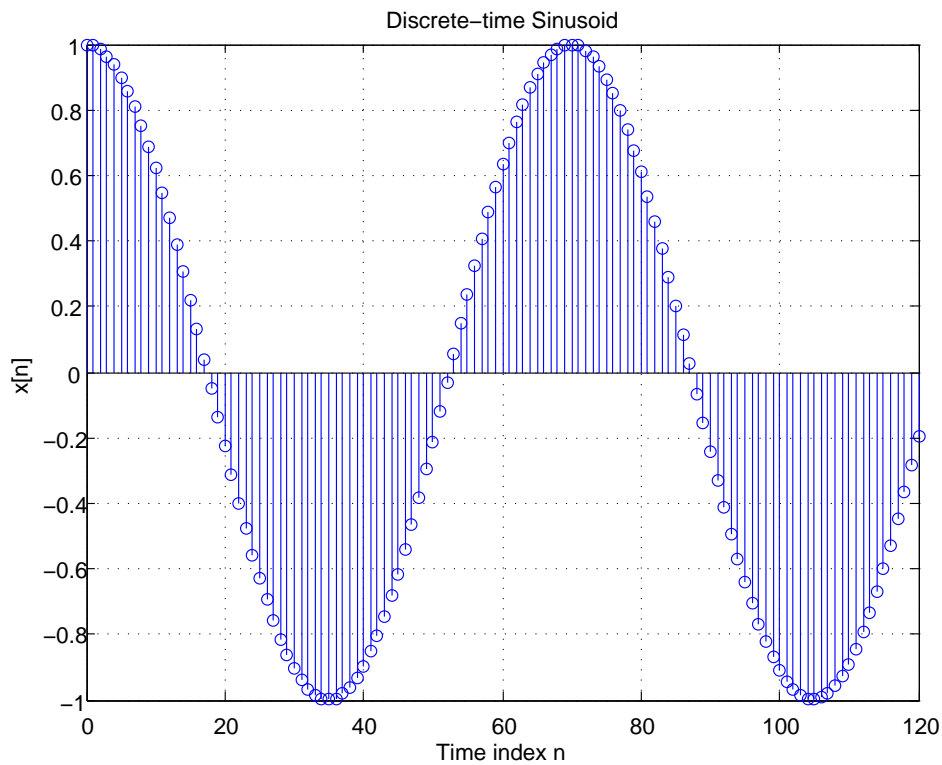
4.

(a)

```

%-----
% P4a
%
% generate and plot a discrete-time cosine signal
%
n = 0:120;                                % values of the time variable
w = 0.09;                                  % frequency of the sinusoid.
xn = cos(w*n);
stem(n,xn);
axis([0 120 -1.0 1.0]);
grid;
title('Discrete-time Sinusoid');
xlabel('Time index n');
ylabel('x[n]');

```



(b) We have:

$$\frac{\omega}{2\pi} = \frac{0.09}{2\pi} \notin \mathbb{Q}.$$

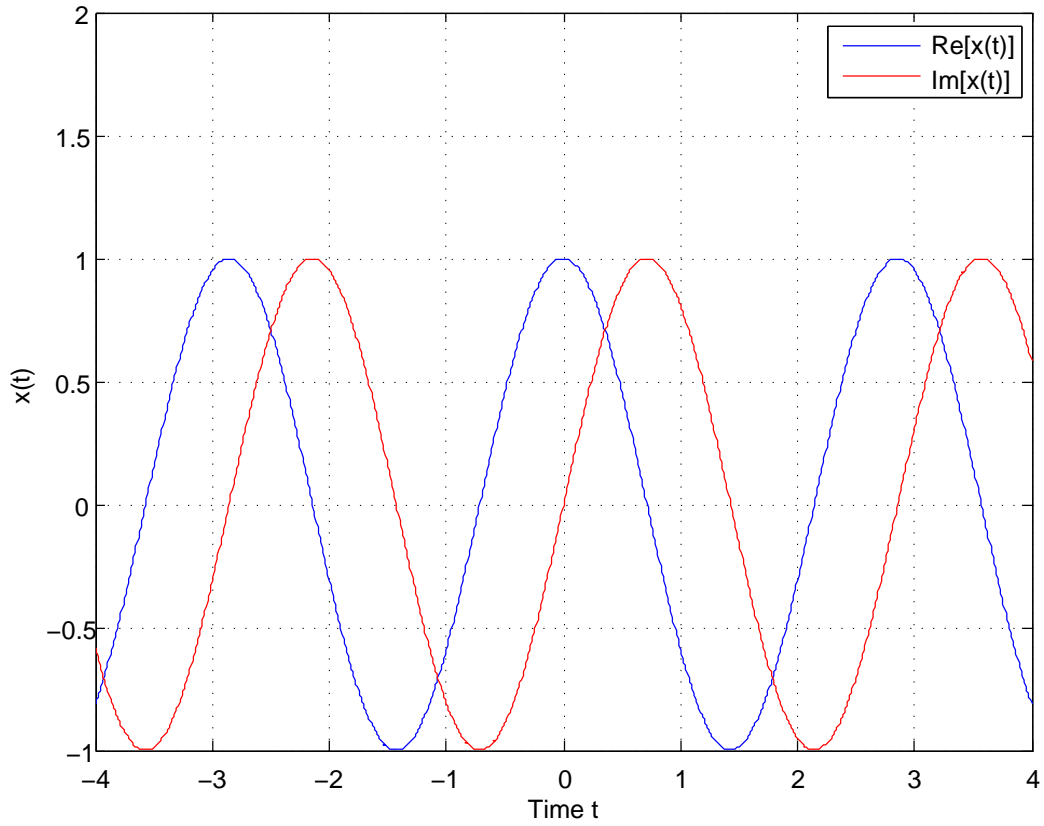
The signal is therefore **not periodic**.

5.

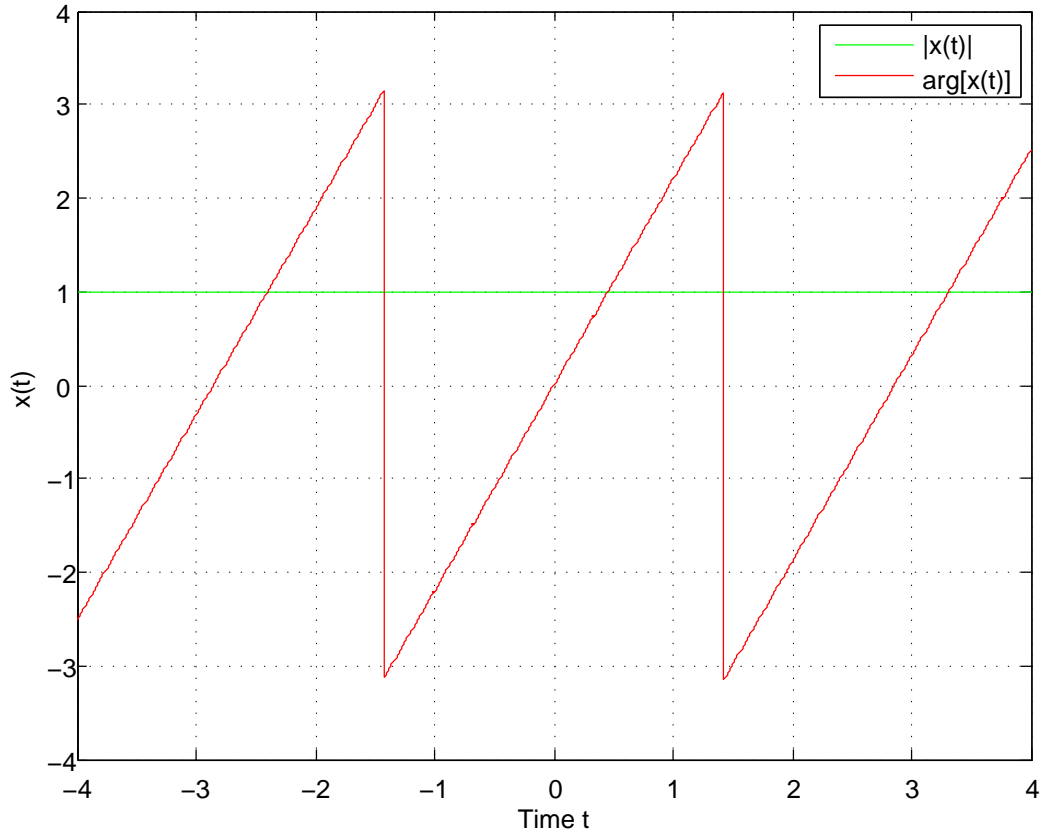
(a)

```
%-----  
% P5a  
%  
% generate and plot a continuous-time complex sinusoid  
%  
t = -4:0.01:4; % values of the time variable  
w = 2.2; % frequency of the sinusoid.  
xt = exp(j*w*t);  
xtR = real(xt);  
xtI = imag(xt);  
figure(1); % make Fig 1 active  
plot(t,xtR,'-b'); % '-b' means 'solid blue line'  
axis([-4 4 -1.0 2.0]);  
grid;  
hold on; % add more curves to the same graph  
plot(t,xtI,'-r'); % 'r' = red  
title('Real and Imaginary parts');  
xlabel('Time t');  
ylabel('x(t)');  
legend('Re[x(t)]','Im[x(t)]');  
hold off;  
  
mag = abs(xt);  
phase = angle(xt);  
figure(2); % make Fig 2 active  
plot(t,mag,'-g'); % '-' = solid line; 'g' = green  
grid;  
hold on; % add more curves to the graph  
plot(t,phase,'-r'); % 'r' = red  
title('Magnitude and Phase');  
legend('|x(t)|','arg[x(t)]');  
xlabel('Time t');  
ylabel('x(t)');  
hold off;
```

Real and Imaginary parts



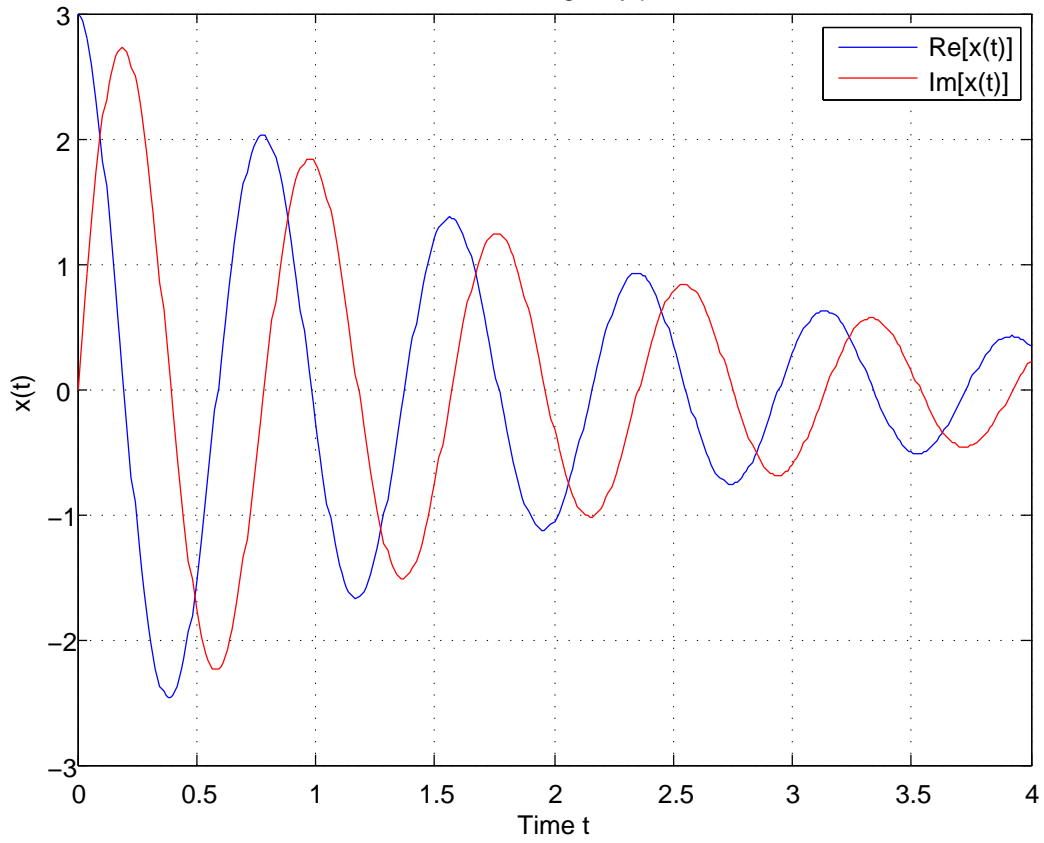
Magnitude and Phase



(b)

```
%-----  
% P5b  
%  
% generate and plot a continuous-time damped complex exponential  
%  
t = 0:0.01:4;           % values of the time variable  
w = 8.0;               % frequency of the sinusoid.  
xt = 3.0*exp(-t/2).*exp(j*w*t);  
xtR = real(xt);  
xtI = imag(xt);  
figure(1);             % make Fig 1 active  
plot(t,xtR,'-b');      % '-b' means 'solid blue line'  
axis([0 4 -3 3]);  
grid;  
hold on;               % add more curves to the same graph  
plot(t,xtI,'-r');      % 'r' = red  
title('Real and Imaginary parts');  
xlabel('Time t');  
ylabel('x(t)');  
legend('Re[x(t)]','Im[x(t)]');  
hold off;  
  
mag = abs(xt);  
phase = angle(xt);  
figure(2);             % make Fig 2 active  
plot(t,mag,'-g');      % '-' = solid line; 'g' = green  
grid;  
hold on;               % add more curves to the graph  
plot(t,phase,'-r');    % 'r' = red  
title('Magnitude and Phase');  
legend('|x(t)|','arg[x(t)]');  
xlabel('Time t');  
ylabel('x(t)');  
hold off;
```

Real and Imaginary parts



Magnitude and Phase

