

ELG3125B

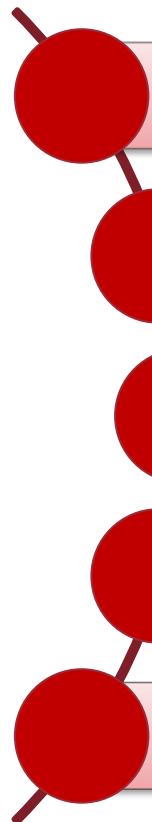
Signal and System Analysis Lab

- **Lab2: Signal Manipulation**

By: Hitham Jleed

hjleed@uottawa.ca

Outline

- 
- Continuous-Time Signals**
 - Discrete-Time Signals**
 - Basic Signals**
 - Operations Signals**
 - Transformation of the independent variable**

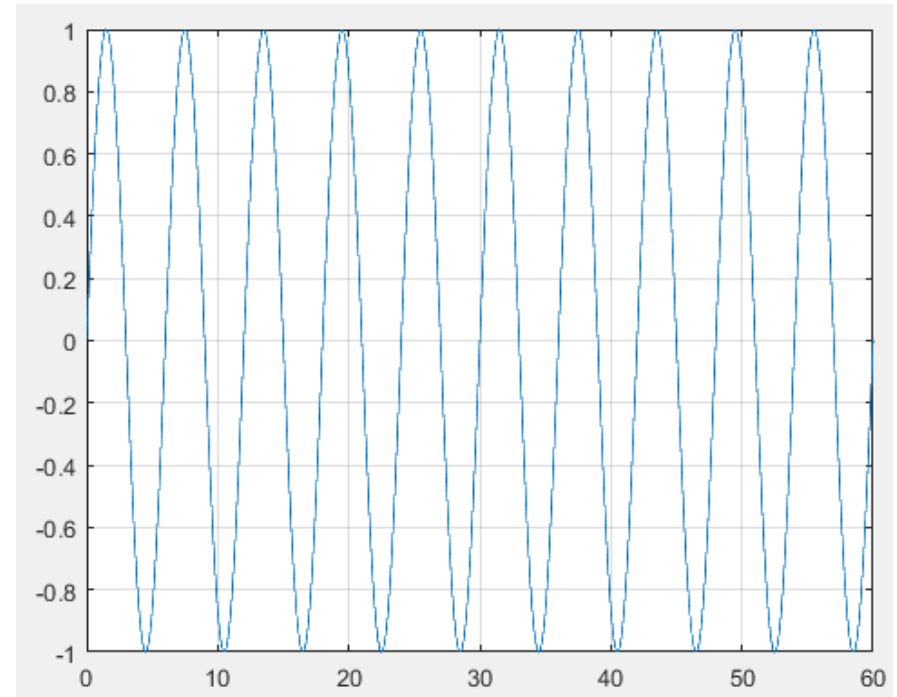


Continuous-Time Signals (1)

1. plot (sin) signal with period T:

$$y = \sin\left(\frac{2\pi}{T}t\right)$$

```
T=6; %period  
t=0:0.01:60;  
y=sin(2*pi/T.*t);  
plot(t,y);  
grid;
```



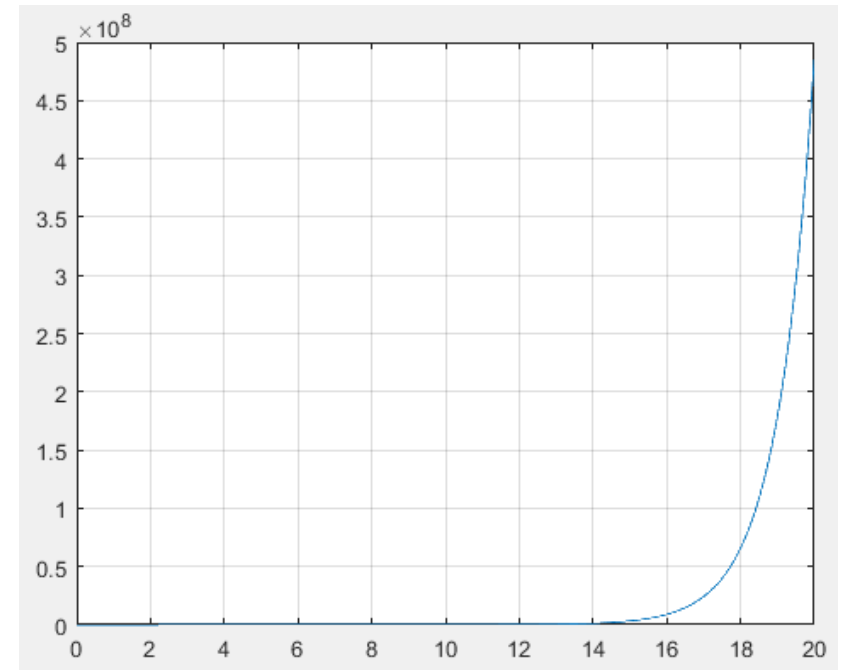
*If you don't see grid use, "grid on" command!
Do you see grid now?

Continuous-Time Signals (2)

2. plot Exponential Function:

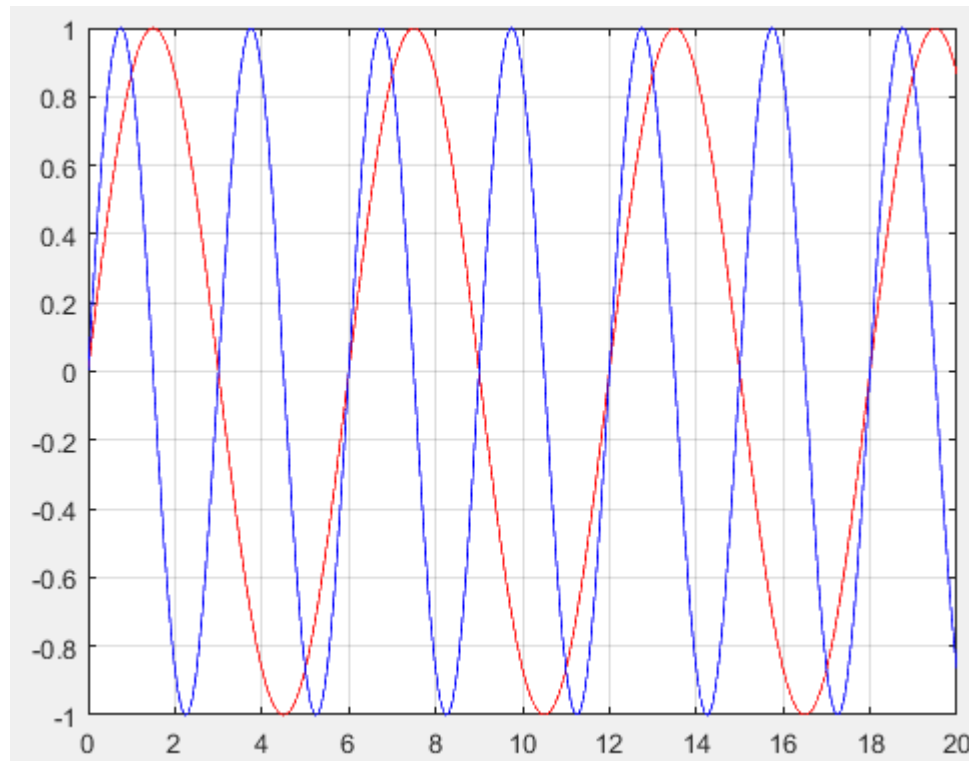
$$x = e^{\omega_0 t}$$

```
t=0:0.01:20;  
omega=1;  
y=exp(omega.*t);  
plot(t,y),grid;
```



Two Continuous-time Signals in One Graph (Method1)

```
% Use plot function  
% Sinusoidal 1  
T=6; t=0:0.01:20;  
y1=sin(2*pi/T.*t);  
% Sinusoidal 2  
y2=sin(4*pi/T.*t);  
plot(t,y1,'r',t,y2,'b'),grid;
```



Two Continuous-time Signals in One Graph (Method2) – use “hold on”

```
% Sinusoidal 1
```

```
T=6; t=0:0.01:20;
```

```
y1=sin(2*pi/T.*t);
```

```
plot(t,y1,'r');
```

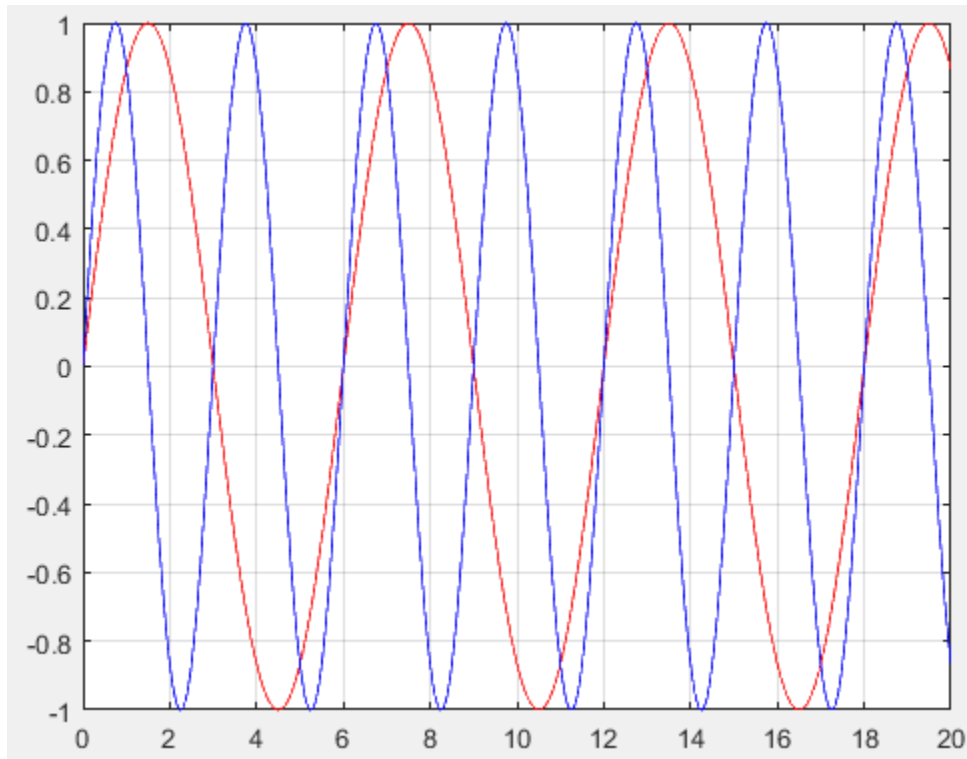
```
hold on
```

```
% Sinusoidal 2
```

```
y2=sin(4*pi/T.*t);
```

```
plot(t,y2,'b');
```

```
grid on;
```

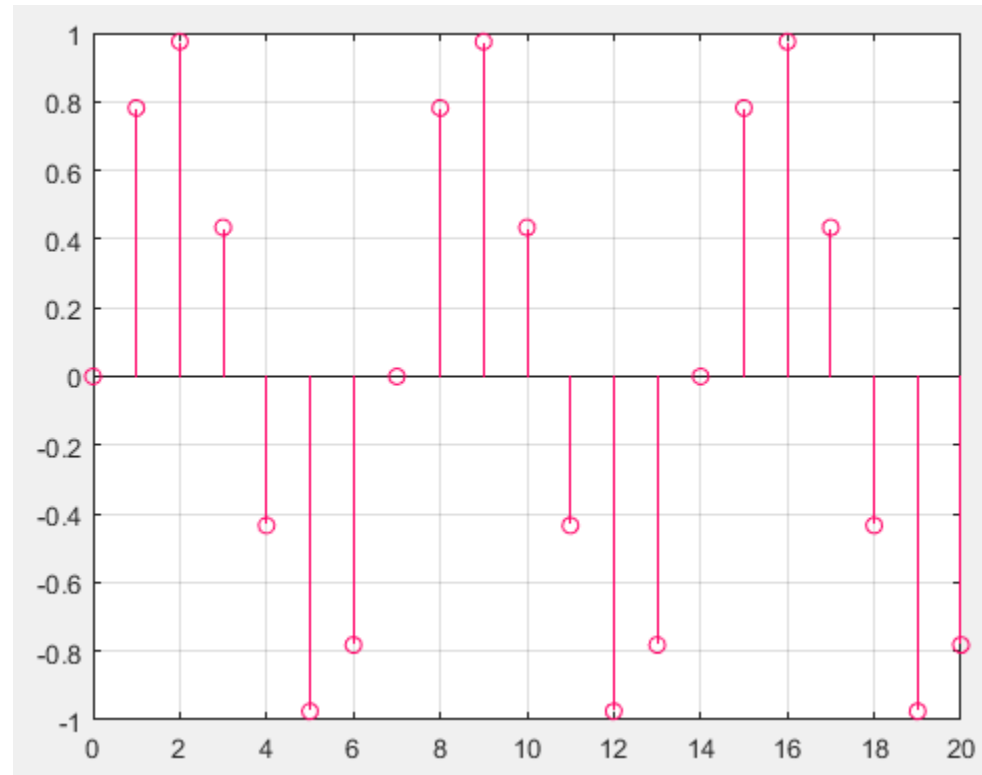


Discrete-Time Signals (1)

1. plot (sin) signal with period N:

$$y = \sin\left(\frac{2\pi}{N}n\right)$$

```
n=0:20;
N=7; %period
y=sin(2*pi/N.*n);
stem(n,y,'r'),grid;
```



`%plot()` is for continuous signals and `stem()` is for discrete signals



Discrete-Time Signals (2)

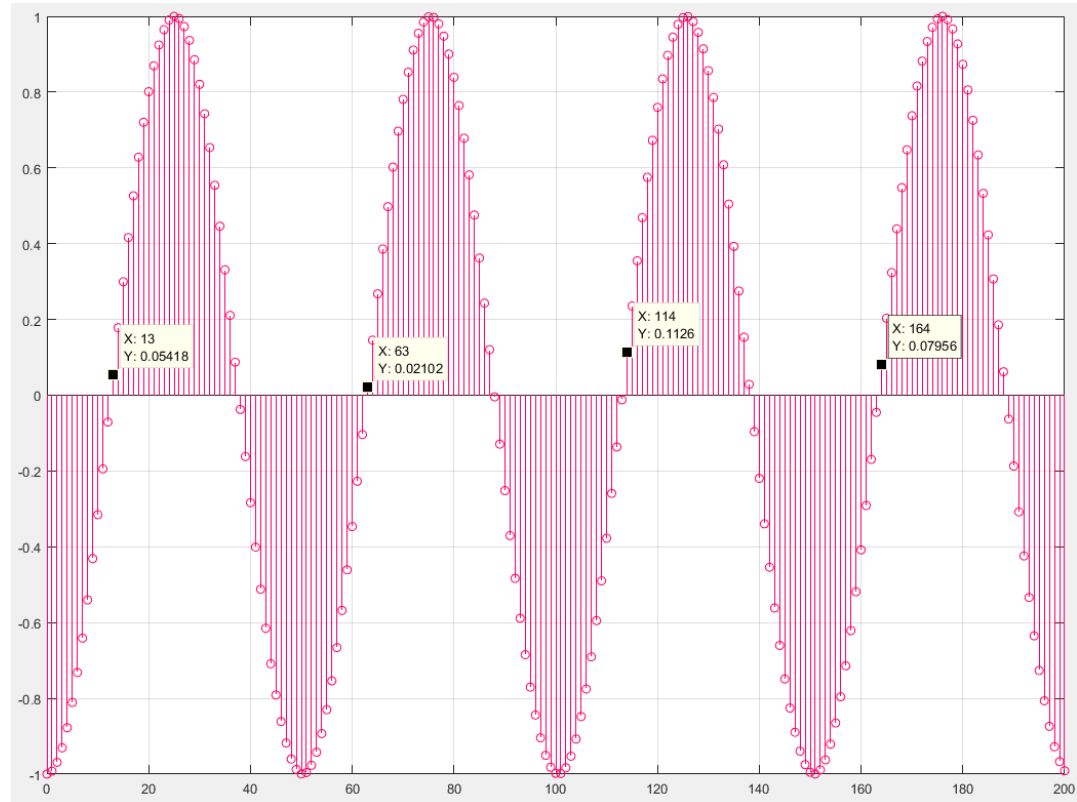
Let's plot Cosine signal:

$$x[n] = \cos\left(\frac{n}{8} - \pi\right)$$

```
n=0:200;
x=cos(n./8-pi);
stem(n,x);
grid on;
```

NOT periodic:

$$x[n] \neq x[n + N]$$



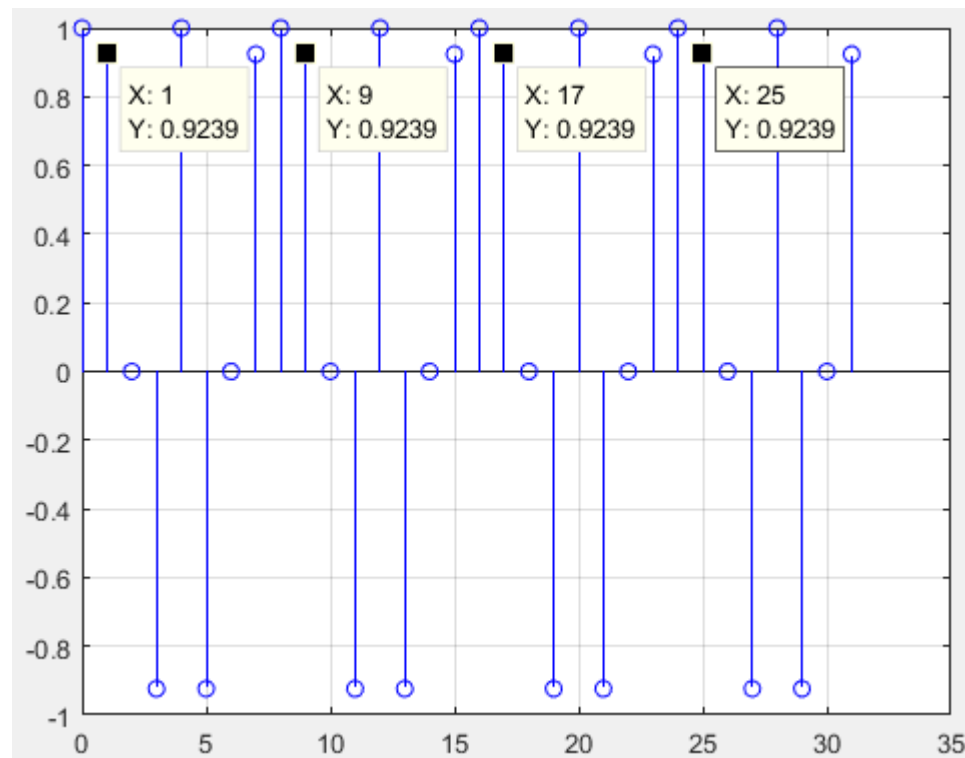
Discrete-Time Signals (3)

$$x[n] = \cos\left(\frac{\pi}{8} \times n^2\right)$$

$$x[n] = x[n + N]$$

Periodic and $N=8$

```
n=0:33;
x=cos(power(n,2)*pi/8);
stem(n,x,'b'),grid;
```

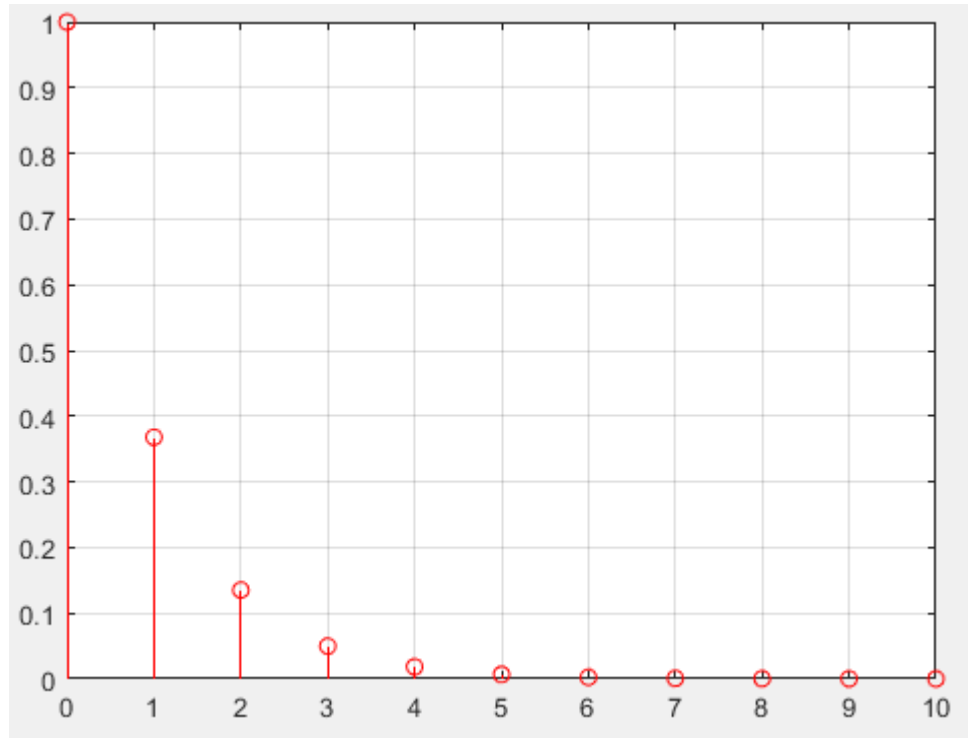


Discrete-Time Signals (4)

Exponential signal:

$$x = e^{-n}$$

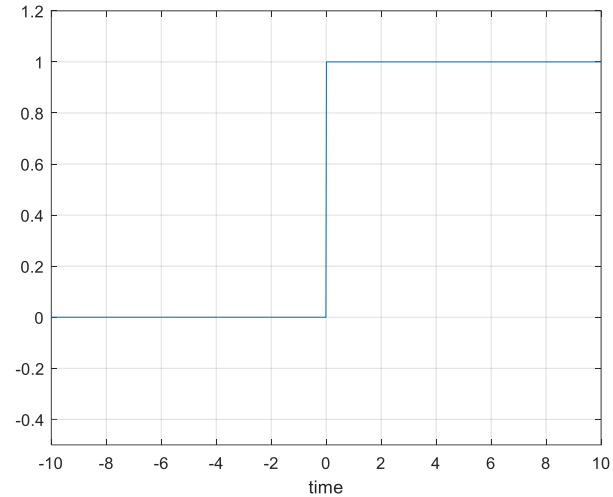
```
n=0:10;  
y=exp(-n);  
stem(n,y,'r');  
grid on;
```



Basic Signals (Continuous-time)

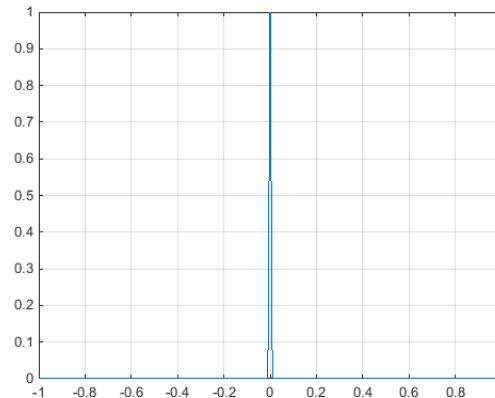
1. Unit Step

```
t=-10:0.01:10; % step is small enough to represent  
continuous-time signal  
f=heaviside(t); % the unit step function.  
plot(t,f); grid on  
xlabel('time')  
axis([-10,10,-0.5,1.2])
```



2. Unit Impulse

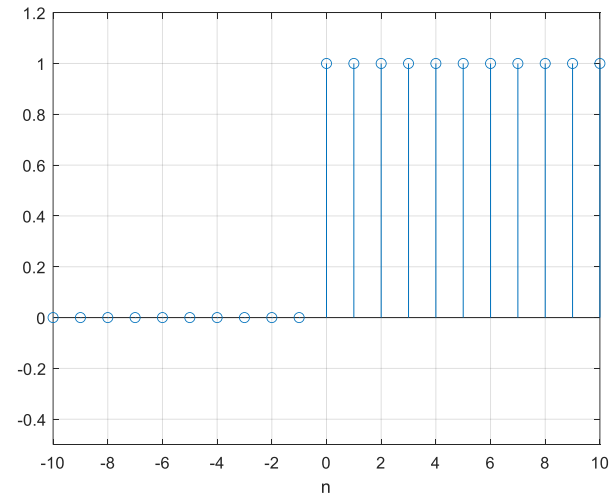
```
t = -1:0.01:1;  
y = dirac(t);  
idx = y == Inf; % find Inf  
y(idx) = 1; % set Inf to finite value  
plot(t,y);grid
```



Basic Signals (Discrete-Time)

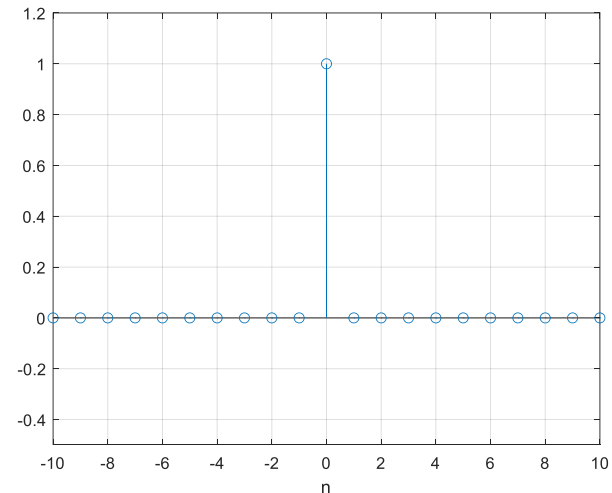
1. Unit Step

```
n = -10:10;  
%discrete-time step function:  
f = (n >= 0);  
stem(n,f) ; grid on  
xlabel('n')  
axis([-10,10,-0.5,1.2])
```



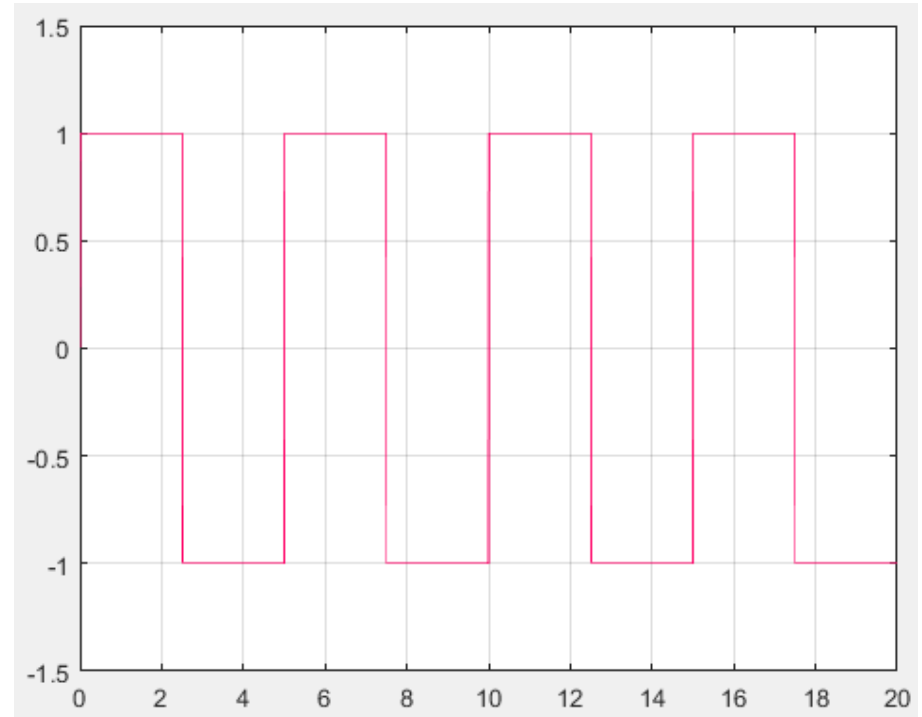
2. Unit Impulse

```
n = -10:10;  
% impulse discrete-time function:  
f = (n == 0);  
stem(n,f) ; grid on  
xlabel('n')  
axis([-10,10,-0.5,1.2])
```



3. Square wave

```
t=0:0.01:20;  
T=5; %period  
y=sign(sin(2*pi/T.*t));  
%or %y=mod(t.*1/T,1)>1/2;  
plot(t,y),grid;  
axis([0 20 -1.5 1.5]);  
grid on;
```

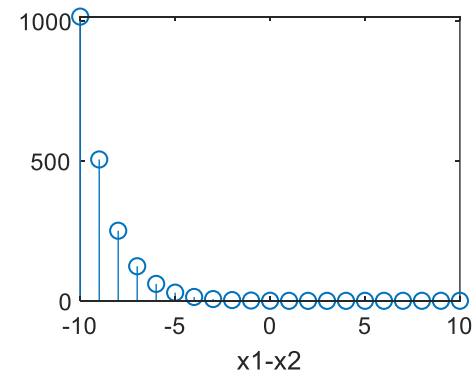
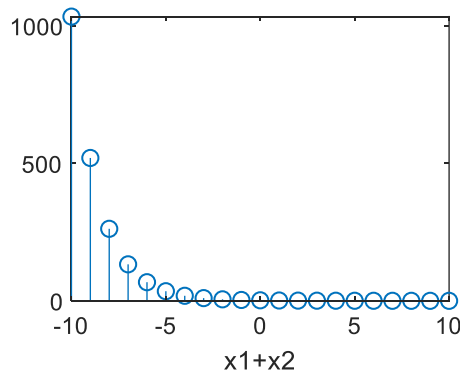
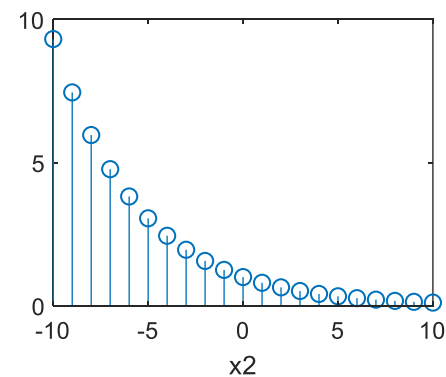
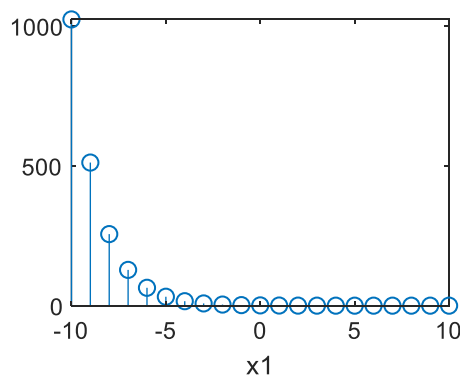


Operations Signals

1. Addition & Subtraction

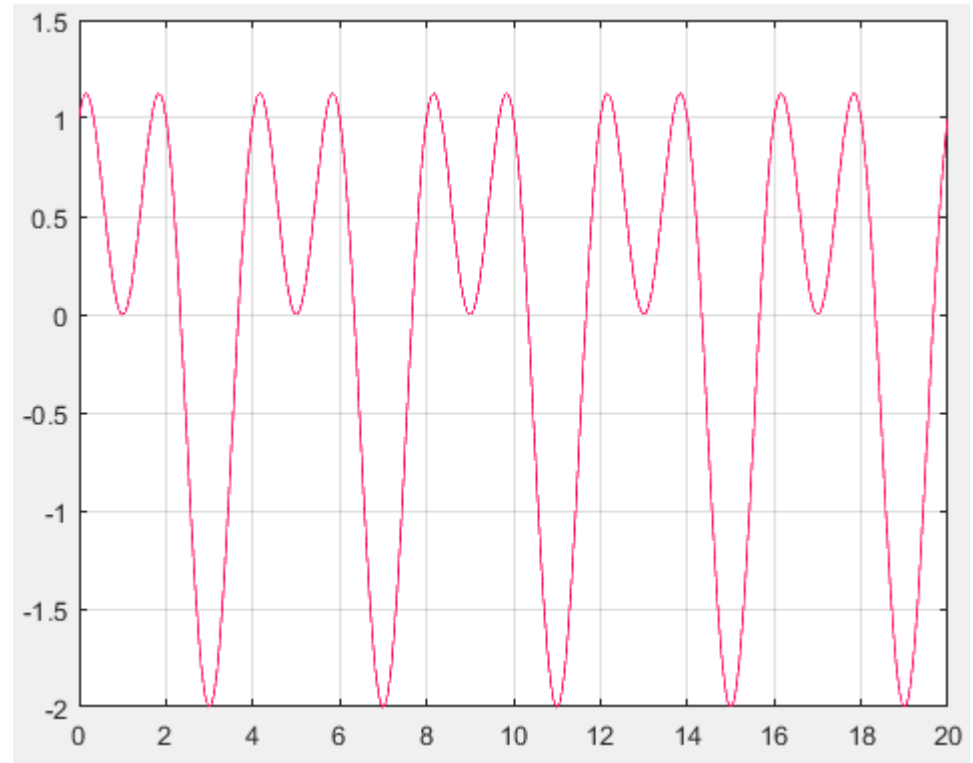
```

n=-10:10;
x1= 0.5.^n;
x2= 0.8.^n;
x_add=x1+x2;
x_sub=x1-x2;
subplot(221)
stem(n,x1); xlabel('x1')
subplot(222)
stem(n,x2); xlabel('x2')
subplot(223)
stem(n,x_add); xlabel('x1+x2')
subplot(224)
stem(n,x_sub); xlabel('x1-x2')
    
```



Addition of Two Continuous-Time Signals

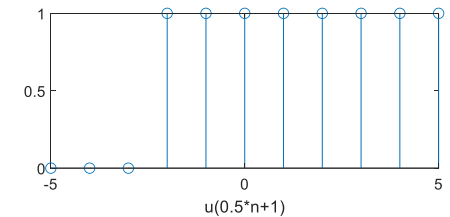
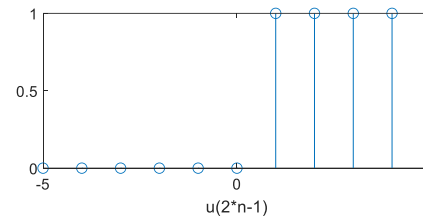
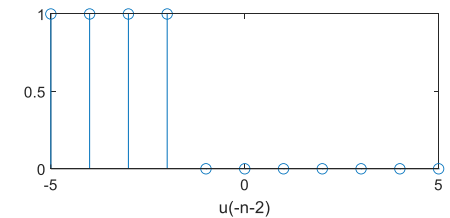
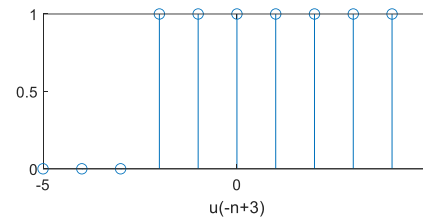
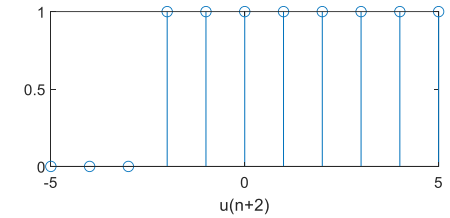
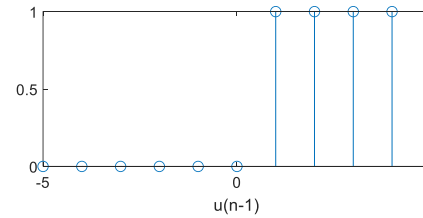
```
t=0:0.01:20;  
T1=2;  
T2=4;  
y1=cos(2*pi/T1*t);  
y2=sin(2*pi/T2*t);  
y3=y1+y2;  
plot(t,y3),grid;
```



Transformation of the independent variable

It is now desired to focus on the transformation of the independent variable while *shifting*, *reversing*, and *scaling*. It will be covered first for the unit step functions and then it will be left for you as a task to do it for any general signal $x[n]$.

```
n=-5:5;
x=heaviside(n-1);
subplot(321); stem(n,x); xlabel('u(n-1)')
y=heaviside(n+2);
subplot(322); stem(n,y); xlabel('u(n+2)')
z= heaviside(-n+3);
subplot(323); stem(n,y); xlabel('u(-n+3)')
k= heaviside(-n-2);
subplot(324); stem(n,k); xlabel('u(-n-2)')
l= heaviside(2*n-1);
subplot(325); stem(n,l); xlabel('u(2*n-1)')
m= heaviside(0.5*n+1);
subplot(326); stem(n,m); xlabel('u(0.5*n+1)')
```



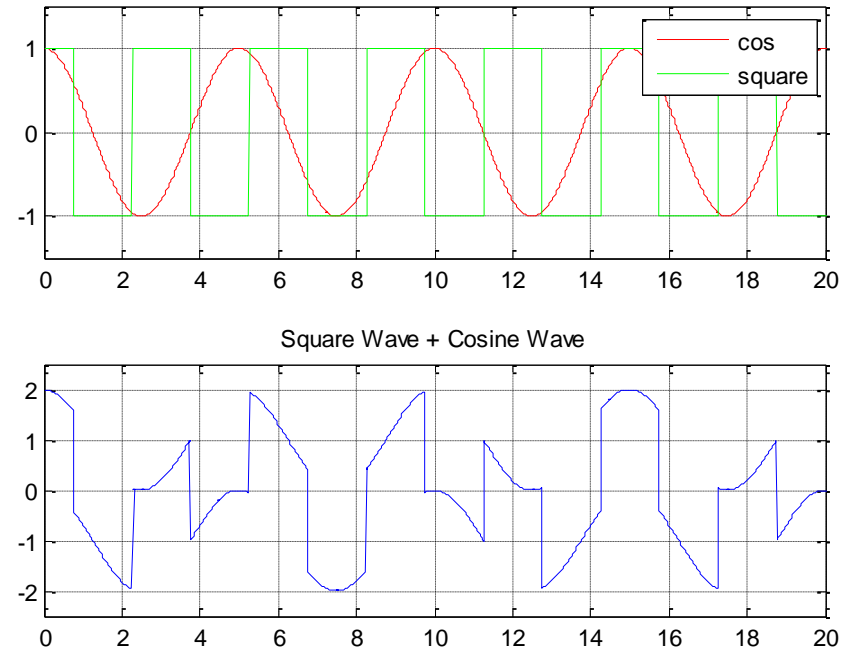
```
function y = heaviside(n)
    % We assume a vector input
    y = 0 * n; % Our default output value is 0
    % we find values in n greater than or equal to 0
    y(find(n >= 0)) = 1;
end
```


Example

- Generate a cosine signal of period 5, for time values $t=0:0.01:30$.
- Generate a signal of square wave of period 3 using the “cos” functions.
- Add the two signals generated before (“cos” signal and “square wave”), and verify that resultant period from the addition of two signals is normally equal to least common multiple of the two original periods (i.e., the least common multiple produces the maximum value possible for the resultant period).

```

t=0:0.01:30;
T_cosine=5; %period of cosine wave
T_square=3; %period of square wave
%% cosine wave with period of 5
y1=cos(2*pi/T_cosine.*t);
%% square wave from cosine wave
y2=sign(cos(2*pi/T_square.*t));
%% summation of square wave and cosine wave
y3=y1+y2;
subplot(211);plot(t,y1, 'r', t,y2, 'g');grid
legend('cos','square');axis([0,20,-1.5,1.5]);
subplot(212);plot(t,y3); grid on
title('Square Wave + Cosine Wave');
axis([0,20,-2.5,2.5]);
    
```



Exercise

- a) A signal in the form of $e^{j\omega_0 t} = e^{j\frac{2\pi}{T}t}$ is a periodic complex signal of period T . In order to visualize this well, use the following Matlab code to generate a signal with a period $T = 10$, posted on three periods with $t=0:0.01:3*T$.

Hint: Remember complex exponential identity.

$$e^{jx} = \cos(x) + j \sin(x) \rightarrow \text{real}(e^{jx}) = \cos(x) \quad \& \quad \text{imag}(e^{jx}) = \sin(x).$$

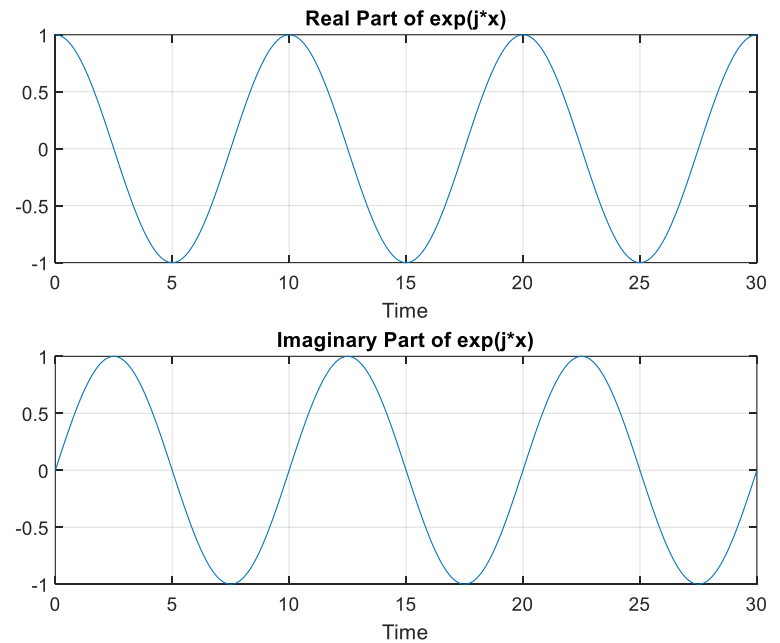
- b) Post up separately the real and imaginary part of this signal (as a time function, with “subplot” and “plot”) and verify that they match well to a cosine and sine signal respectively.

Start Now ...



```

clear all; close all;
%-----a)-----
---
% generate the complex function exp_t
T=10;
t1=0:0.01:3*T;
exp_t=cos(2*pi/T.*t1)+1i.*sin(2*pi/T.*t1)
;
%-----b)-----
----
figure(1)
subplot(2,1,1) % plot real part
plot(t1,real(exp_t)); grid on
xlabel('Time');
title('Real Part of exp(j*x)');
subplot(2,1,2) % plot imaginary part
plot(t1,imag(exp_t)); grid on
xlabel('Time');
title('Imaginary Part of exp(j*x)');
    
```



Real part = cosine

Imaginary part = sine

The END