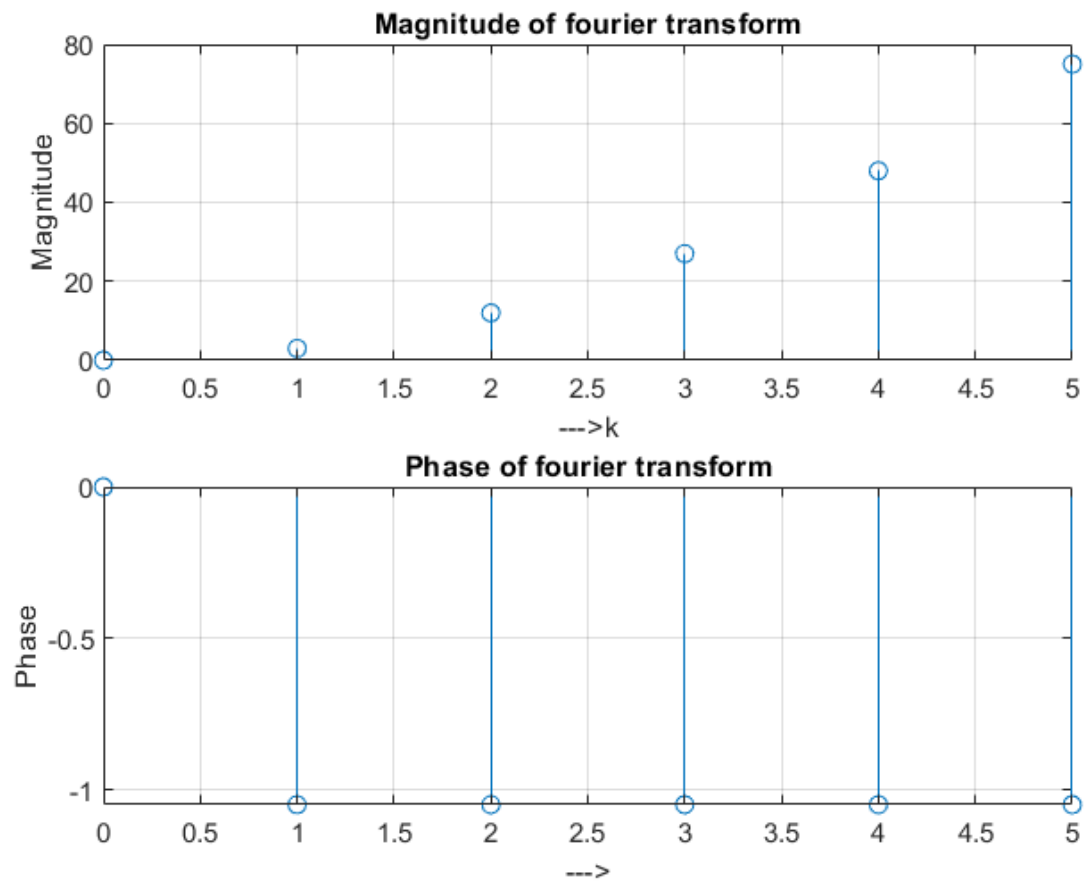


Source Code:

```
%DfT of a sequence
clc;
close all;
N=input('Length of sequence=');
x=input('The Sequence=');
n=0:1:N-1;
k=0:1:N-1;
wN=exp(-1i*2*pi/N);
nk=n.*k;
wNnk=wN*nk;
xk=x*wNnk;
disp('xk=');
disp(xk);
mag=abs(xk);
subplot(2,1,1);
stem(k,mag);
grid on;
xlabel('--->k');ylabel('Magnitude');
title('Magnitude of fourier transform');
phase=angle(xk);
subplot(2,1,2);
stem(k,phase);
grid on;
xlabel('--->');ylabel('Phase');
title('Phase of fourier transform');
```

Output :



Source Code:

```
clc;
close all;
%figure(1);
x = [1 2 3 4 5 6 7 6 5 4 3 2 1];
n = -2:10;

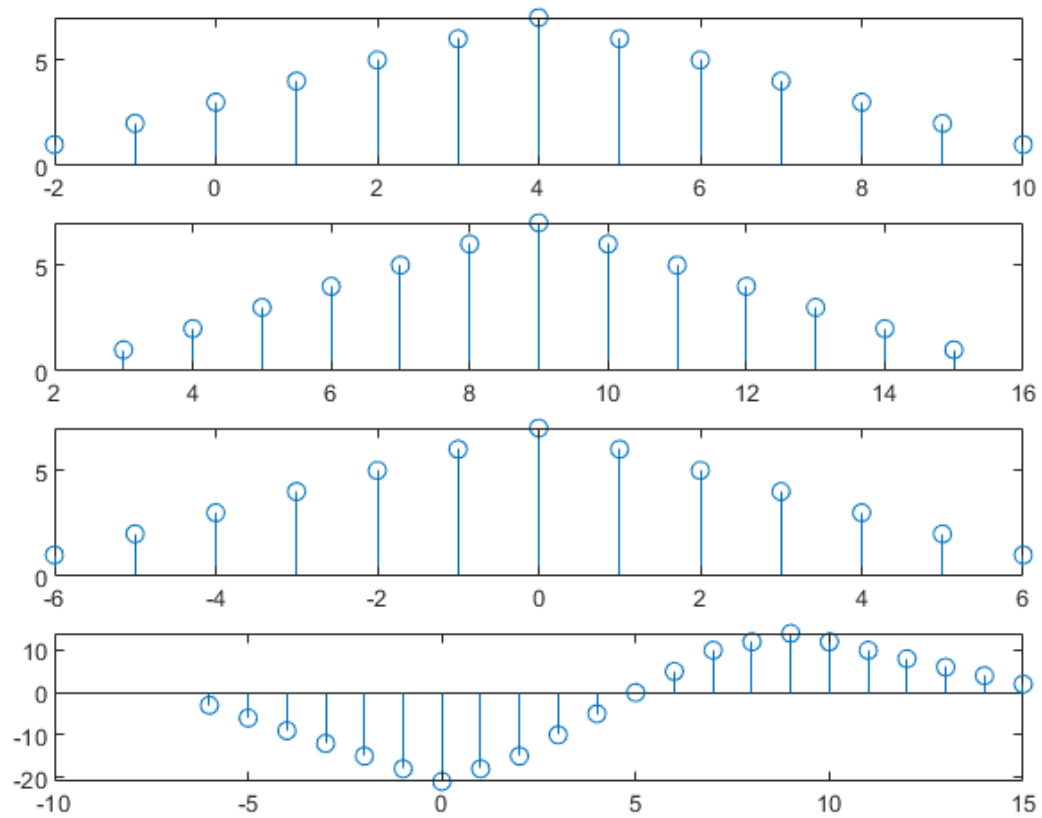
subplot(4,1,1);
stem(n,x);
%figure(2);
n1 = 3:15;
%a = n+n1;
subplot(4,1,2);
stem(n1,x);
%figure(3);
n2 = -6:6;
%b = n-n2;
subplot(4,1,3);
stem(n2,x);
m = min(min(n1),min(n2)):max(max(n1),max(n2));
y1 = [];
temp = 1;
for i = 1:length(m)
    if (m(i)<min(n1)||m(i)>max(n1))
        y1 = [y1 0];
    else
        y1 = [y1 x(temp)];
        temp = temp+1;
    end
end
y2 = [];
temp = 1;
for i = 1:length(m)
    if (m(i)<min(n2)||m(i)>max(n2))
        y2 = [y2 0];
    else
        y2 = [y2 x(temp)];
```

```

        temp = temp+1;
    end
end
y = (2.*y1)-(3.*y2);
subplot(4,1,4);
stem(m,y);

```

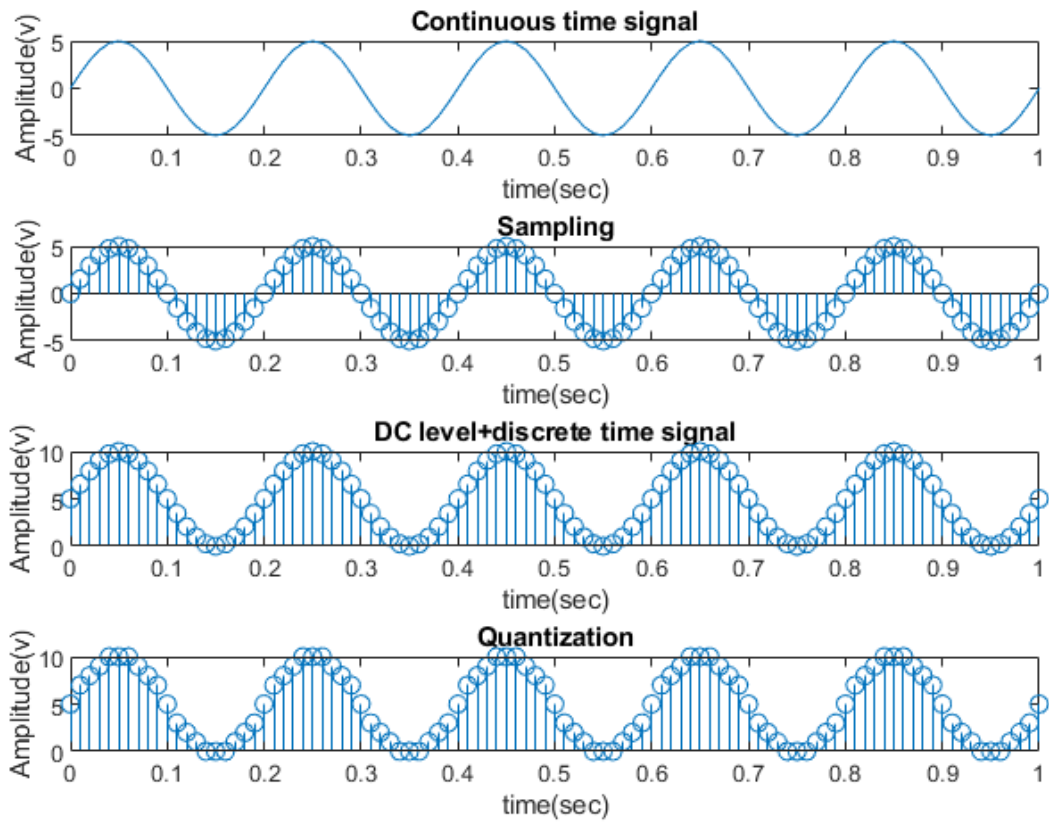
Output:



Source Code:

```
clc;
close all;
close all;
A = 5;
f = 5;
t = 0:0.01:1;
x = A*sin(2*pi*f*t);
subplot(4,1,1);
plot(t,x);
title('Continuous time signal');
xlabel('time(sec)');
ylabel('Amplitude(v)');
%%After sampling discrete time signal
subplot(4,1,2);
stem(t,x);
title('Sampling');
xlabel('time(sec)');
ylabel('Amplitude(v)');
%%Dc level+discrete time signal
x1 = A+x;
subplot(4,1,3);
stem(t,x1);
title('DC level+discrete time signal');
xlabel('time(sec)');
ylabel('Amplitude(v)');
%%Quantized
x2 = round(x1);
subplot(4,1,4);
stem(t,x2);
title('Quantization');
xlabel('time(sec)');
ylabel('Amplitude(v)');
%%coding
x3 = dec2bin(x2);
disp(x3);
```

Output:

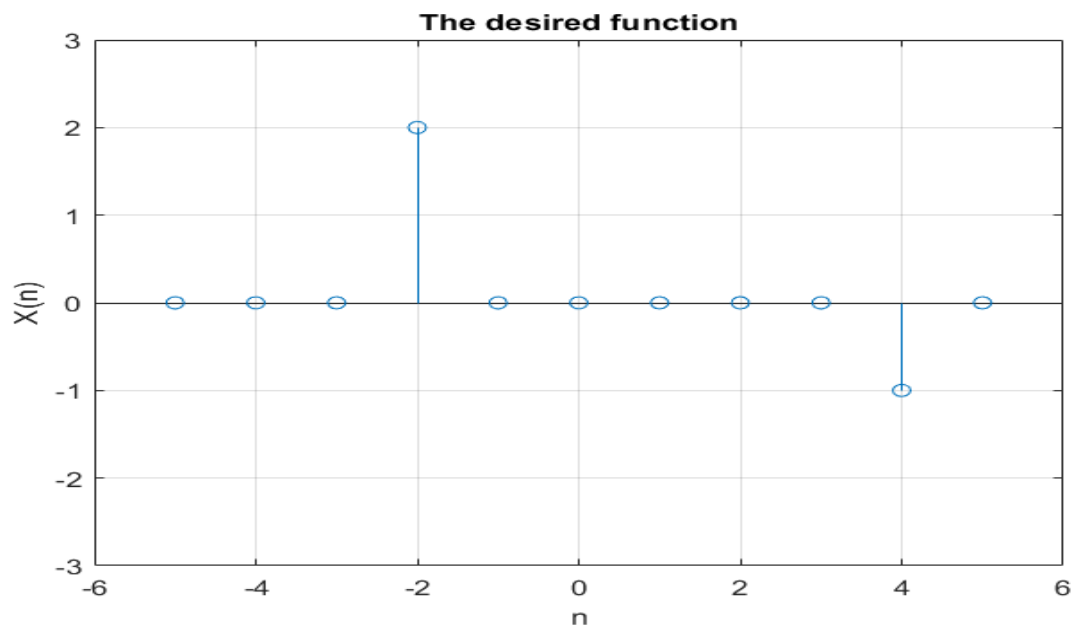


Source Code:

```
clc;  
close all;  
n=-5:5;  
x=2*deltaF(-2,-5,5)-deltaF(4,-5,5);  
stem(n,x);  
xlabel('n');  
ylabel('X(n)');  
title('The desired function');  
axis([-6 6 -3 3]);  
grid on;
```

```
function [x,n]=deltaF(n0,n1,n2)  
n=n1:n2;  
x=(n-n0)==0;  
end
```

Output:



Source Code:

```
clc;
close all;
figure(1);
x = [1 0 3 4];
n1 = -2:1;
subplot(3,1,1);
stem(n1,x);
grid on;
title('X=');
xlabel('n');
ylabel('x(n)');
axis([-3 3 0 5]);

y = [1 1 1 1];
n2 = 0:3;
subplot(3,1,2);
stem(n2,y);
grid on;
title('Y=');
xlabel('n');
ylabel('x(n)');
axis ([-3 5 0 5]);

m = min(min(n1),min(n2)):max(max(n1),max(n2));
y1 = [];
temp = 1;
for i = 1:length(m)
    if(m(i)<min(n1)||m(i)>max(n1))
        y1 = [y1 0];
    else
        y1 = [y1 x(temp)];
        temp = temp+1;
    end
end
y2 = [];
temp = 1;
for i = 1:length(m)
```



```

        if(m(i)<min(n2)||m(i)>max(n2))
            y2 = [y2 0];
        else
            y2 = [y2 y(temp)];
            temp = temp+1;
        end
    end
end
add = y1+y2;
subplot(3,1,3)
stem(m,add);
grid on;
title('Addition of signals(X+Y)');
xlabel('n');
ylabel('x(n)+y(n)');
axis ([-3 5 0 7]);

```

```

figure(2);
x = [3 -1 0 -4];
n = -1:2;
subplot(2,1,1);
stem(n,x);
title('Original signal x(n)');
xlabel('n');
ylabel('x(n)');
axis([-2 3 -5 4]);

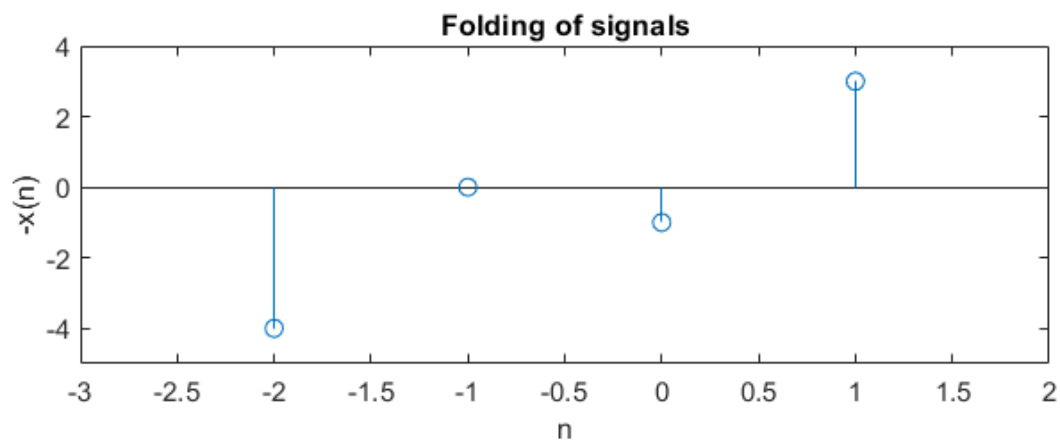
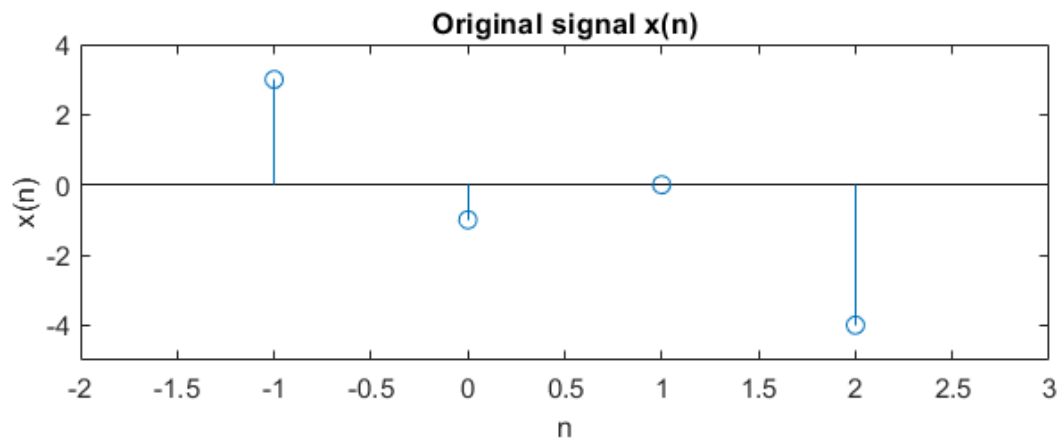
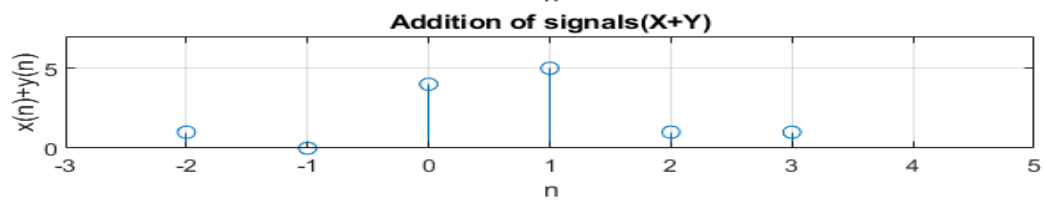
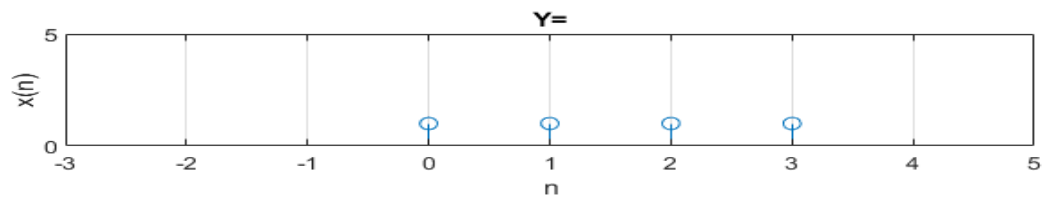
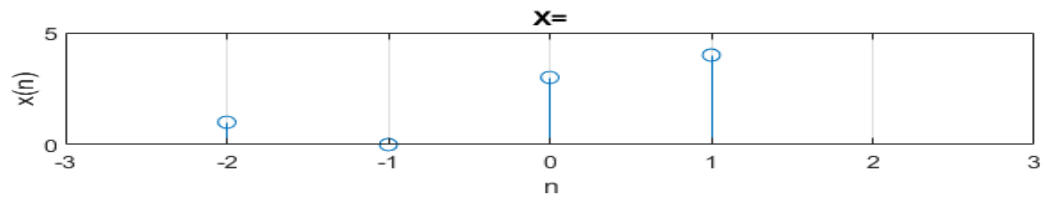
```

```

c =fliplr(x);
y = fliplr(-n);
subplot(2,1,2)
stem(y,c);
title('Folding of signals');
xlabel('n');
ylabel('-x(n)');
axis([-3 2 -5 4]);

```

Output:



Source Code:

```
clc;
close all;
figure(1);
x = [1 2 3 4];
n1 = -2:1;
subplot(3,1,1);
stem(n1,x);
xlabel('n');
ylabel('x1(n)');
title('X1(n)');
axis([-8 10 -2 5]);
grid on;

y = [1 1 1 1];
n2 = 0:3;
subplot(3,1,2);
stem(n2,y);
xlabel('n');
ylabel('x2(n)');
title('X2(n)');
axis([-8 10 -2 5]);
grid on;

m = min(min(n1),min(n2)):max(max(n1),max(n2));
y1 = [];
temp = 1;
for i = 1:length(m)
    if(m(i)<min(n1)||m(i)>max(n1))
        y1 = [y1 0];
    else
        y1 = [y1 x(temp)];
        temp = temp+1;
    end
end
y2 = [];
```

```

temp = 1;
for i = 1:length(m)
    if(m(i)<min(n2)||m(i)>max(n2))
        y2 = [y2 0];
    else
        y2 = [y2 y(temp)];
        temp = temp+1;
    end
end
mul = y1.*y2;
subplot(3,1,3);
stem(m,mul);
xlabel('n');
ylabel('X1(n)*X2(n)');
title('Signal Multiplication');
axis([-8 10 -2 5]);
grid on;

figure(2);
n = -2:2;
x = [-2 3 0 1 5];
subplot(3,1,1);
stem(n,x);
xlabel('n');
ylabel('x1(n)');
title('X1(n)');
axis([-8 10 -4 6]);
grid on;

n1 = 5;
a = n+n1;
subplot(3,1,2);
stem(a,x);
xlabel('n');
%ylabel('x2(n)');
title('X1(n-5)');
axis([-8 10 -4 6]);
grid on;

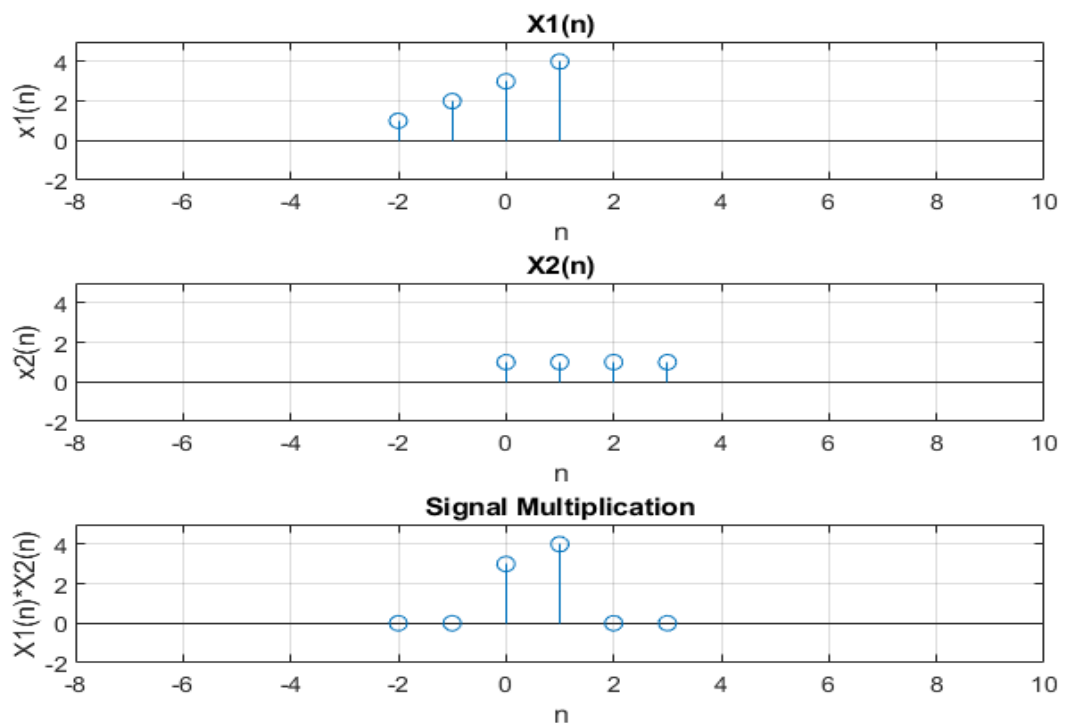
```

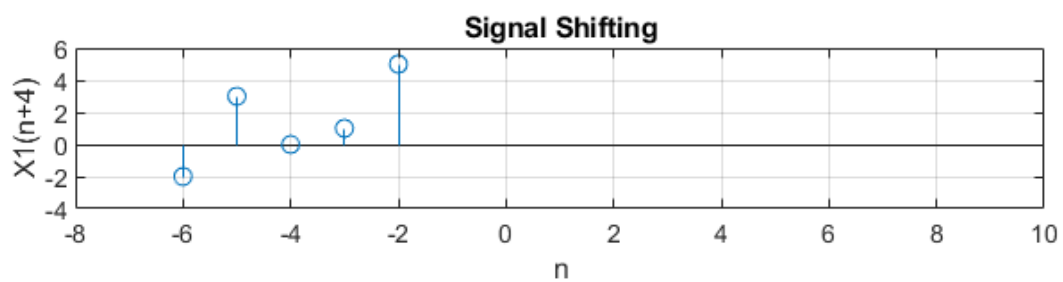
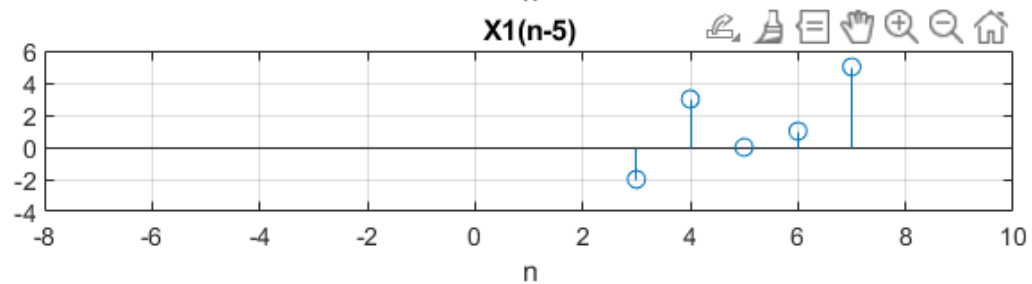
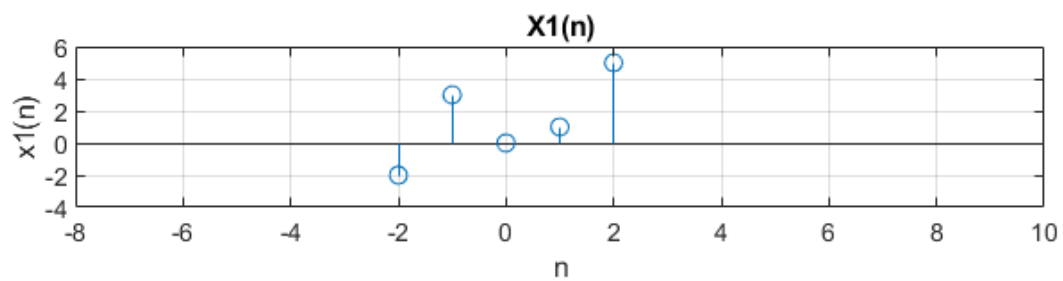
```

n2 = 4;
b = n-n2;
subplot(3,1,3);
stem(b,x);
xlabel('n');
ylabel('X1(n+4)');
title('Signal Shifting');
axis([-8 10 -4 6]);
grid on;

```

Output:

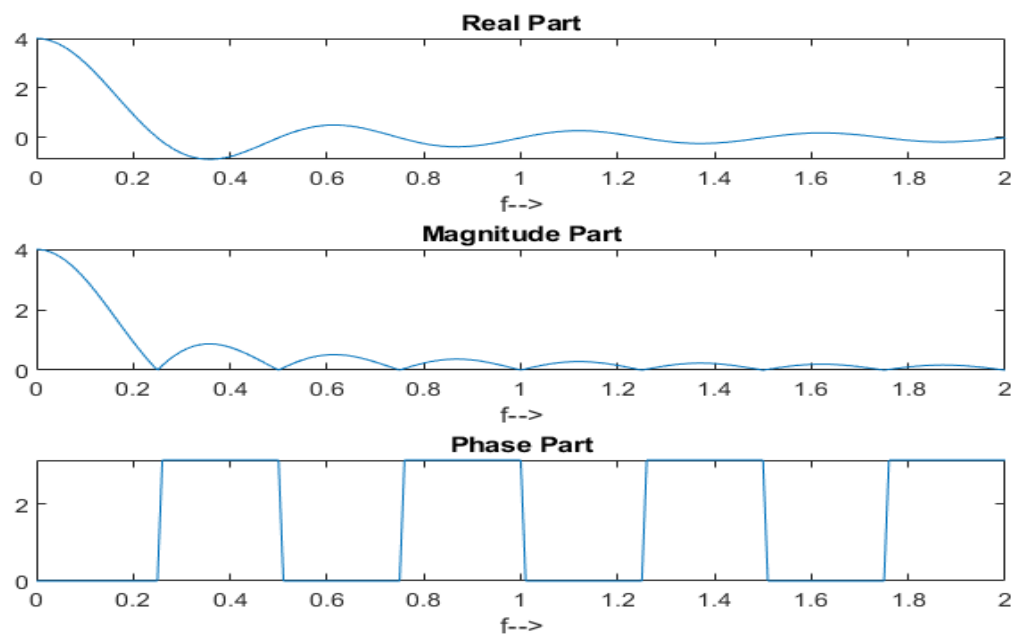




Source Code:

```
clc;
close all;
f=0:0.01:2;
x=4*sinc(4*f);
figure(1);
subplot(3,1,1);
plot(f,real(x));
title('Real Part');
xlabel('f-->');
subplot(3,1,2);
plot(f,abs(x));
title('Magnitude Part');
xlabel('f-->');
subplot(3,1,3);
plot(f,angle(x));
title('Phase Part');
xlabel('f-->');
```

Output:

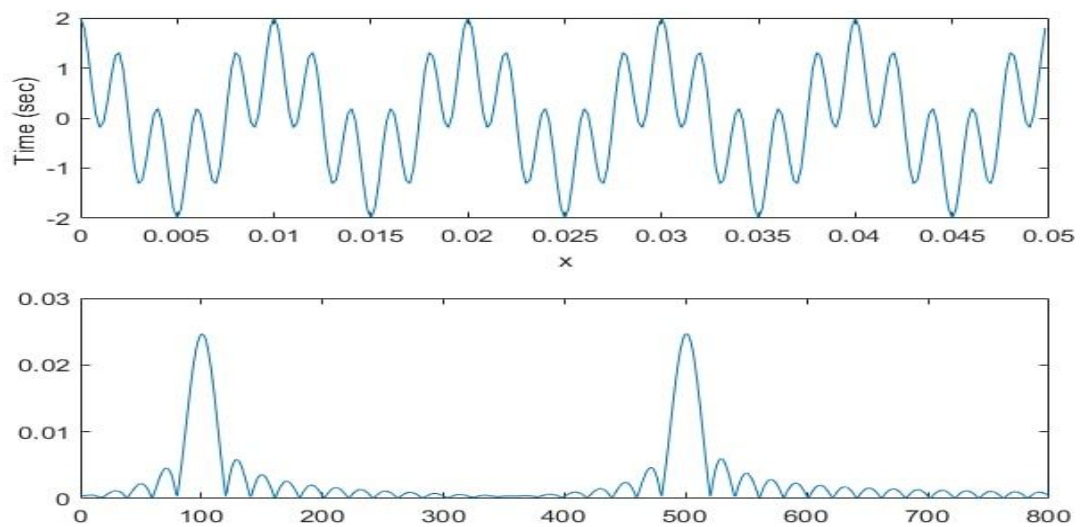


Source Code:

```
clc;
close all;
N=250;
ts=.0002;
t=(0:N-1)*ts;
x=cos(2*pi*100*t)+cos(2*pi*500*t);
subplot(2,1,1)
plot(t,x)
xlabel("x");
ylabel("Time (sec) ");

k=0;
for f=0:1:800
    k=k+1;
    X(k)=trapz(t,x.*exp(-1i*2*pi*f*t));
end
f=0:800;
subplot(2,1,2);
plot(f, abs(X));
```

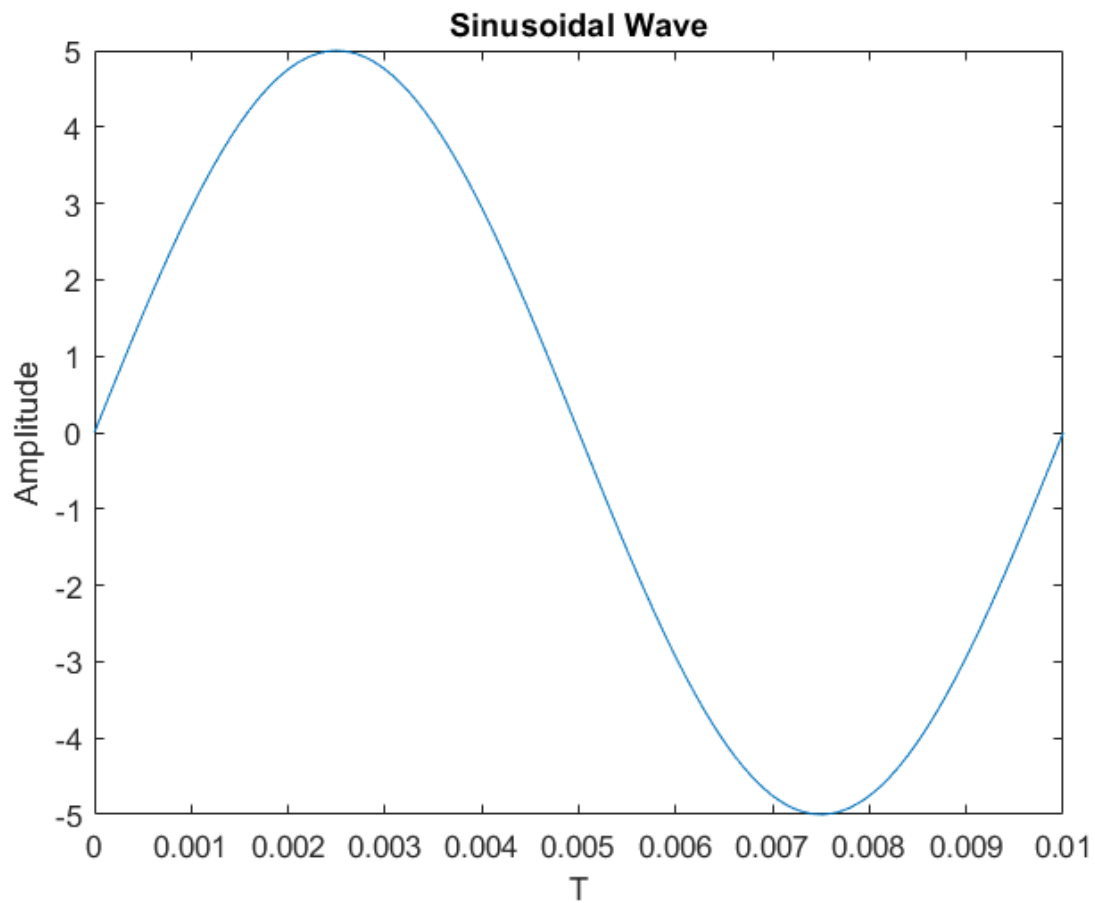
Output:



Source Code:

```
clc;  
close all;  
A=5;  
f=100;  
T=1/f;  
t=0:T/100:T;  
y=A*sin(2*pi*f*t);  
plot(t,y);  
xlabel('T');ylabel('Amplitude');  
title('Sinusoidal Wave');
```

Output:



Source Code:

```
% Unit sample sequence
clc;
close all;
n=-5:5;
x=n==0;
stem(n,x);
subplot(3,1,1);
title('Unit Sample Sequence');
xlabel('n');ylabel('x');

%unit step signal

N=5;
n2=-N:1:N;
x2=[zeros(1,N) 1 ones(1,N)];
subplot(3,1,2);
stem(n2,x2);
xlabel('Time');ylabel('Amplitude');
title('Unit Step');

%unit ramp function

n3=-5:5;
x3=n3.*(n3>=0);
subplot(3,1,3);
stem(n3,x);
xlabel('Time');ylabel('Amplitude');
title('Unit ramp');
```

Output:

