

به نام خدا

نام: امیرمسعود

نام خانوادگی: شاکر

شماره دانشجویی: 97243081

تمرین سوم

- 1) در این سوال با استفاده از تابع CK ، ضرایب فوریه برای بازه ی 20- تا 20 محاسبه شده اند. سپس ضرایب محاسبه شده به عنوان ورودی به توابع آماده abs , angle پاس داده شده و بدین ترتیب magnitude, phase محاسبه شده اند. در نهایت با استفاده از دستور plot ، magnitude, phase نمایش داده شده اند.

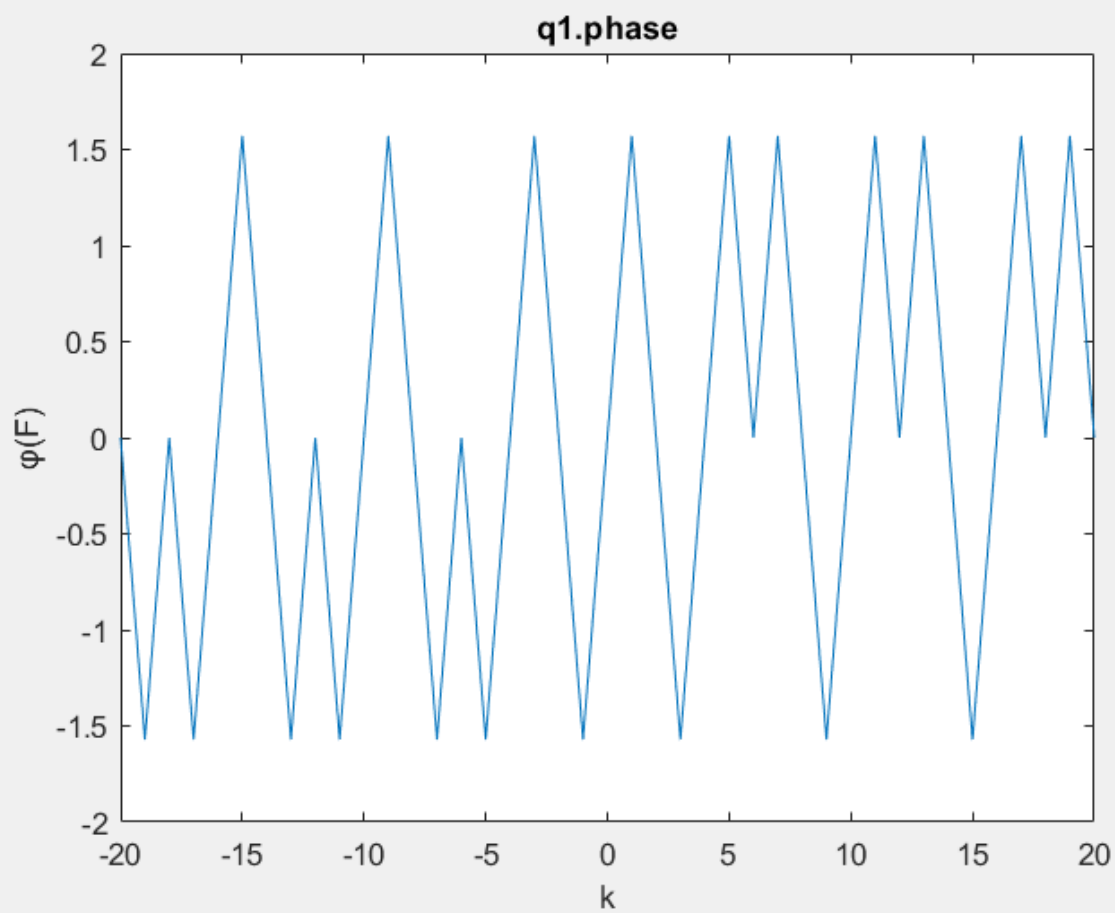
q_1:

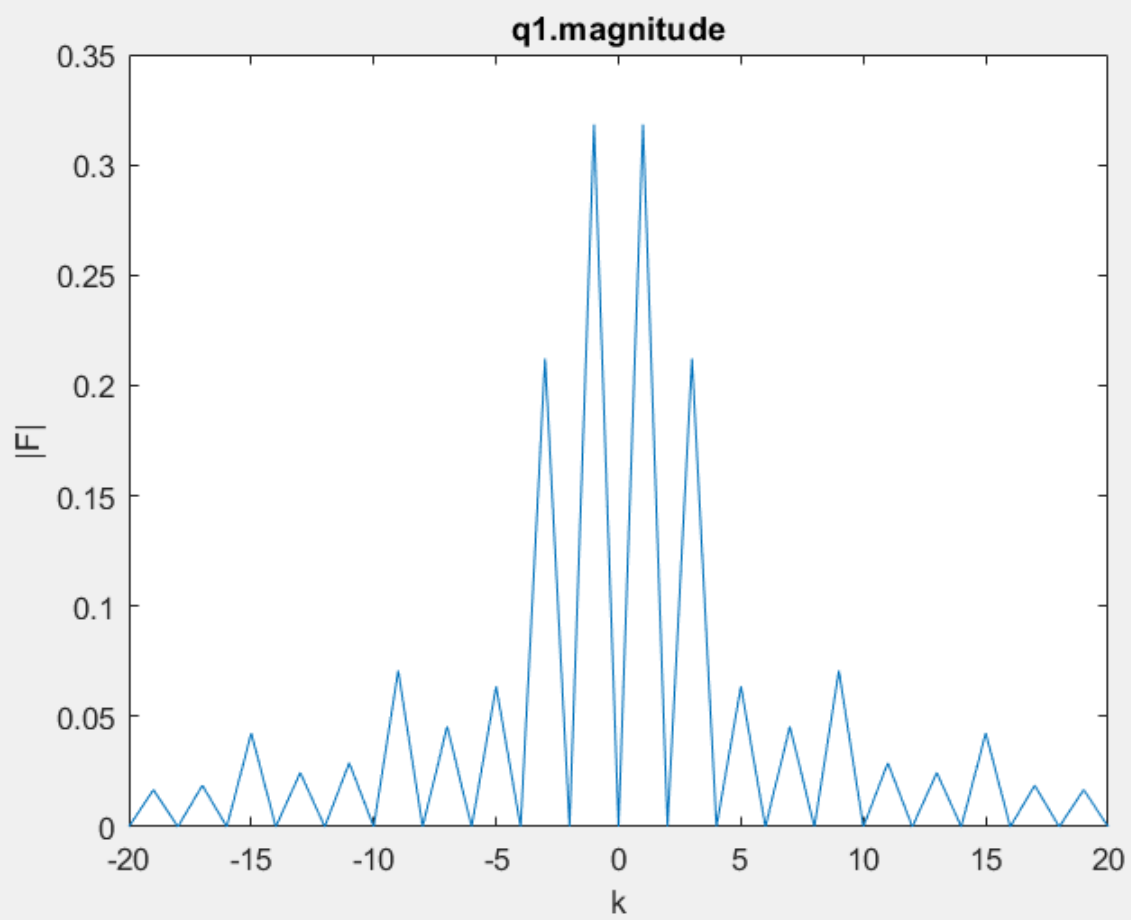
```
clc;
clear;

n = -20:20;
% create the range
fourier_coeff = CK(n);
% calculate the fourier series coefficients using CK function

figure(1);
% let us have more than one figure
fourier_phase = angle(fourier_coeff);
% calculate the phases using angle function
% with fourier series coefficients in input
plot(n, fourier_phase);
% show us the phases
xlabel('k');
ylabel('? (F) ');
title('q1.phase');

figure(2);
% let us have more than one figure
fourier_mag = abs(fourier_coeff);
% calculate the magnitudes using abs function
% with fourier series coefficients in input
plot(n, fourier_mag);
% show us the magnitudes
xlabel('k');
ylabel('|F| ');
title('q1.magnitude');
```





(2) در این سوال ابتدا تابع $xEstim$ بر اساس فرمول پیاده سازی شده است.

ورودی این تابع وکتور زمانی t و $Kmax$ بوده و خروجی این تابع تخمین مقدار $x(t)$ است.

$xEstim$:

```
function y = xEstim(t, Kmax)
    y = zeros(size(t));
    for m = 1:length(t)
        for k=-1*Kmax:Kmax
            y(m) = y(m) + (CK(k) * exp(1i*k*t(m)*pi/3));
        end
    end
end
```

سپس ابتدا وکتور زمانی t تعریف شده است.

با صدا زدن تابع $xEstim$ به ازای مقادیر مختلف $Kmax$ ، تخمین مقدار $x(t)$ به دست آمده است.

در نهایت با استفاده از تخمین به دست آمده، $magnitude, phase$ محاسبه و رسم شده اند.

q_2 :

```

clc;
clear;

t = -6:0.01:6;
% create the range
x_estim = xEstim(t,5);
% estimate x using xEstim function
% with Kmax = 5
mag = abs(x_estim);
% calculate magnitude of x_t using abs function
% for Kmax = 5
phase = angle(x_estim);
% calculate phase of x_t using angle function
% for Kmax = 5
figure(1);
% let us have more than one figure
plot(t, mag);
xlabel('t');
ylabel('|x(t)|');
title('q2.Kmax5');

figure(2);
plot(t, phase);
xlabel('t');
ylabel('?(x(t))');
title('q2.Kmax5');

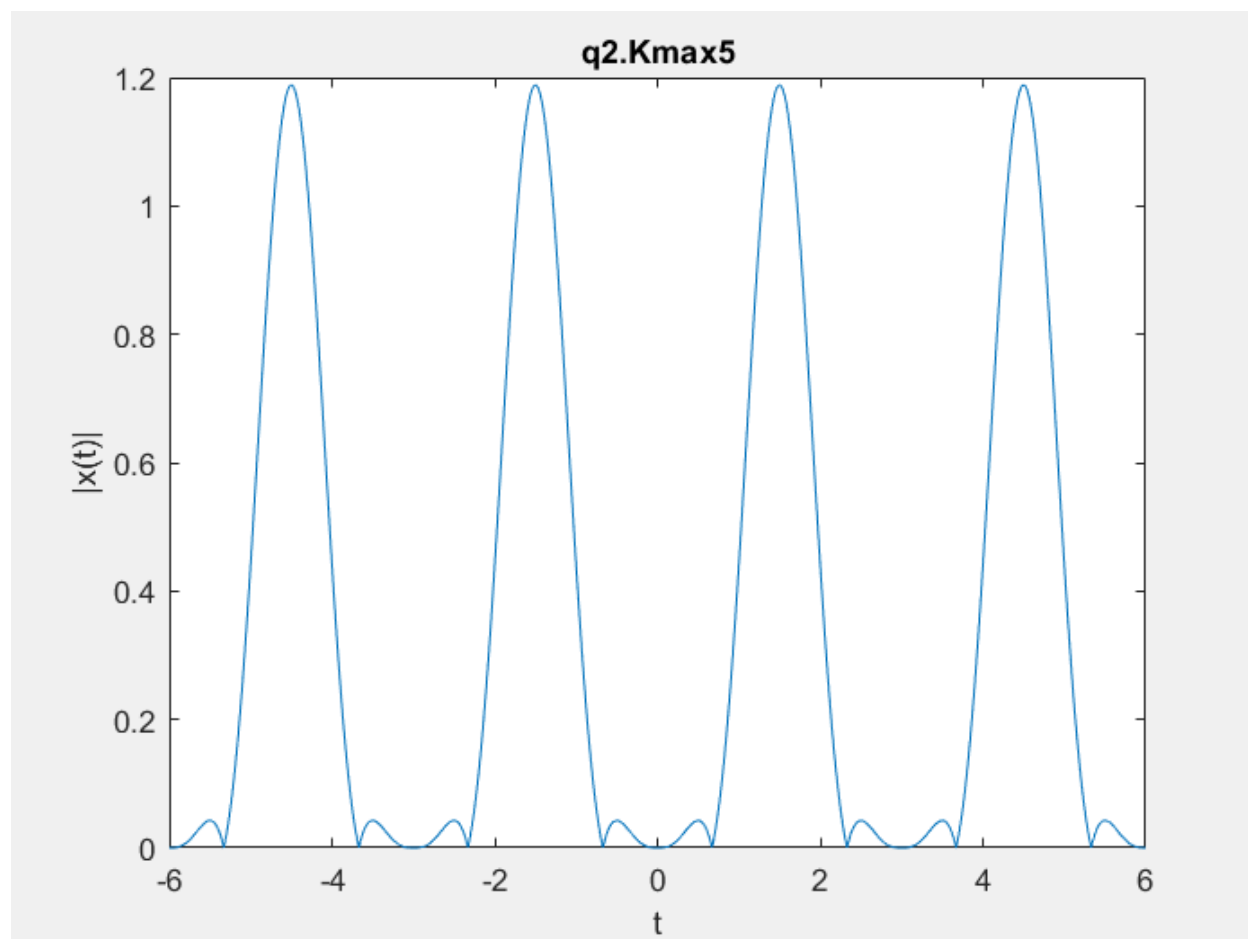
x_estim = xEstim(t,15);
% estimate x using xEstim function
% with Kmax = 15
mag = abs(x_estim);
% calculate magnitude of x_t using abs function
% for Kmax = 15
phase = angle(x_estim);
% calculate phase of x_t using angle function
% for Kmax = 15

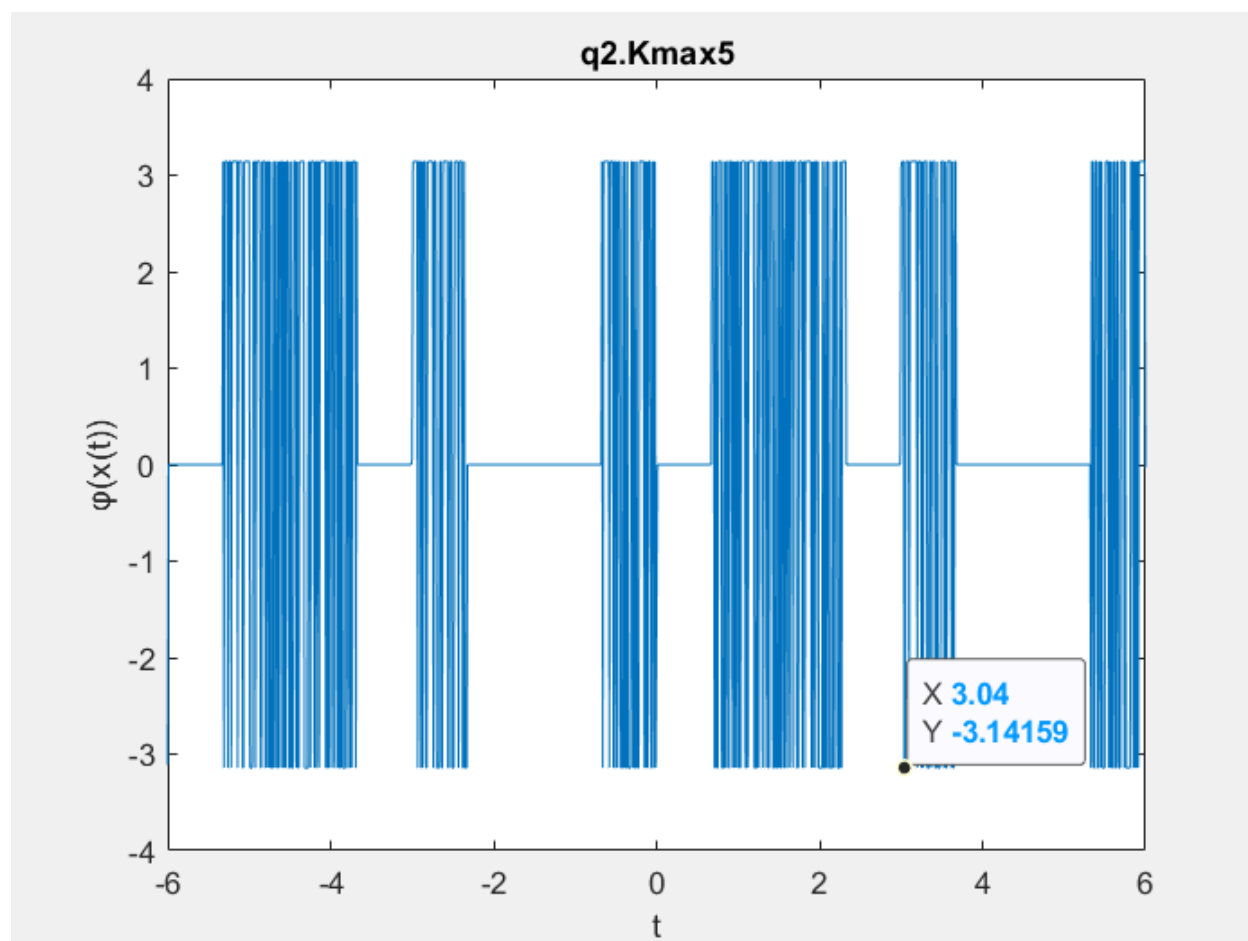
figure(3);
plot(t, mag);
title('q2.Kmax5');
xlabel('t');
ylabel('|x(t)|');

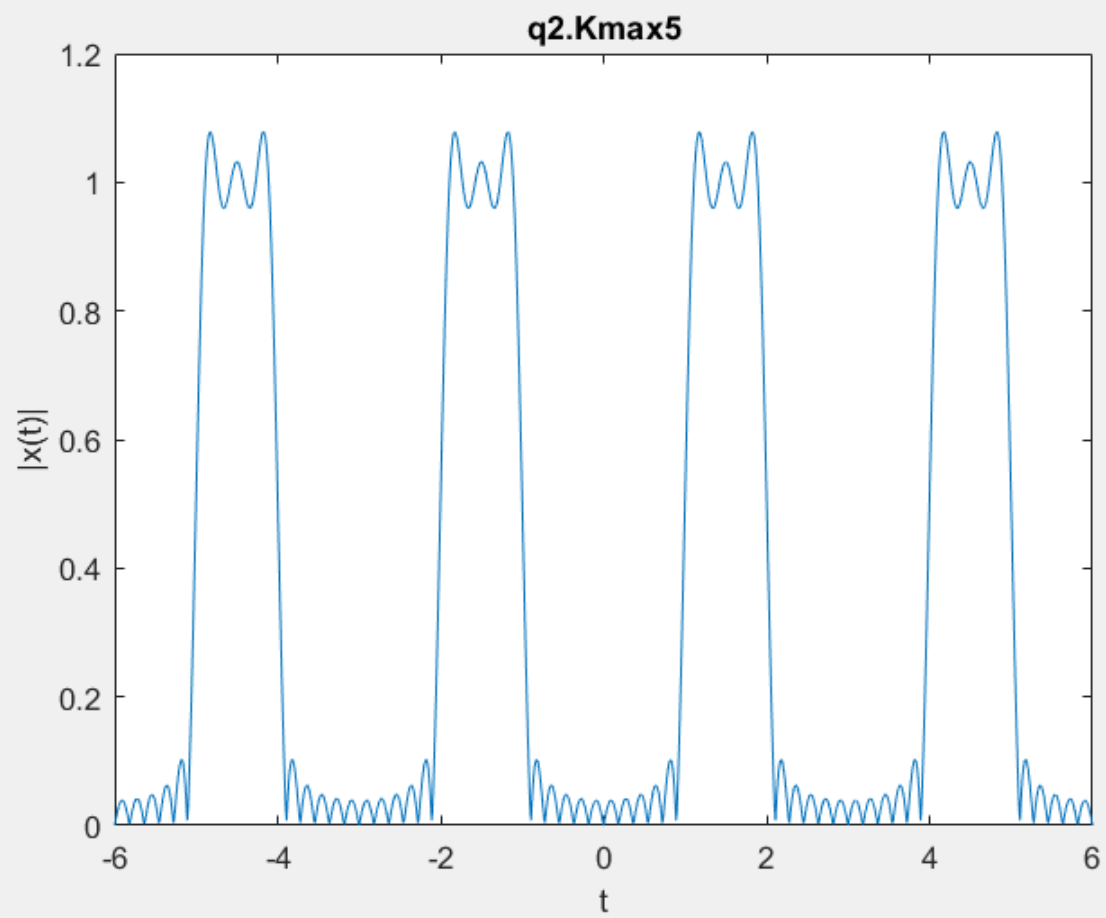
figure(4);
plot(t, phase);
xlabel('t');
ylabel('?(x(t))');
title('q2.Kmax15');

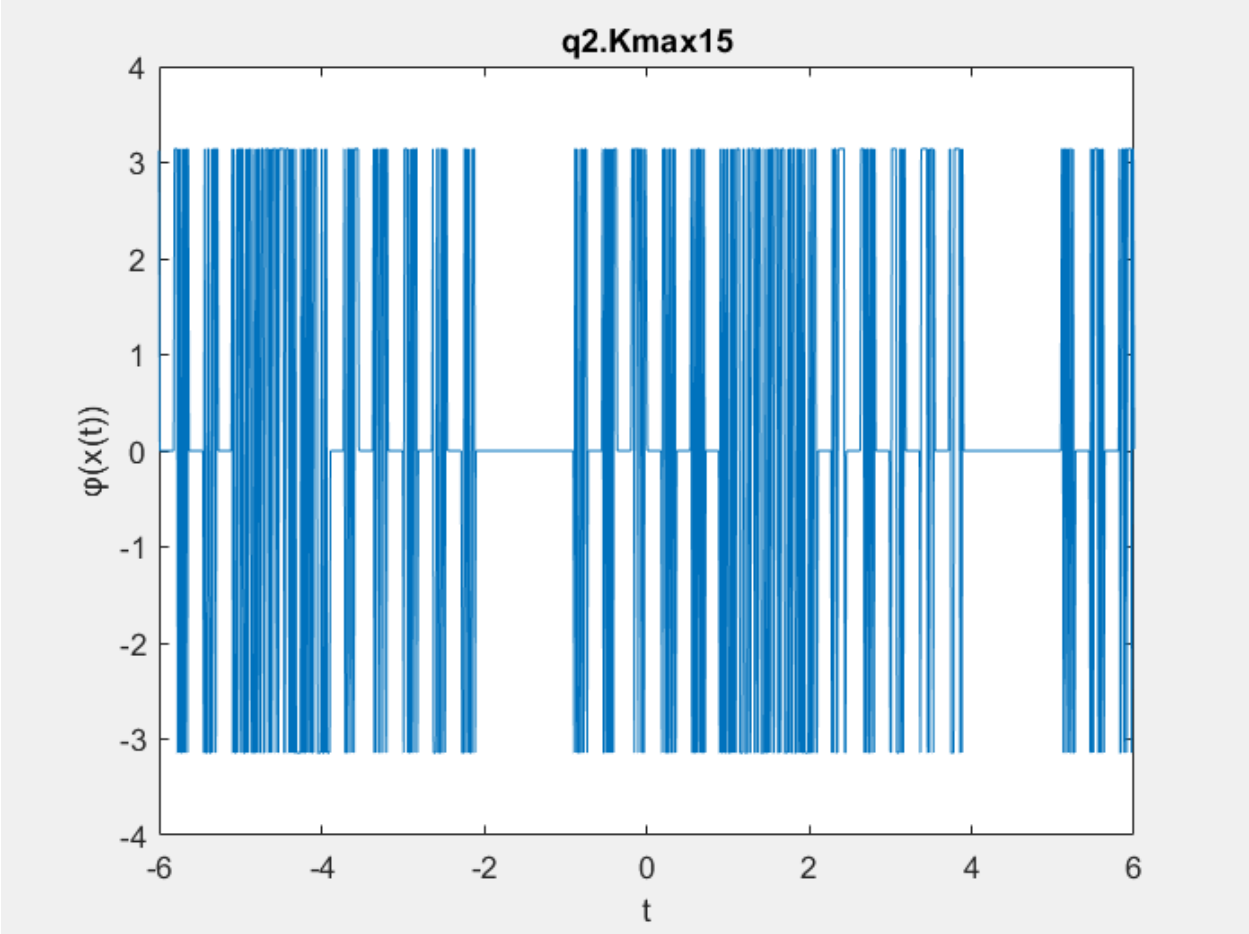
x_estim = xEstim(t,30);
% estimate x using xEstim function
% with Kmax = 30
mag = abs(x_estim);
% calculate magnitude of x_t using abs function
% for Kmax = 30
phase = angle(x_estim);
% calculate phase of x_t using angle function
% for Kmax = 30
figure(5);
plot(t, mag);
xlabel('t');
ylabel('|x(t)|');

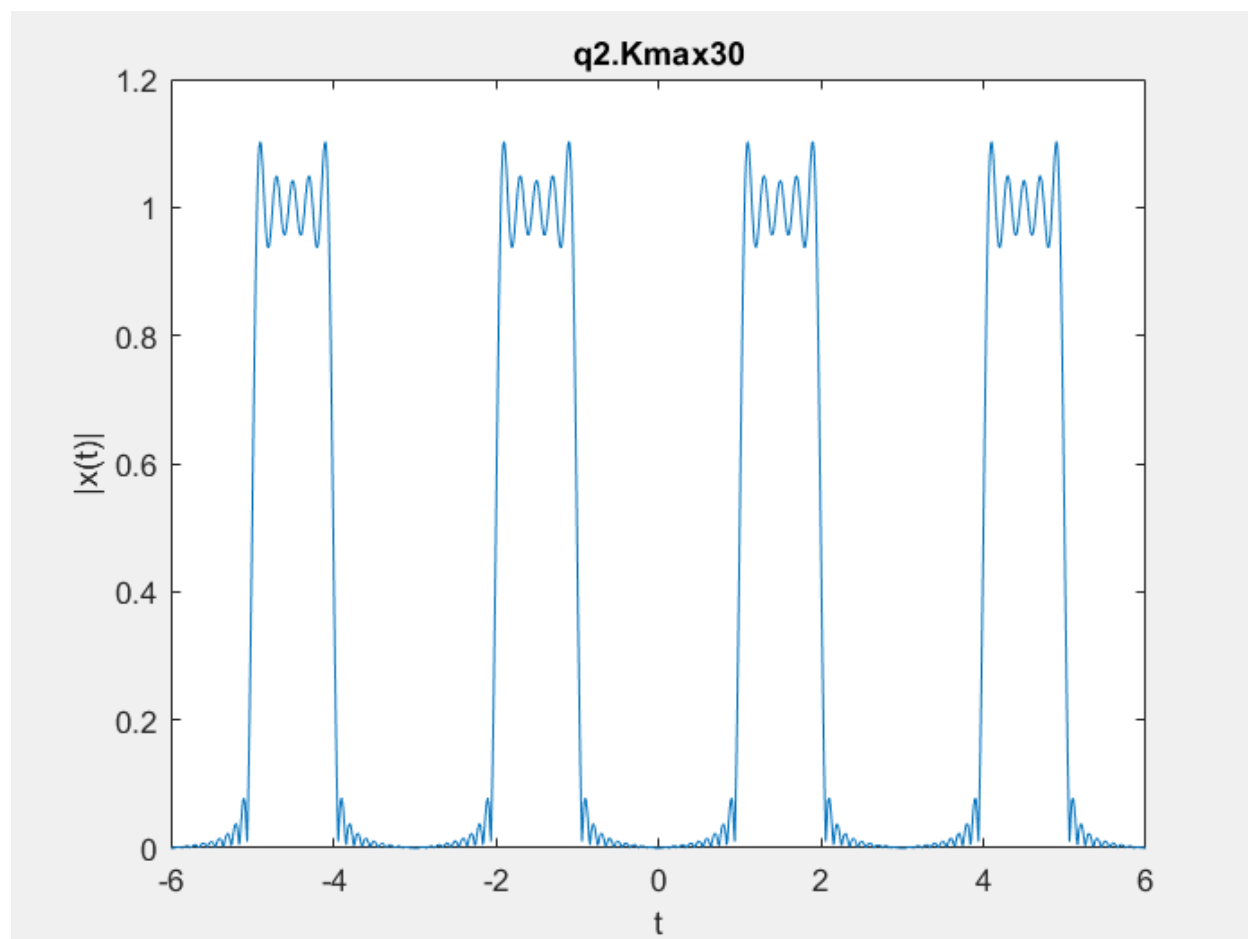
```

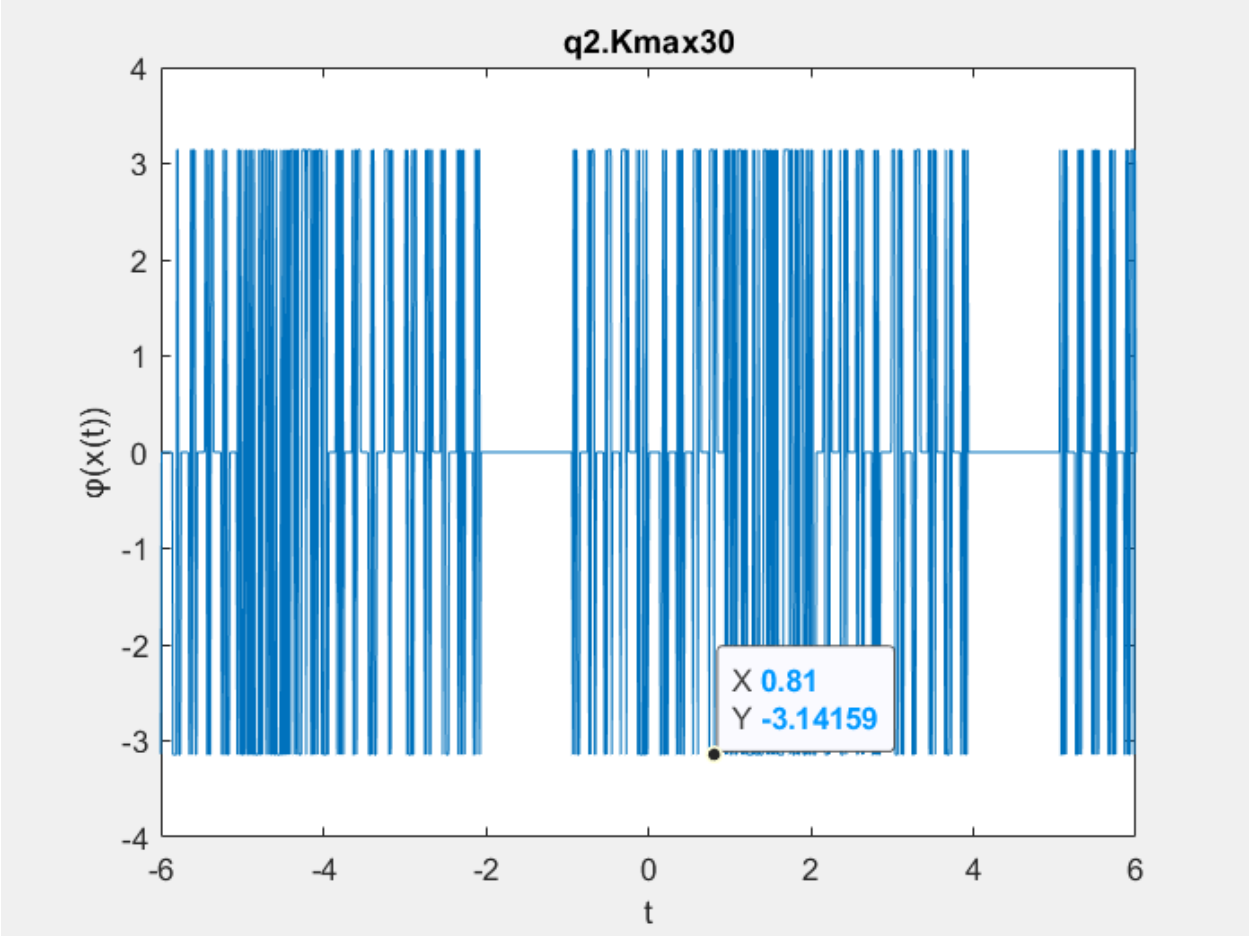












(3) در این سوال ابتدا تابع `fourier_transform` پیاده سازی شده است. این تابع تبدیل فوریه ی زمان گسسته ی سیگنال دریافت شده از ورودی را به کمک تابع آماده `symsum` محاسبه میکند.

`fourier_transform`:

```
function F_T = fourier_transform(func, range, start_p, end_p)
syms w
% define symbolic variable w
syms n
% define symbolic variable n

ft(w) = symsum(func * exp(-1i*w*n), n, start_p, end_p);
F_T = ft(range);
end
```

سپس سیگنالهای `x1`, `x2`, `x3` با توجه به فرمول هایشان تعریف شده اند و با استفاده از تابع `fourier_transform` نوشته شده، تبدیل فوریه ی هر کدام در بازه ی مد نظر محاسبه شده است.

در آخر اندازه و فاز تبدیل فوریه ی هر کدام از سیگنالها با استفاده از دستور `stem` رسم شده است.

q_3:

```
clc;
clear;

syms n
% define symbolic variable x

x1(n)=piecewise(n<0, 0, (4>=n)&(n>=0), 1, n>4, 0);
% create x1(n) function according to its formula
x2(n)=piecewise(n~=0, sin(n)*sin(2*n)/(n*pi)^2, 2/pi^2);
% create x2(n) function according to its formula
x3(n)=sin(100*n) * exp(-1i*pi*n);
% create x3(n) function according to its formula

n_range=-50:50;

FT1 = fourier_transform(x1(n), n_range, 0, 4);
% compute the fourier transform of x1(n) using the function written
FT2 = fourier_transform(x2(n), n_range, -50, 50);
% compute the fourier transform of x2(n) using the function written
FT3 = fourier_transform(x3(n), n_range, -50, 50);
% compute the fourier transform of x3(n) using the function written

n_range_2 = [-20:20];

figure(1)
% let us have more than one figure
subplot(2, 1, 1)
stem(n_range, abs(FT1));
title('a.magnitude');

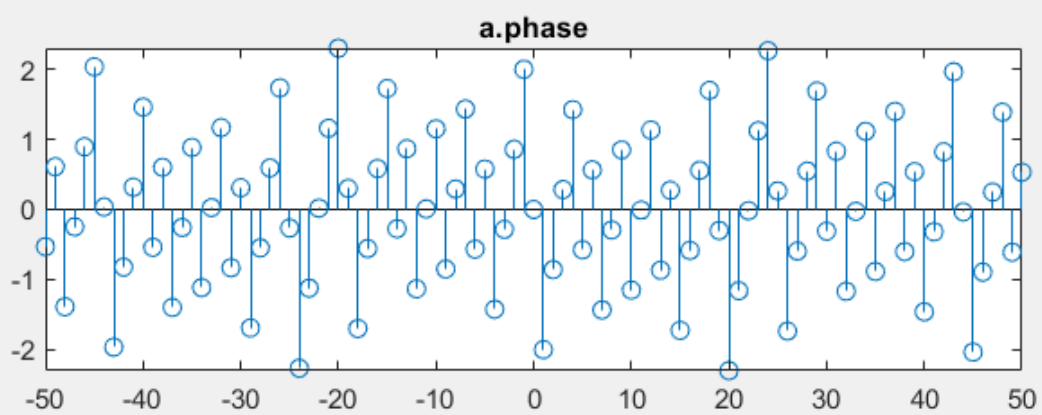
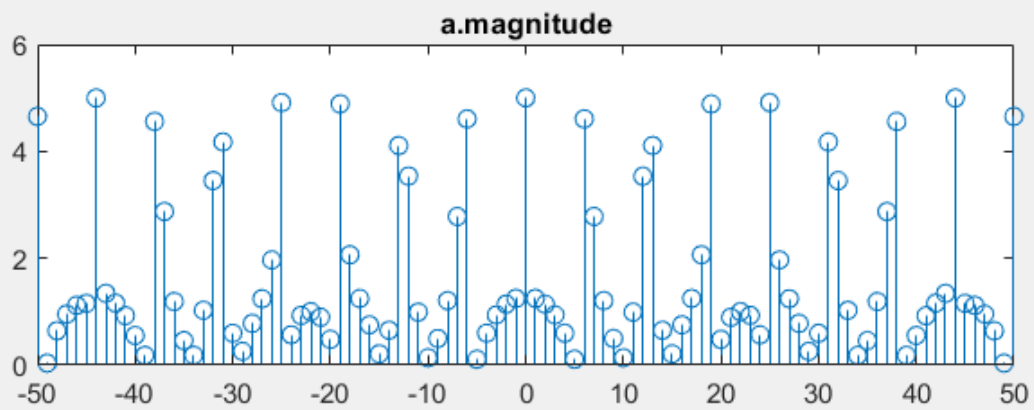
subplot(2, 1, 2)
stem(n_range, angle(FT1));
title('a.phase');

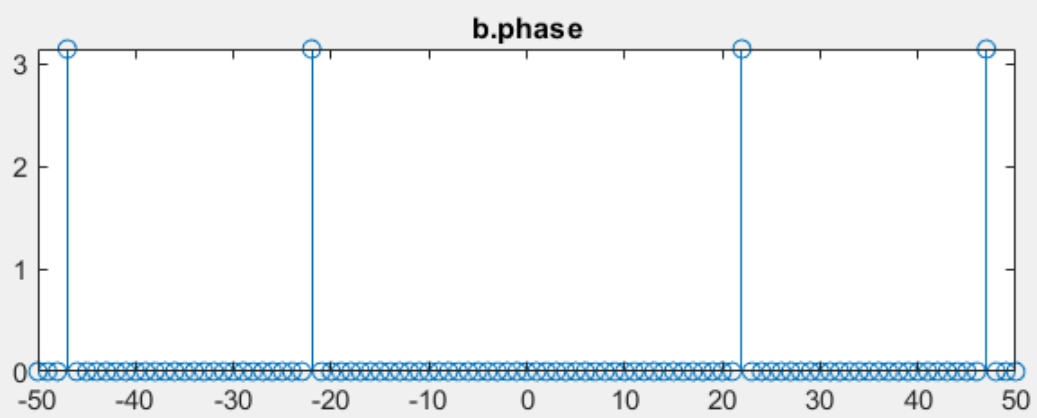
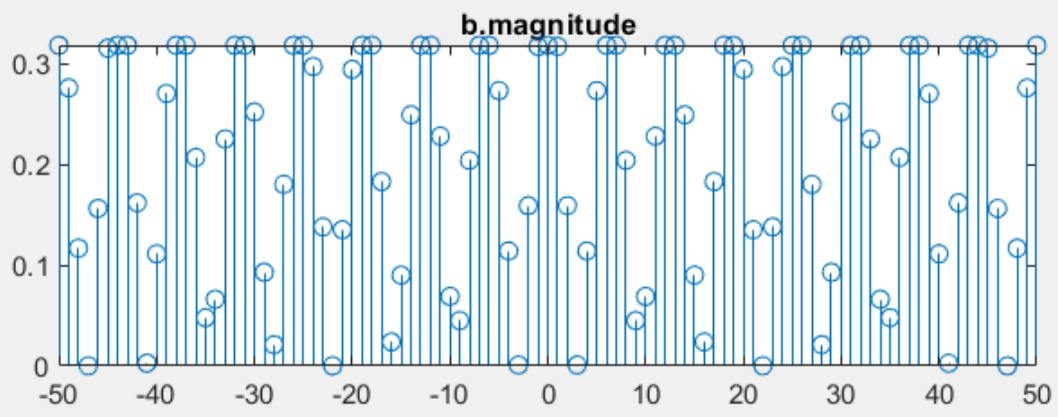
figure(2)
% let us have more than one figure
subplot(2, 1, 1)
stem(n_range, abs(FT2));
title('b.magnitude');

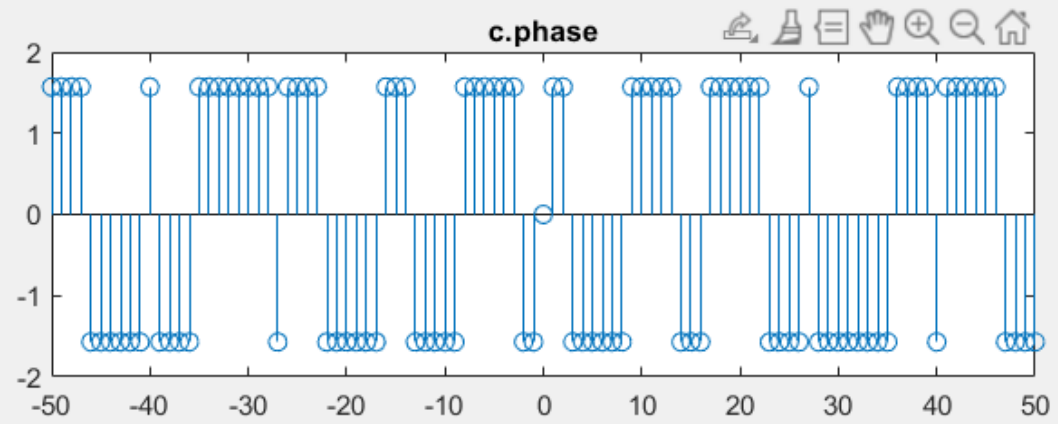
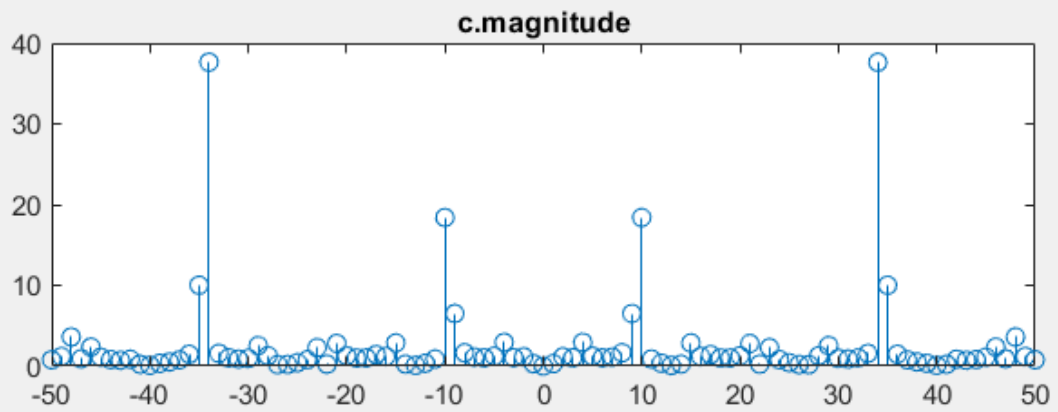
subplot(2, 1, 2)
stem(n_range, angle(FT2));
title('b.phase');

figure(3)
% let us have more than one figure
subplot(2, 1, 1)
stem(n_range, abs(FT3));
title('c.magnitude');

subplot(2, 1, 2)
stem(n_range, angle(FT3));
title('c.phase');
```







4) در این سوال ابتدا دو تابع به منظور محاسبه ضرایب سری فوریه سیگنال ورودی نوشته شده است. یکی برای اولین ضریب (C_0) و دیگری برای سایر ضرایب.

پیاده سازی توابع بر اساس فرمول ضرایب سری فوریه بوده و از تابع آماده `int` به منظور انتگرال گیری استفاده شده است.

سپس سیگنال $x(t)$ پیاده سازی شده و با استفاده از توابع `C0_periodic`, `Cn_periodic` تعریف شده، ضرایب سری فوریه سیگنال $x(t)$ محاسبه شده و با استفاده از آن، اندازه و فاز، محاسبه و با دستور `plot` رسم شده اند.

C0_periodic:

```
function C0 = C0_periodic(f,T)
syms x
% define symbolic variable x
C0 = int(f, x, -T/2, T/2) / T;
% calculate coefficient C0
end
```

Cn_periodic:

```
function Cn = Cn_periodic(f,T)
syms n
% define symbolic variable n
syms x
% define symbolic variable x
Cn = int(f * exp(-2i*x*n*(pi/T)), x, -T/2, T/2) / T;
% calculate fourier series coefficients of a periodic signal
% according to formula
% instead of C0
end
```

q_4:

```
clc;
clear;

t1 = -20:-1;
% create sub range
t2 = 1:20;
% create sub range
t = -20:20;
% create the range

syms x
% define symbolic variable x
T = 10;
% arbitrary period

x_t = rectangularPulse(x/4) * (abs(x)-1);
% create x(t) function according to its formula
Cn = Cn_periodic(x_t,T);
% calculate fourier series coefficients of x(t)
P = [subs(Cn, t1), C0_periodic(x_t, T), subs(Cn, t2)];
phase = angle(P);
% calculate the phases using angle function
mag = abs(P);
% calculate the magnitudes using abs function

figure(1);
% let us have more than one figure
plot(t, phase);
xlabel('t');
ylabel('∠x(t)');
title('q4.phase');

figure(2);
% let us have more than one figure
plot(t, mag);
xlabel('t');
ylabel('|x(t)|');
title('q4.magnitude');
```

