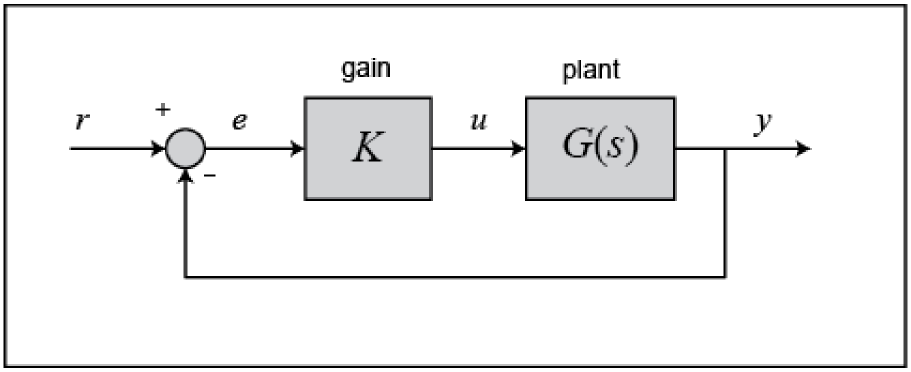
EE324 CONTROL SYSTEMS LAB

Problem Sheet 6

Yatish Vaman Patil | 190070076

Question 1: Proportional Controller

$$\mathrm{Gs} = \frac{1}{(\mathrm{s}+3)(\mathrm{s}+4)(\mathrm{s}+12)}$$



Part A)

We have to find the value of gain K such that the steady-state error of the system becomes 0.489 for a step input

We have T(s) equal to

$$\mathrm{Ts} = \frac{\mathrm{KG}}{1+\mathrm{KG}}$$

$$\mathrm{Es} = \mathrm{Rs} - \mathrm{Ys}$$

$$\mathrm{Ys} = \mathrm{Rs} * \mathrm{Ts}$$

$$\mathrm{Es} = \mathrm{Rs}(1-\mathrm{Ts})$$

$$\mathrm{Rs} = \frac{1}{\mathrm{s}}$$

$$\mathrm{s} * \mathrm{Es} = 1 - \mathrm{Ts}$$

from Final Value Theorem

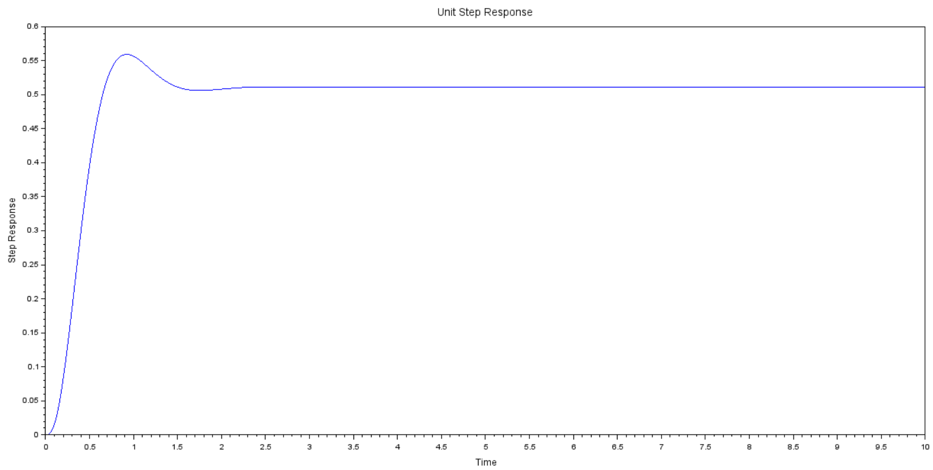
$$\mathrm{e}(\infty) = 1 - \mathrm{Ts}(0)$$

$$\mathrm{e}(\infty) = \frac{1}{1+\mathrm{KG}(0)}$$

$$0.489 = \frac{1}{1 + \frac{k}{3 * 4 * 12}}$$

K=150.478

Unit step response for a given value of k



Code for the same:

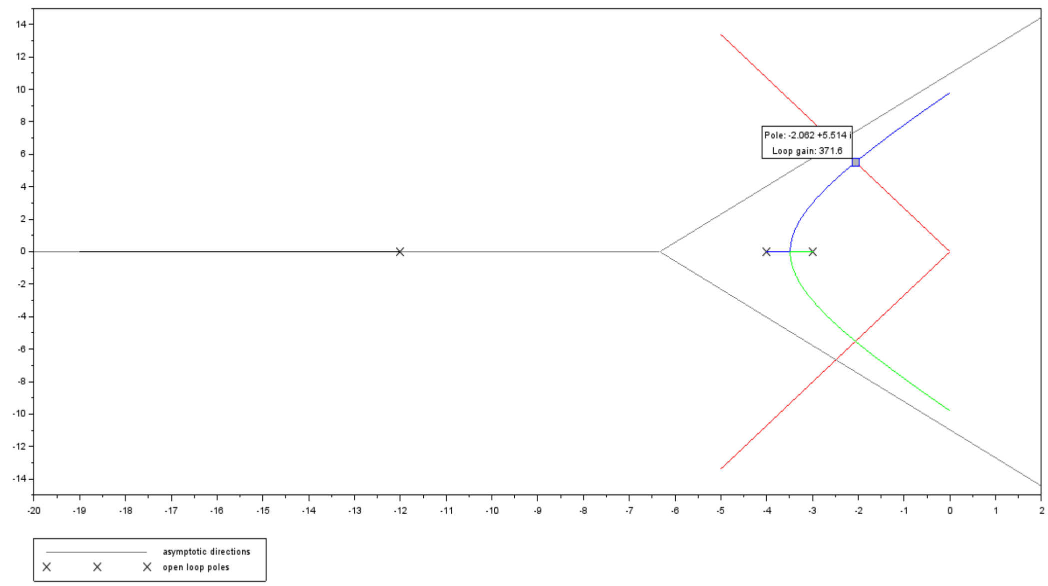
```
s=%s;
Gs=syslin('c',1,(s+3)*(s+4)*(s+12));
time=0:0.01:10;
k=150.478;
P1=csim('step',time,(k*Gs)/(1+k*Gs));
plot(time,P1);
xlabel("Time");
ylabel("Step Response");
xtitle("Unit Step Response");
```

Part B)

We have to find the value of 'k' such that the damping ratio of the system becomes 0.35

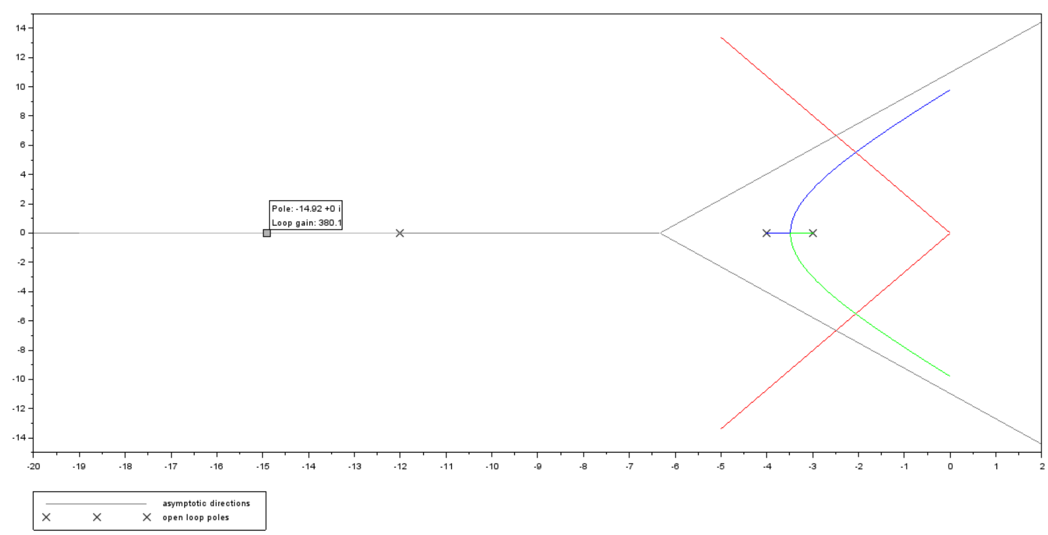
ζ=0.35

From Scilab, we get k=371.6



At this gain, the non-dominant pole is at -14.92, and the dominant poles are at -2.08

Hence second-order approximation is valid here



Code for the same:

```
s=%s;
Gs=syslin('c',1,(s+3)*(s+4)*(s+12));
zeta=0.35;
slope=sqrt(1-zeta^2)/zeta;
x=-5:0.01:0;
OS_line1=slope.*x;
```

```
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
evans(Gs,kpure(Gs));
```

Part C)

We have to find gain value at breakaway point

From the closed-loop transfer function, we have the following relation for root locus plots

1+K*Gs=0

$$k = -\frac{1}{Gs}$$

$$k = -(s+3)(s+4)(s+12)$$

$$k = -s^3 - 19s^2 - 96s - 144$$

differentiate k w.r.t. s

$$k' = -3s^2 - 38s - 96$$

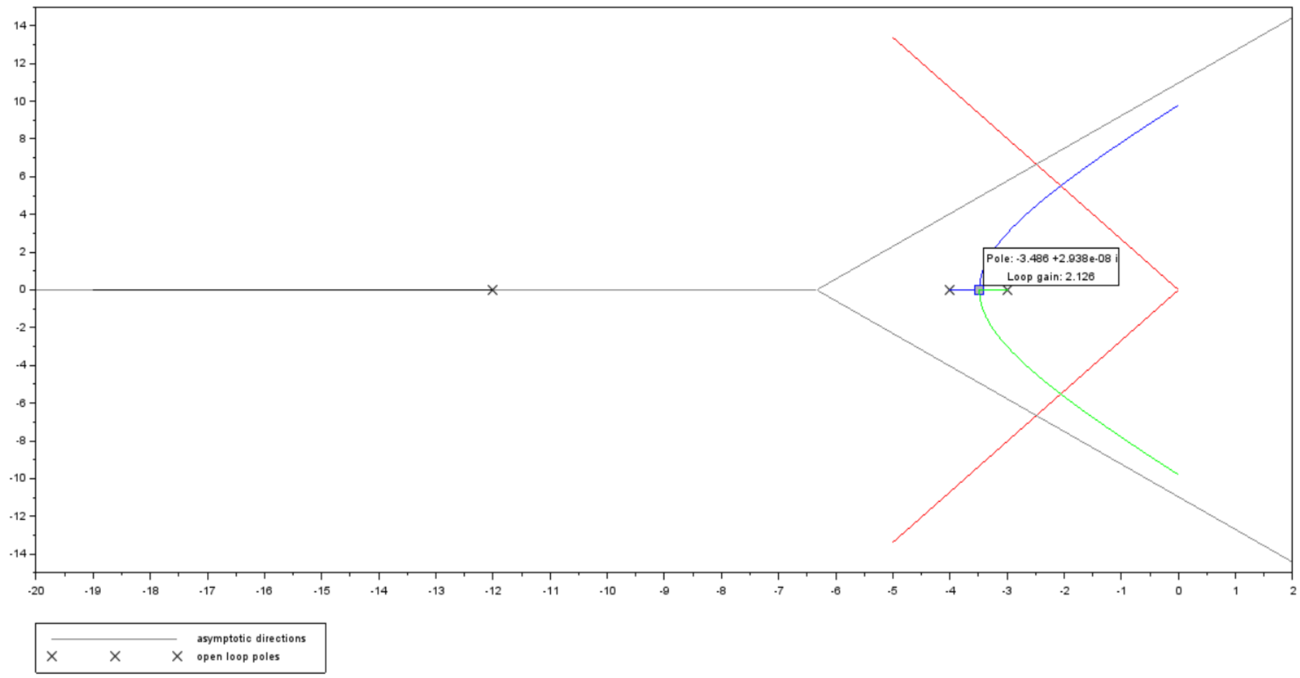
K'=0

s= -9.18 or s= -3.49

we have to take a value between -3 and -4

s=-3.49

k=2.126

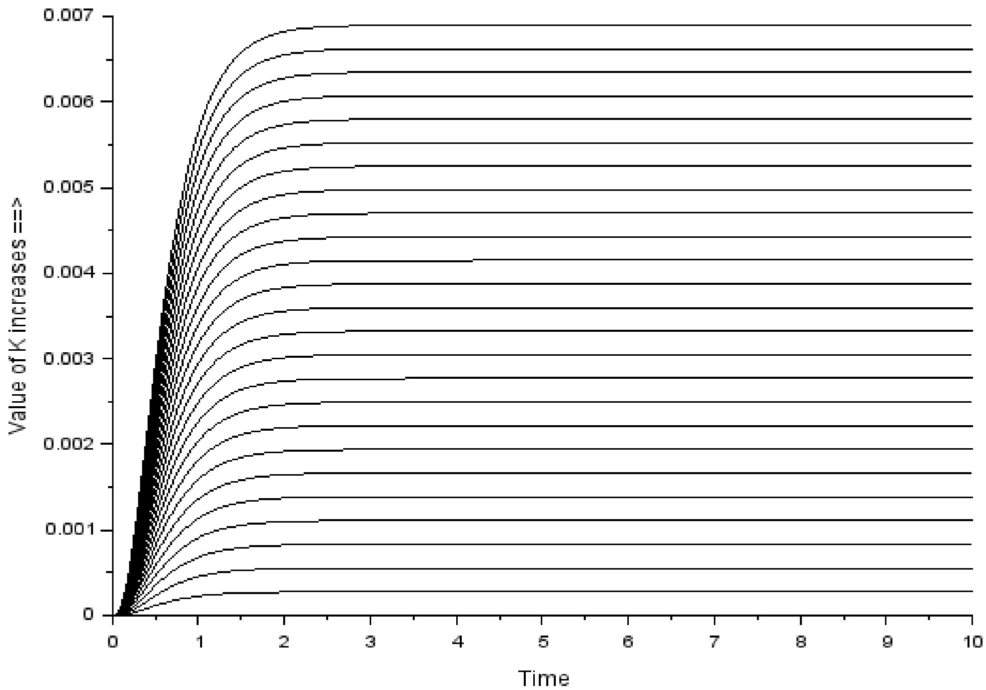


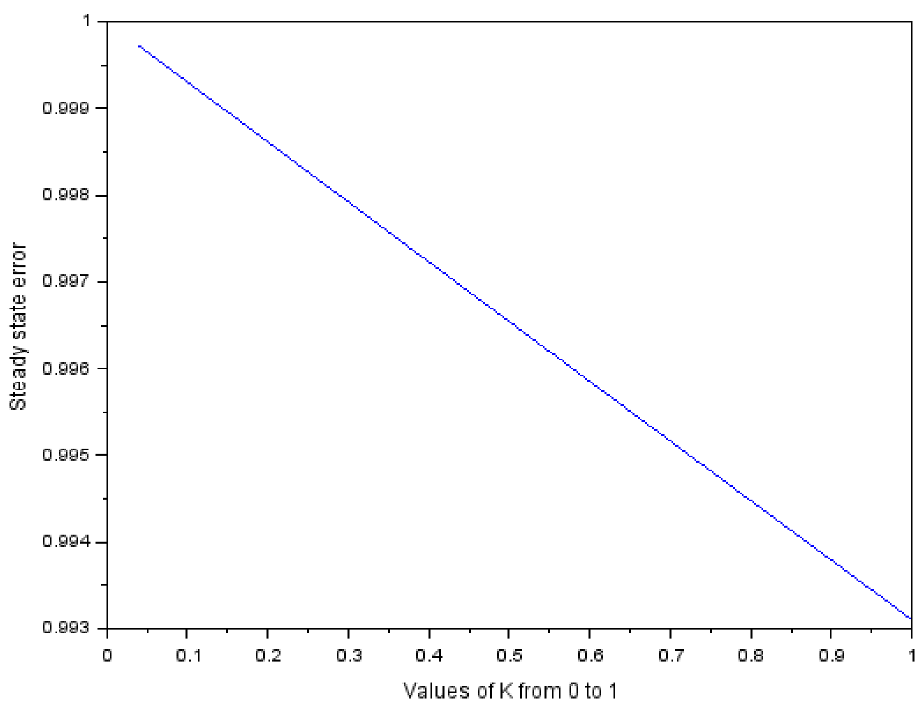
Code for the same:

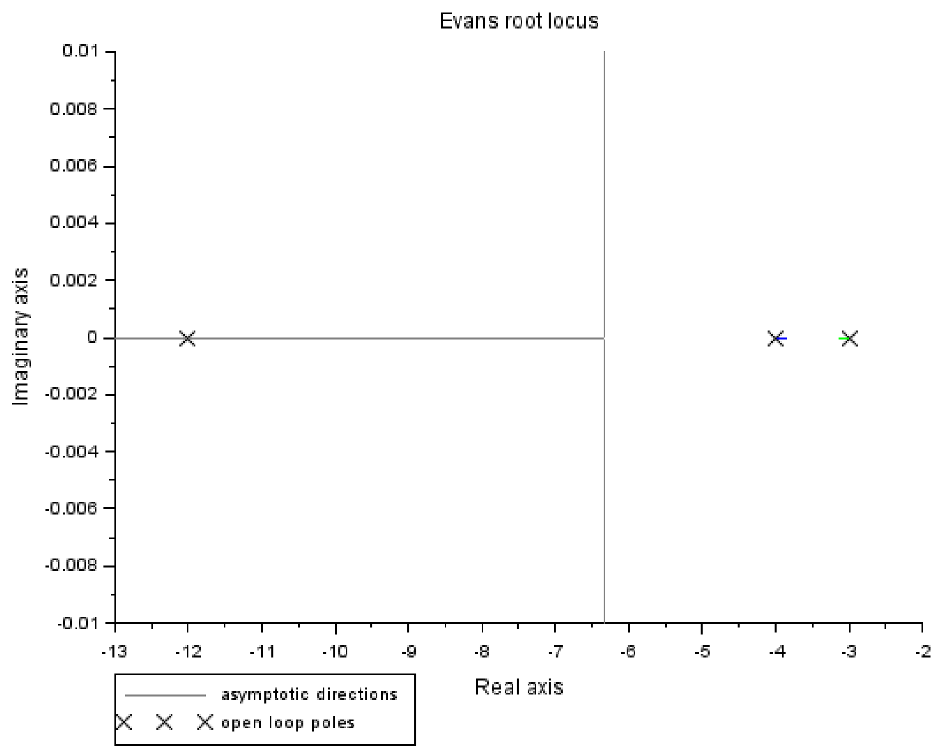
```
s=%s;
Gs=syslin('c',1,(s+3)*(s+4)*(s+12));
zeta=0.35;
slope=sqrt(1-zeta^2)/zeta;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
evans(Gs,kpure(Gs));
```

Part D)

We have to vary the value of k from 0 to 1 in small steps and find its unit step response Using Scilab


0.007
0.006
0.005
0.004
0.003
0.002
0.001
0
0
1
2
3
4
5
6
7
8
9
10
Value of K increases ==>
Time


1
0.999
0.998
0.997
0.996
0.995
0.994
0.993
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Steady state error
Values of K from 0 to 1



1. As we increase the value of k, the steady-state value increase
2. As k increases, steady-state error decreases
3. Fork between 0 and 1, the root locus is on the real axis. I.e. closed-loop poles are real. Hence system remains overdamped

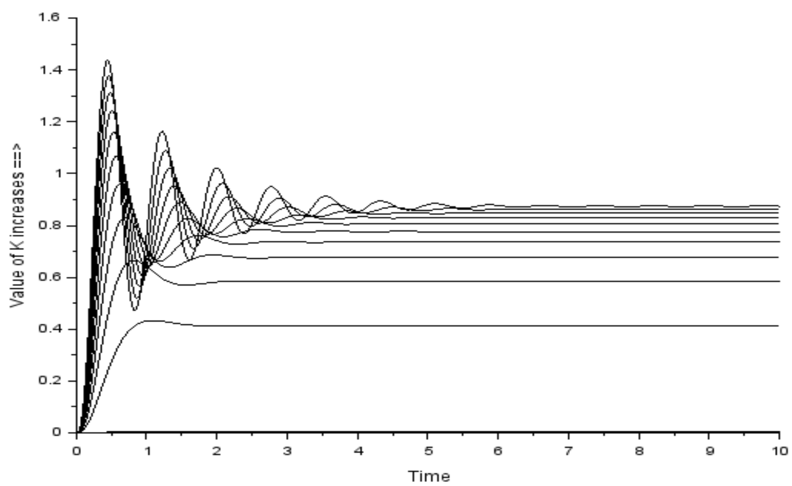
Code for the same:

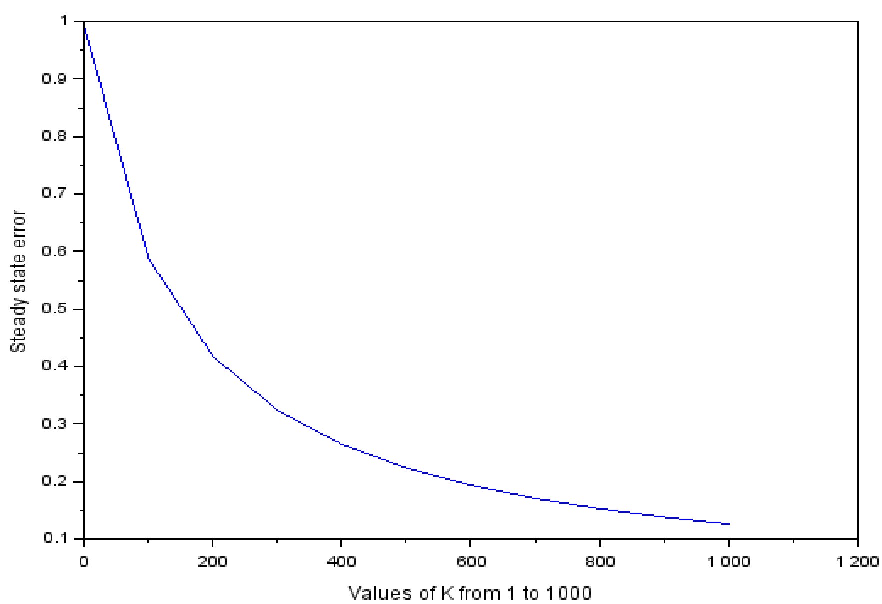
```
s=%s;
Gs=syslin('c',1,(s+3)*(s+4)*(s+12));
k_value=0.04:0.04:1;
time=0:0.01:10;
ss_error=ones(1,25);
i=1;
for k = k_value
    Ts=k*Gs/(1+k*Gs);
    Ts=syslin('c',Ts);
    scf(1);
    plot2d(time,csim('step',time,Ts));
    ylabel("Value of K increases ==>");
    xlabel("Time");
    ss_error(i)=1/(1+k/144);
    i=i+1;
end
scf(2);
plot(k_value,ss_error);
```

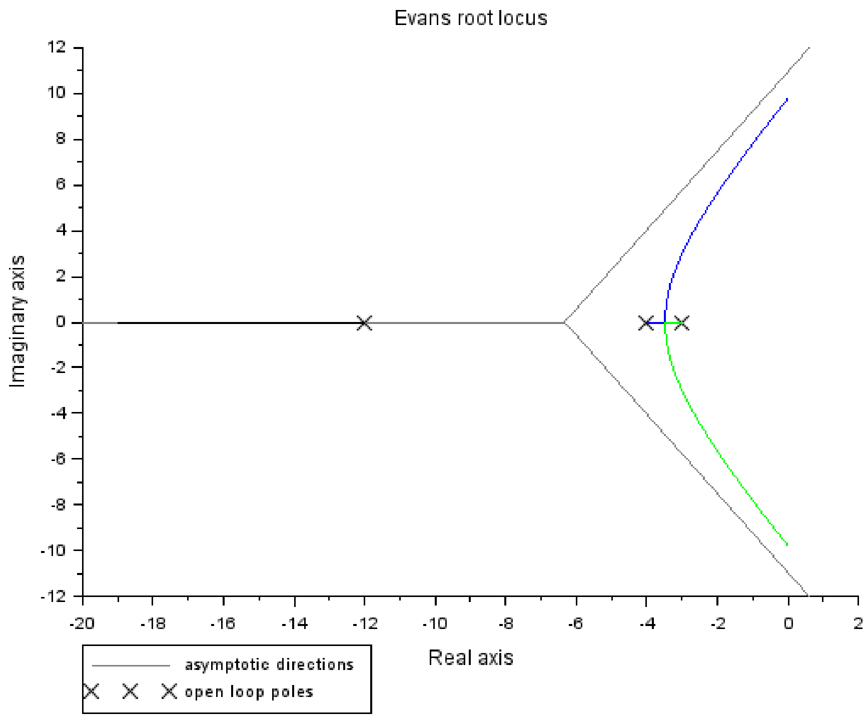
```
xlabel("Values of K from 0 to 1");
ylabel("Steady state error");
scf(3);
evans(Gs,1);
```

Part E)

We have to vary the value of k from 1 to 1000







1. For 'k' between 1 to 2.126, we have real closed-loop poles
2. For 'k' between 2.126 to 1000, we have an underdamped system
3. As 'k' increases, the steady-state error decreases
4. As 'k' increases, the settling time of unit step response increases
5. As 'k' increases, the system goes towards the instability

Code for the same:

```
s=%s;
Gs=syslin('c',1,(s+3)*(s+4)*(s+12));
k_value=1:100:1001;
time=0.01:0.01:10;

ss_error=ones(1,11);

i=1;
for k = k_value
    Ts=k*Gs/(1+k*Gs);
    Ts=syslin('c',Ts);
    scf(1);
    Ps=csim('step',time,Ts);
    plot2d(time,Ps);
    ylabel("Value of K increases ==>");
```

```
    xlabel("Time");
    ss_error(i)=1/(1+k/144);
    i=i+1;
end
scf(2);
plot(k_value,ss_error);
xlabel("Values of K from 1 to 1000");
ylabel("Steady state error");
scf(3);
evans(Gs,kpure(Gs));
```

Question 2: PI Controller

$$Gs = \frac{1}{(s+3)(s+4)(s+12)}$$

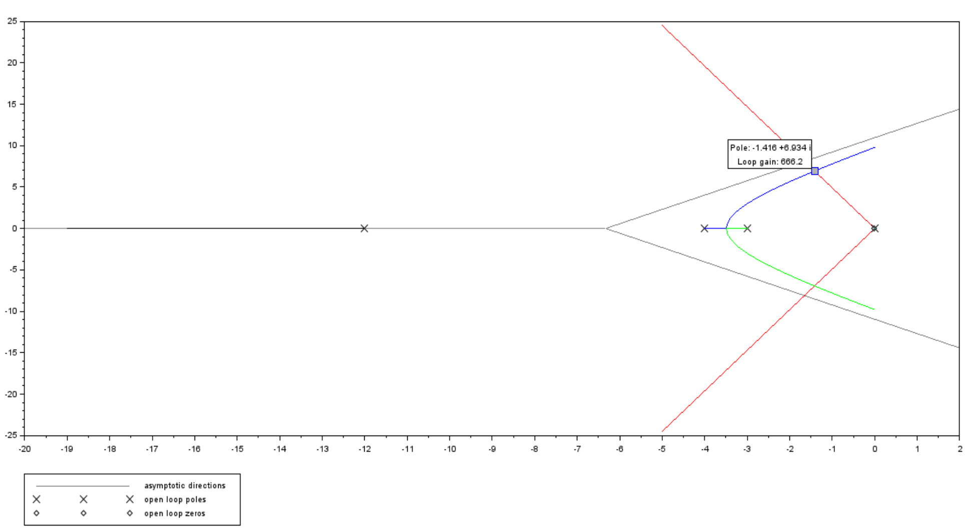
Part A)

Here we are given pole and zero of PI controller. We have to find a value of K for which the damping ratio becomes 0.2

After adding PI controller, our Gs becomes

$$Gs = \frac{(s+0.01)}{s(s+3)(s+4)(s+12)}$$

Now using Scilab, we get



K=666.2

PI Controller➔ $\frac{666.2(s+0.01)}{s}$=$666.2+\frac{6.662}{s}$

K_p=666.2 and K_I=6.662

Here PI controller makes a steady-state error for step input equal to 0

Code for the same:

```
s=%s;
Gs=syslin('c',(s+0.01),s*(s+3)*(s+4)*(s+12));
zeta=0.2;
slope=sqrt(1-zeta^2)/zeta;
x=-5:0.01:0;
OS_line1=slope.*x;
OS_line2=-1*slope.*x;
plot(x,OS_line1,'r');
plot(x,OS_line2,'r');
evans(Gs,kpure(Gs));
```

Part B)

Here we have to find value k for which w_n equals 8 and 9

$$Gs=\frac{(s+0.01)}{s(s+3)(s+4)(s+12)}$$

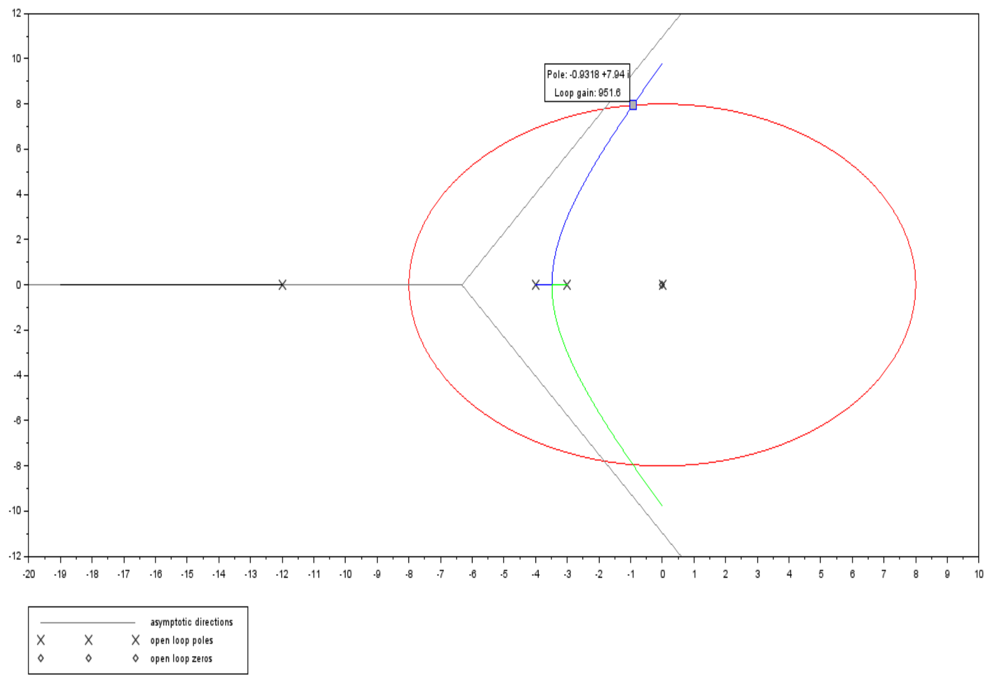
We know that points on a circle centred at the origin in the Laplace plane have a constant w_n value. Using this property and Scilab, we can find the value of K

1)For w_n=8

K=951.6

Non-dominant Pole➔-17.11

The second-order approximation is valid

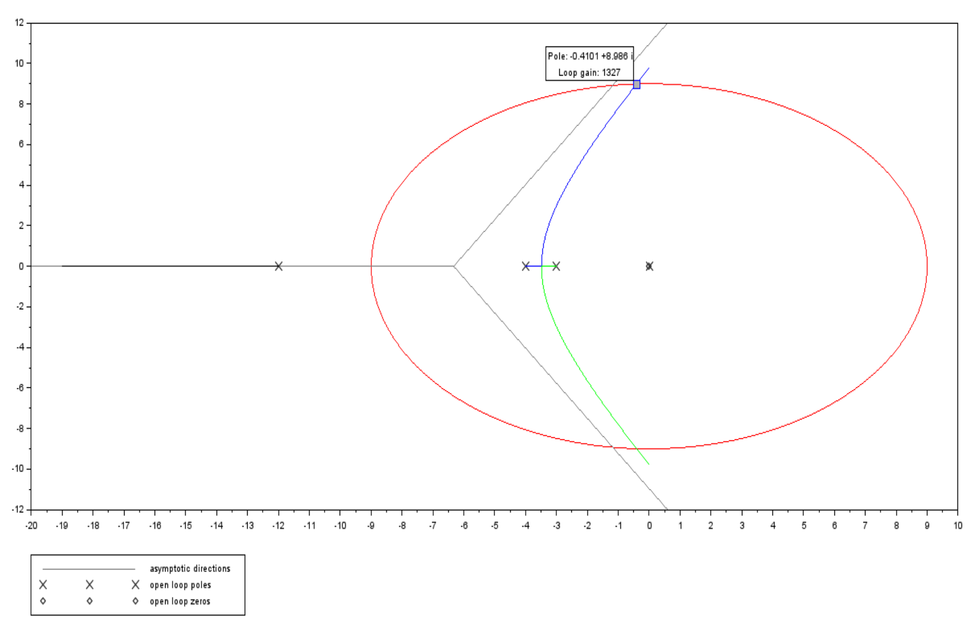


2)For w_n=9

K=1327

Non-dominant pole➔-18.18

The second-order approximation is valid

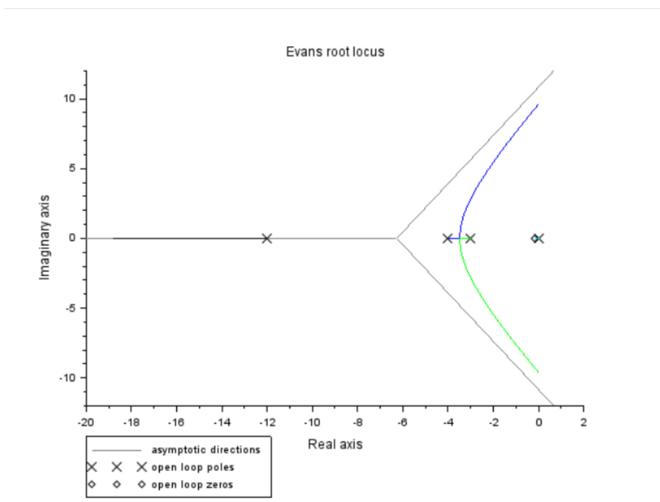


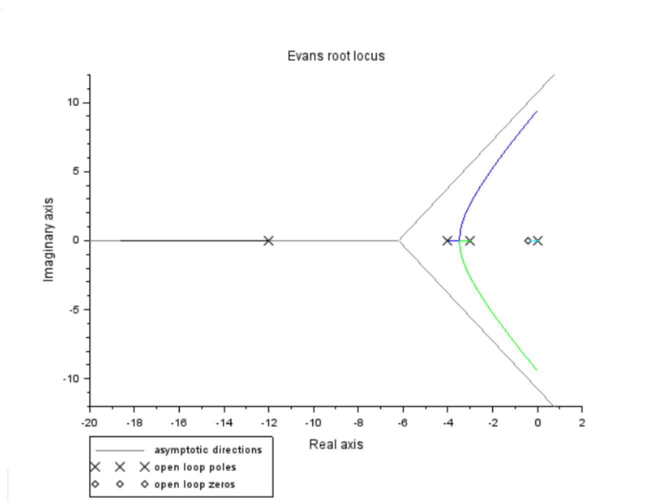
Code for the same:

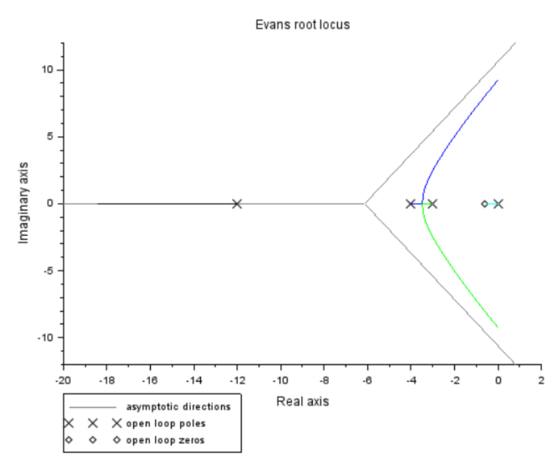
```
s=%s;
Gs=syslin('c',(s+0.01),s*(s+3)*(s+4)*(s+12));
zeta=0.2;

r=9;
t=linspace(0,2*%pi,100);
x=r*cos(t);
y=r*sin(t);
plot(x,y,'r');
evans(Gs,kpure(Gs));
```

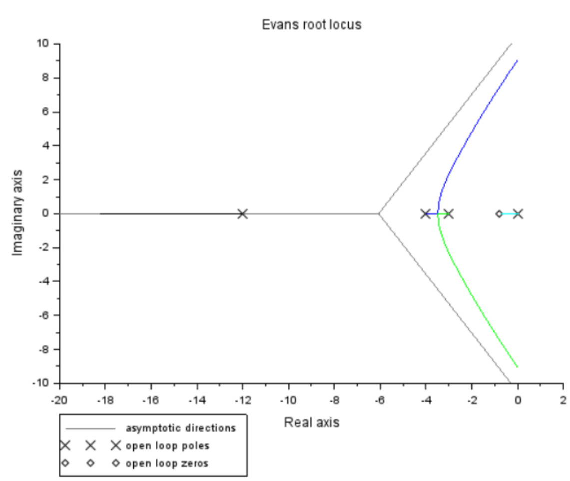
Part C)

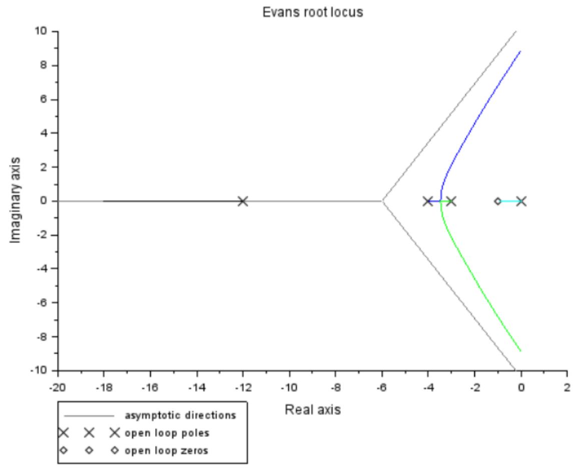
Here I have varied the value of z from 0.2 to 2 in steps of 0.2

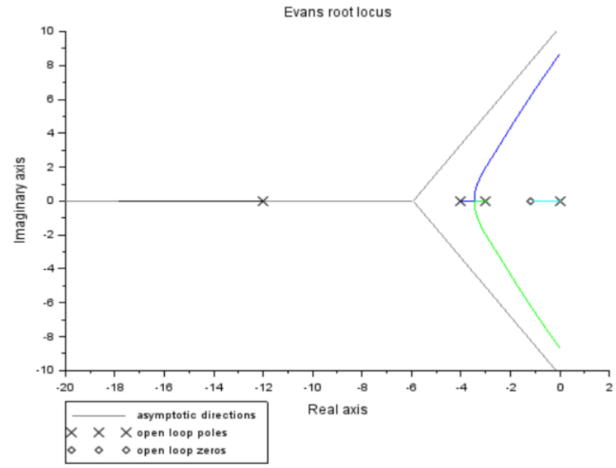


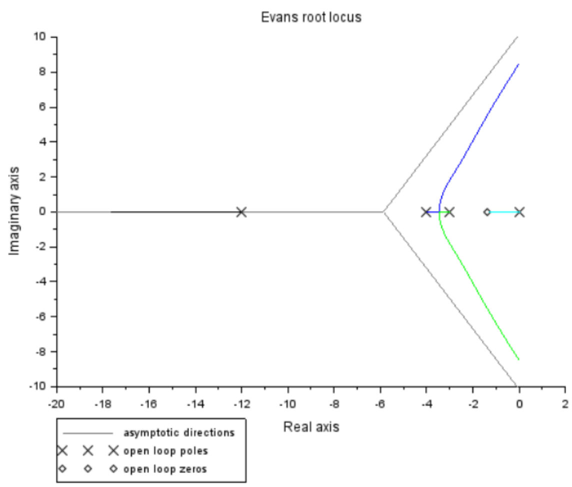


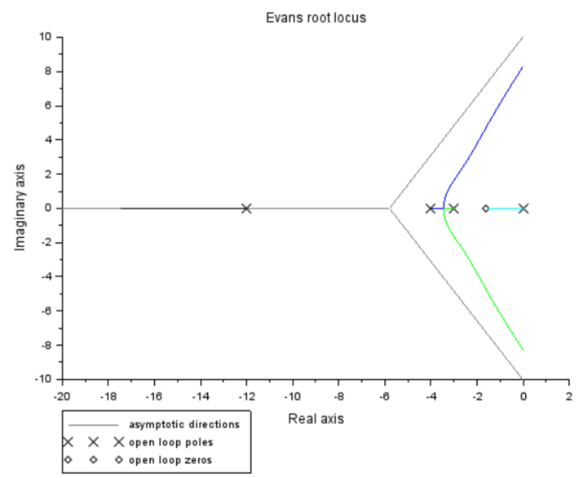


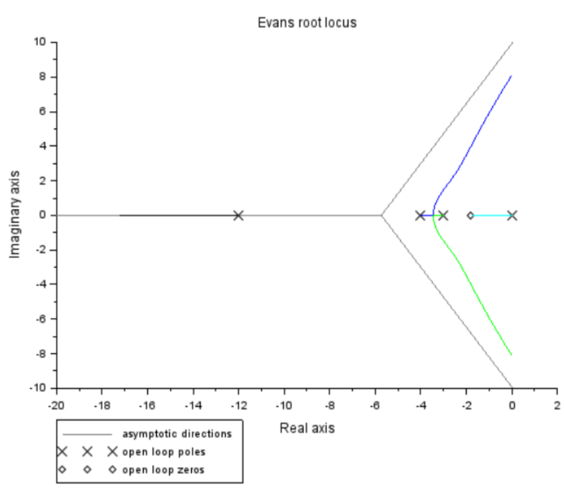


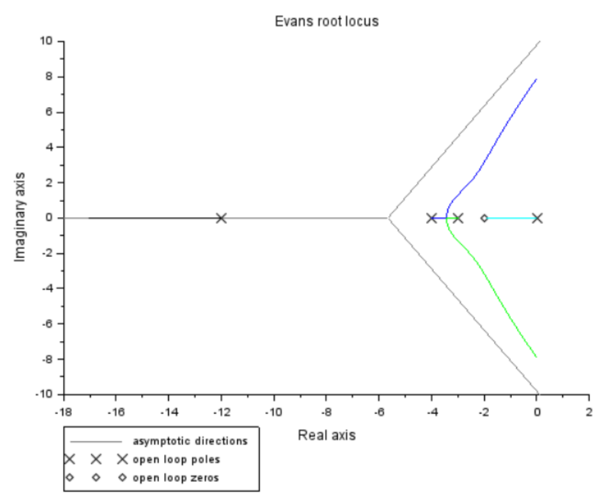

Evans root locus
Imaginary axis
Real axis
asymptotic directions
open loop poles
open loop zeros


Evans root locus
Imaginary axis
Real axis
asymptotic directions
open loop poles
open loop zeros


Evans root locus
Imaginary axis
Real axis
asymptotic directions
open loop poles
open loop zeros


Evans root locus
Imaginary axis
Real axis
asymptotic directions
open loop poles
open loop zeros


Evans root locus
Imaginary axis
Real axis
asymptotic directions
open loop poles
open loop zeros


Evans root locus
Imaginary axis
Real axis
asymptotic directions
open loop poles
open loop zeros

Code for the same:

```
s=%s;
z_value=0.2:0.2:2
for z=z_value
    scf(z*2)
    Gs=syslin('c',(s+z),s*(s+3)*(s+4)*(s+12));
    evans(Gs,kpure(Gs));

end
```

Part D)

It is possible to alter the closed-loop pole location of the system using the PI controller without changing the damping ratio. We can achieve this by the varying value of Z. For slight variations in z, closed-loop poles do not change that much. But for the significant variations in z, as we can see in Part C, for increasing value of z(more negative), Root locus become more unstable. Centroid shifts toward the right as we increase the value of z

Question 3: Frequency Response

Part A)

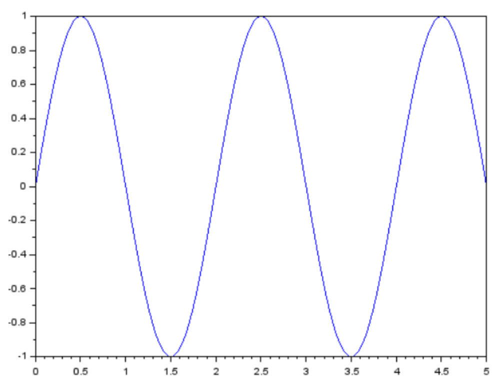
We have Gs equal to

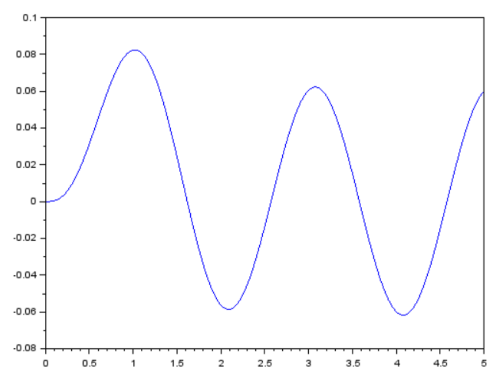
$$Gs = \frac{1}{s^2 + 5s + 6}$$

Input =sin(2πft)

We will plot output for the following five frequencies

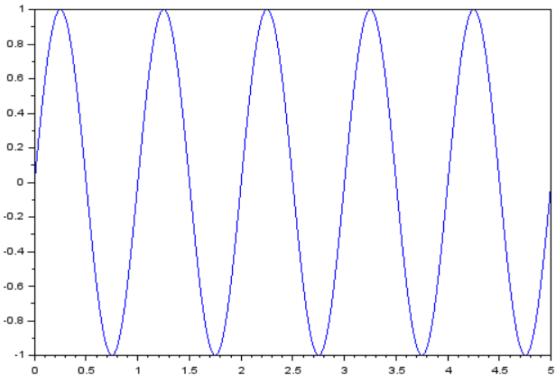
f1=0.5 Hz



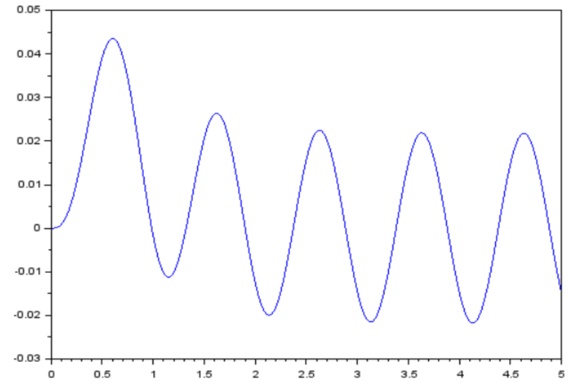


Input **Output**

f2=1Hz

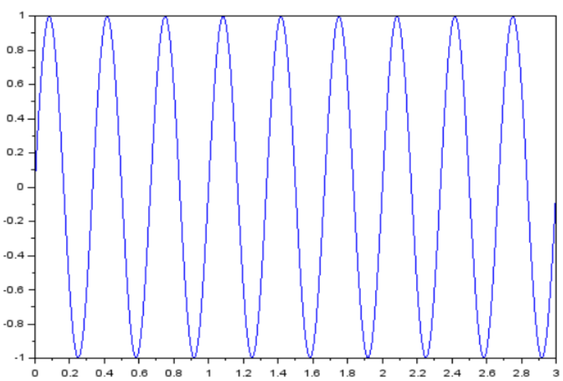


Input

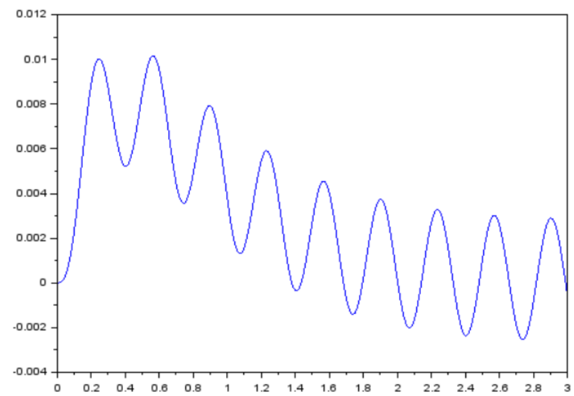


Output

f3=3Hz

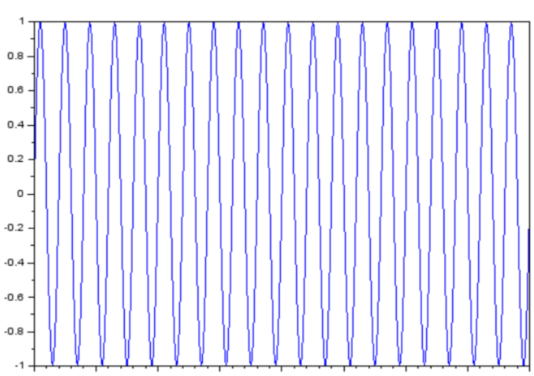


Input

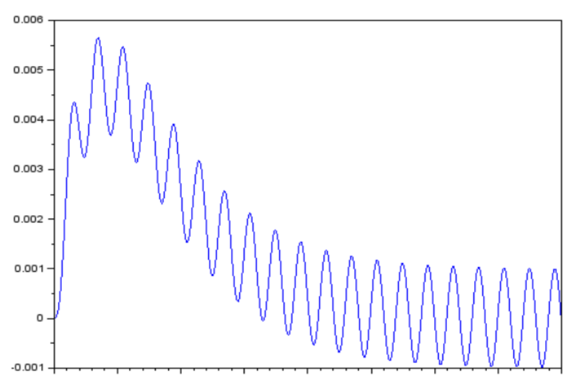


Output

f4=5Hz

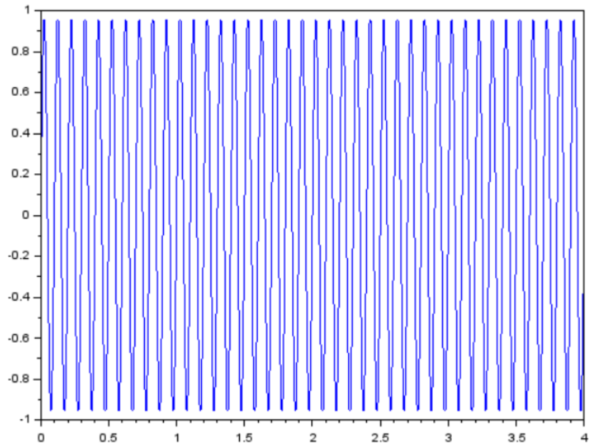


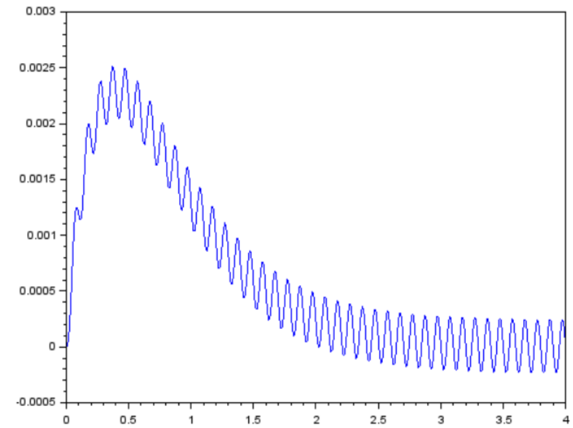
Input



Output

f5=10Hz



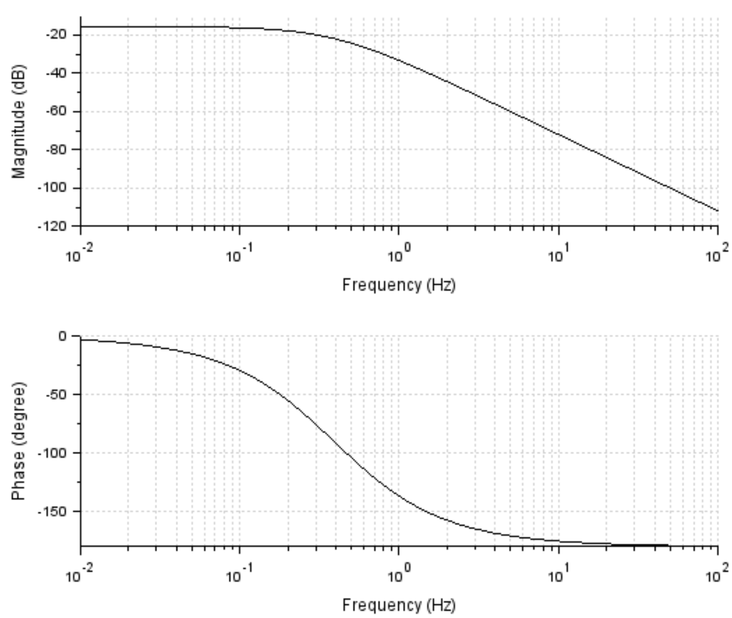


Input **Output**

Code for the same:

```
s=%s;
f=10
t=0:0.01:4;
Gs=syslin('c',1/(s^2+5*s+6));
Ps=csim(sin(2*%pi*f*t),t,Gs);
plot(t,Ps);
scf(2);
plot(t,sin(2*%pi*f*t));
```

Bode Plots



In the above time-domain responses. For different frequencies, we can notice that the ratio of the magnitude of output and input is equal to |G(jw)| and change in phase is equal to the angle of G(jw), i.e. phase of G(jw)

Part B)

The desired relation obtained between phase difference and angle of G(jw) is in rad/sec because we are plotting the value of w, and it is expressed in rad/sec

We get this angle of G(jw) from the phase plot of G(jw) for different values of frequencies