**Program #3**

**EE 5350**

**Burhan Ayub Wani**

**1001050444**

**Listing of Main**

Nx = 100;

Nm = 100;

%[S]= Signal(w,n);

for n=0:Nx

S(n+1) = Signal(.2,n) + Signal(1.3,n) + Signal(2.5,n);

end

figure;

plot(0:Nx,S);

title ('signal S');

xlabel ('n');

ylabel ('S');

[Am,w] = Amp(Nm,S,Nx);

figure;

plot(Am);

title ('signal Am');

xlabel ('w');

ylabel ('Am');

[F]= DSINE(0.2,1);%passing wc = 0.2 as cuttoff frequency

figure;

plot(F);

title('h(n)');

xlabel('n');

ylabel('impulse response');

z= length(F)-1;

[X,w]= Amp(Nm,F,z);

figure;

plot(X);

title('frequency response of filter');

xlabel('w');

ylabel('H(ejw)');

Nx=100;

M=length(F);

Ny=Nx+M;

y1=zeros(Ny+M+1,1);%creating a vector of all zeros of length Ny+M+1

x1=zeros(Nx+M+1,1);

for m=1:Nx+1

x1(m+M)=S(m);%we will store the values of S temporarily in x1 starting at M+1 location

end

for n= M:Nx+M

y1(n+1)=y1(n)+(x1(n+1)-x1(n+1-M)/M);%we shift the values so that the inputs to each variable remains within bound

end

for m=1:Ny+1

y(m)=y1(m+M);

end

figure;

plot(y);

[B,w]=Amp(Nm,y,Nx);%Frequency domain representation

figure;

plot(B);

title('Y(ejw)');

**Listing of Signal**

function [S]= Signal(w,n)

S = n\*exp(-n/6)\*cos(w\*n);

end

**Listing of DSINE**

function [F]= DSINE(Wc,m) % This function creates digital low pass moving average filter

M2 = floor((2\*pi\*m)/Wc)-1; % Here floor is used to round off the digit to it's lower interger value

for n = 0:M2+1

F(n+1)= (1/(M2+1))\*1; % Here value of M2 has been calculated after equating H(e^jw) to zero.

end

end

**Listing of Amp**

function [Am,w] = Amp(Nm,S,Nx);

for k=0:Nm

w(k+1)= (pi\*k)/Nm;

Z(k+1) = exp(1i\*w(k+1));

end

for k=0:Nm

Am(k+1)=0;

for n =0:Nx

Am(k+1)= (Am(k+1) + S(n+1)\*Z(k+1)^n);

end

end

for k=0:Nm

Am(k+1)=abs(Am(k+1)); %absolute value or magnitude

end

end





