3. LINEAR AND CIRCULAR CONVOLUTIONS

PROGRAM:-

```
%Program for linear convolution
%to get the input sequence
n1=input('enter the length of input sequence');
n2=input('enter the length of impulse sequence');
x=input('enter the input sequence');
h=input('enter the impulse sequence');

%convolution operation
y=conv(x,h);
%to plot the signal
subplot(3,1,1);
stem(x);
ylabel('amplitude');
xlabel('n1. .. >');
title('input sequence')
subplot(3,1,2);
stem(h);
ylabel('amplitude');
xlabel('n2. .. >');
title('impulse signal')
subplot(3,1,3);
stem(y);
ylabel('amplitude');
xlabel('n3');
disp('the resultant signal is');y
```

OUTPUT : LINEAR CONVOLUTION

```
Enter the length of input sequence 4
Enter the length of impulse sequence 4
Enter the input sequence [1 2 3 4]
Enter the impulse sequence  [4  3 2 1]

The resultant signal is
y= 4 11 20 30 20 11 4
```

```
%%Program for circular convolution
clc;
clear all;
close all;
%to get the input sequence
g=input('enter the input sequence');
h=input('enter the impulse sequence');
N1=length(g);
N2=length(h);
N=max(N1,N2);
N3=N1-N2
%loop for getting equal length sequence
if(N3>=0)
h=[h,zeros(1,N3)];
else
g=[g,zeros(1,-N3)];
end
%computation of circular convoluted sequence
for n=1:N;
y(n)=0;
for i=1:N;
j=n-i+1;
if(j<=0)
j=N+j;
end
y(n)=y(n)+g(i)*h(j);
end
end
figure;
subplot(3,1,1);
stem(g);
ylabel('amplitude');
xlabel('n1..>');
title('input sequence')
subplot(3,1,2);
stem(h);
ylabel('amplitude');
xlabel('n2');
title('impulse sequence')
subplot(3,1,3);
stem(y);
ylabel('amplitude');
xlabel('n3');
disp('the resultant signal is');
```

OUTPUT : CIRCULAR CONVOLUTION
Enter the input sequence [1 2 2 1]
Enter the impulse sequence [4 3 2 1]

The resultant signal is
y= 15 17 15 13

4. AUTO-CORRELATION AND CROSS-CORRELATION

PROGRAM:

```
clc; clear all; close all;
t=0:0.01:1;
f1=3;
x1=sin(2*pi*f1*t);
figure;
subplot(2,1,1);
plot(t,x1);
title('sine wave');
xlabel('time ---- >');
ylabel('amplitude ---- >');
grid;
[rxx lag1]=xcorr(x1);
subplot(2,1,2);
plot(lag1,rxx);
grid;
title('auto-correlation function of sine wave');
figure;
subplot(2,2,1);
plot(t,x1);
title('sine wave x1');
xlabel('time ---- >');
ylabel('amplitude ---- >');
grid;
f2=2;
x2=sin(2*pi*f2*t);
subplot(2,2,2);
plot(t,x2);

title('sine wave x2');
```

```
xlabel('time ---->');,ylabel('amplitude ---- >');
grid;
[cxx lag2]=xcorr(x1,x2);
subplot(2,2,[3,4]);
plot(lag2,cxx);
grid;
title('cross-correlation function of sine wave');
```

PROGRAM:

```
clc;
close all;
clear all;
% two input sequences
x=input('enter input sequence');
h=input('enter the impulse suquence');
subplot(2,2,1);
stem(x);
xlabel('n');
ylabel('x(n)');
title('input sequence');
subplot(2,2,2);
stem(h);
xlabel('n');
ylabel('h(n)');
title('impulse sequence');
% cross correlation between two sequence
y=xcorr(x,h);
subplot(2,2,3);
stem(y);
xlabel('n');
ylabel('y(n)');
title(' cross correlation between two sequences ');
% auto correlation of input sequence
z=xcorr(x,x);
subplot(2,2,4);
stem(z);
xlabel('n');
ylabel('z(n)');
title('auto correlation of input sequence');
```

INPUT:

```
enter input sequence [1 2 5 7]
```

enter the impulse sequence [2 6 0 5 3]

5.DISCRETE FOURIER TRANSFORM

PROGRAM:

```
clc;
close all;
clear all;
tic;
fprintf('Date & Time:');
Date= datestr(now);
disp(Date);
%DFT
disp('D.F.T');
a=input('Enter the input sequence:');
n=length(a);
x=fft(a,n);
for k=1:n;
y(k)=0;
for i=1:n
y(k)=y(k)+a(i)*exp((-j)*2*pi*(i-1)*(k-1)*(1/n));
end
end
error=x-y;
disp(x);
disp(y);
disp(error);
subplot(3,2,1);
stem(a);
xlabel('time index n ----- >');
ylabel('amplitude');
title('input sequence');
subplot(3,2,2);
stem(0:n-1,abs(x));
xlabel('time index n ----- >');
ylabel('amplitude');
title('FFT sequence by inbuilt command');
subplot(3,2,5);
stem(0:n-1,abs(y));
xlabel('time index n ----- >');
```

```
ylabel('amplitude');
title('DFT by formula calculation');
subplot(3,2,6);
stem(error);
xlabel('time index n ----- >');
ylabel('amplitude');
title('error sequence');
% Power Spectral Density
P = x.* conj(x) / 512;
f = 1000*(0:256)/512;
figure,plot(f,P(1:257))
title('Frequency content of y');
xlabel('frequency (Hz)');
```

INPUT:
D.F.T
Enter the input sequence:[1 1 2 1 2 1 3 1]

OUTPUT
Columns 1 through 4
12.0000 -1.0000 + 1.0000i -2.0000 -1.0000 - 1.0000i
Columns 5 through 8
4.0000 -1.0000 + 1.0000i -2.0000 -1.0000 - 1.0000i
Columns 1 through 4
12.0000 -1.0000 + 1.0000i -2.0000 - 0.0000i -1.0000 - 1.0000i
Columns 5 through 8
4.0000 + 0.0000i -1.0000 + 1.0000i -2.0000 - 0.0000i -1.0000 - 1.0000i
1.0e-014 *
Columns 1 through 4
0 0.0444 + 0.0222i 0.0444 + 0.0888i -0.0666 - 0.0666i
Columns 5 through 8
0 - 0.1715i 0.7105 + 0.1887i 0.2665 + 0.2665i -0.1887 - 0.0666i
Elapsed time is 27.683359 seconds

7. INVERSE FAST FOURIER TRANSFORM (IFFT)

PROGRAM:

```
clc;
clear all;
close all;
n=input('enter value of n=');
x=input('enter input sequence=');
a=1:1:n;
y=fft(x,n);
disp('fft of input sequence');
disp(y);
z=ifft(y);
disp('ifft of input sequence');
disp(z);
```

OUTPUT:

```
enter value of n=8
enter input sequence=[1 2 3 4 5 6 7 8]
fft of input sequence
Columns 1 through 6
36.0000 + 0.0000i -4.0000 + 9.6569i -4.0000 + 4.0000i -4.0000 + 1.6569i -4.0000 + 0.0000i -
4.0000 - 1.6569i
Columns 7 through 8 -4.0000 - 4.0000i -4.0000 - 9.6569i
ifft of input sequence
1 2 3 4 5 6 7 8
```