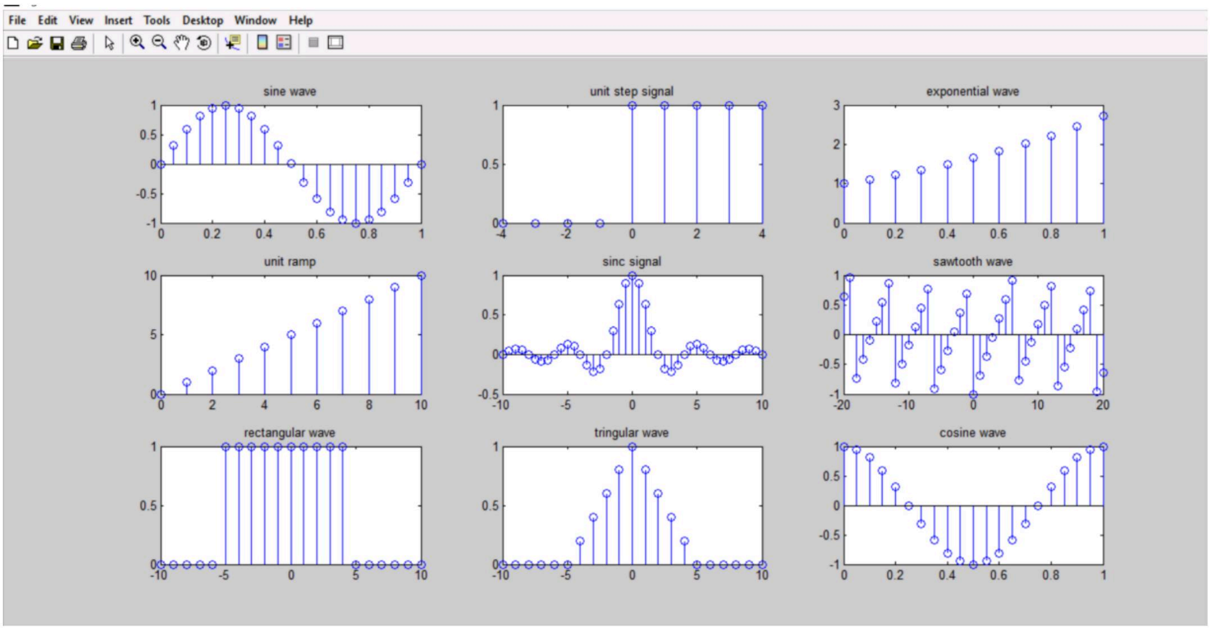
Generation Of DT Signal’s

```
clc;
n = 0:0.05:1;
x1 = sin(2*pi*n);
subplot(3,3,1);
stem(n,x1);
title('sine wave');

n9 = -4:1:4;
x2 = [zeros(1,4),ones(1,5)];
subplot(3,3,2);
stem(n9,x2);
title('unit step signal')

n1 = 0:0.1:1;
x3 = exp(n1);
subplot(3,3,3);
stem(n1,x3);
title('exponential wave');

n2 = 0:1:10;
x3 = n2;
subplot(3,3,4);
stem(n2,x3);
title('unit ramp');

n3 = -10:0.5:10;
x4 = sinc(n3/2);
subplot(3,3,5);
stem(n3,x4);
title('sinc signal');

n4 = -20:1:20;
x5 = sawtooth(n4);
subplot(3,3,6);
stem(n4,x5);
title('sawtooth wave');

n5 = -10:1:10;
x6 = rectpuls(n5/10);
subplot(3,3,7);
stem(n5,x6);
title('rectangular wave');
```

```
x7 = tripuls(n5/10);
subplot(3,3,8);
stem(n5,x7);
title('tringular wave');

x8 = cos(2*pi*n);
subplot(3,3,9);
stem(n,x8);
title('cosine wave');
```

Output



Auto and Cross Correlation

Autocorrelation

```
  x=input ('Enter sequence x(n)=');
  rxx= conv(x,fliplr(x));
  disp('rxx=');
  disp(rxx);
  figure(1);
  stem(rxx,'filled');
  title('Autocorrelation output');
  xlabel('lag index');
  ylabel('amplitude');
```

Result:

```
Enter sequence x(n)=[1 2 3 4]
rxx=
     4    11    20    30    20    11     4
```

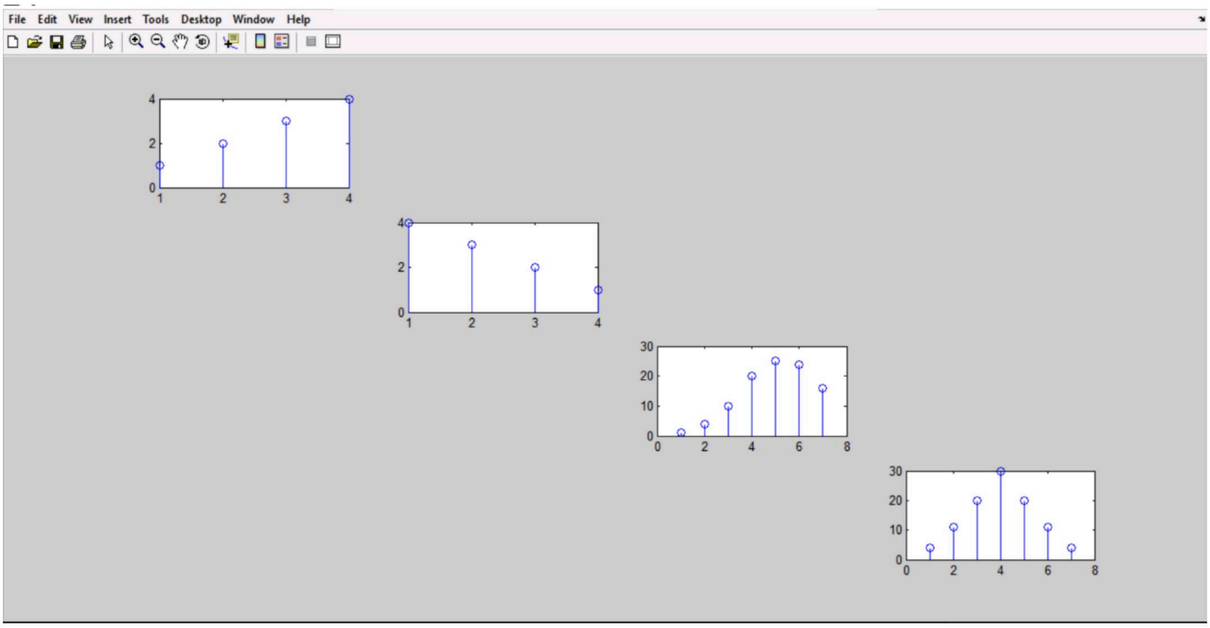
Cross Correlation

```
 x=input ('Enter sequence x(n)=');
  y=input ('Enter sequence y(n)=');
  rxy= conv(x,fliplr(y));
  disp('rxy=');
  disp(rxy);
  figure(1);
  stem(rxy,'filled');
  title('cross correlation output');
  xlabel('lag index');
  ylabel('amplitude');
```

Result:

```
Enter sequence x(n)=[1 2 3 4]
Enter sequence y(n)=[4 3 2 1]
rxy=
     1     4    10    20    25    24    16
```

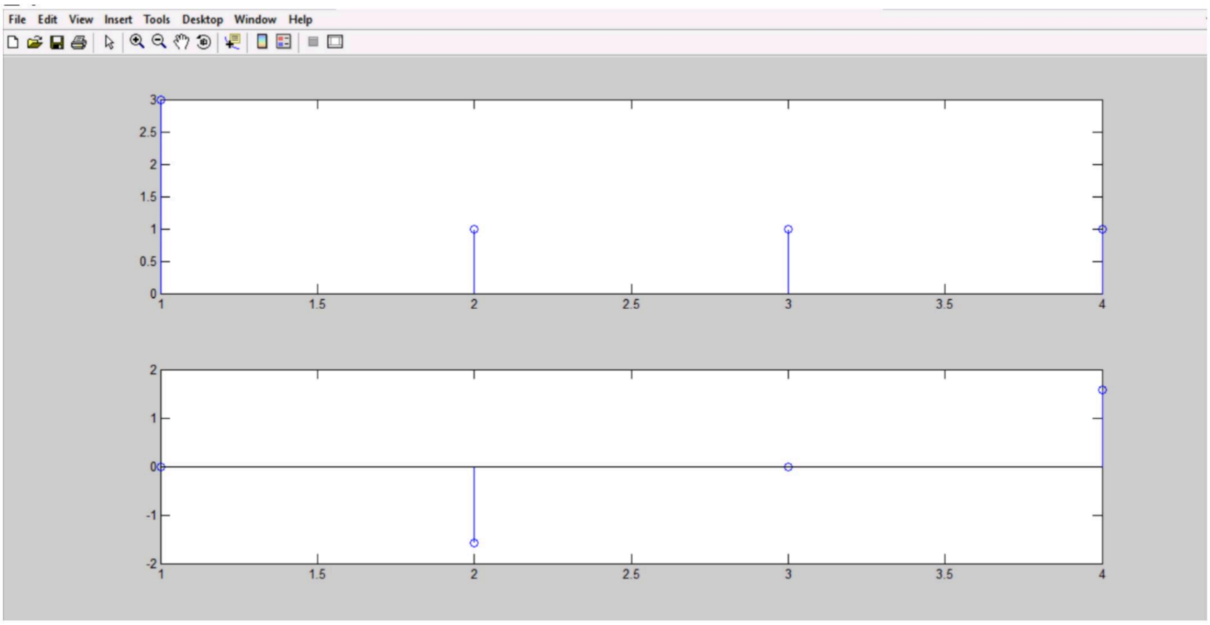
Output :


File Edit View Insert Tools Desktop Window Help
4
2
0
1 2 3 4
4
2
0
1 2 3 4
30
20
10
0
0 2 4 6 8
30
20
10
0
0 2 4 6 8

DFT

```
%DFT using direct function
x=input('enter i/p seq. in square bracket')
x=fft(x)
stem(abs(x))
stem(angle(x))
i/p [3 2 1 1]

3.)DFT without using direct function
x=input('enter i/p seq. in square bracket')
N=length(x);
k=0:1:N-1;
n=0:1:N-1;
Wn=exp(-i*2*pi/N);
nk=n'*k;
Wnk=Wn.^nk;
x=x*Wnk;
subplot(2,1,1)
stem(abs(x))
disp(x);
title'amplitude'
subplot(2,1,2)
stem(angle(x))
title'phase'
```

Output


File Edit View Insert Tools Desktop Window Help
3
2.5
2
1.5
1
0.5
0
1
1.5
2
2.5
3
3.5
4
2
1
0
-1
-2
1
1.5
2
2.5
3
3.5
4

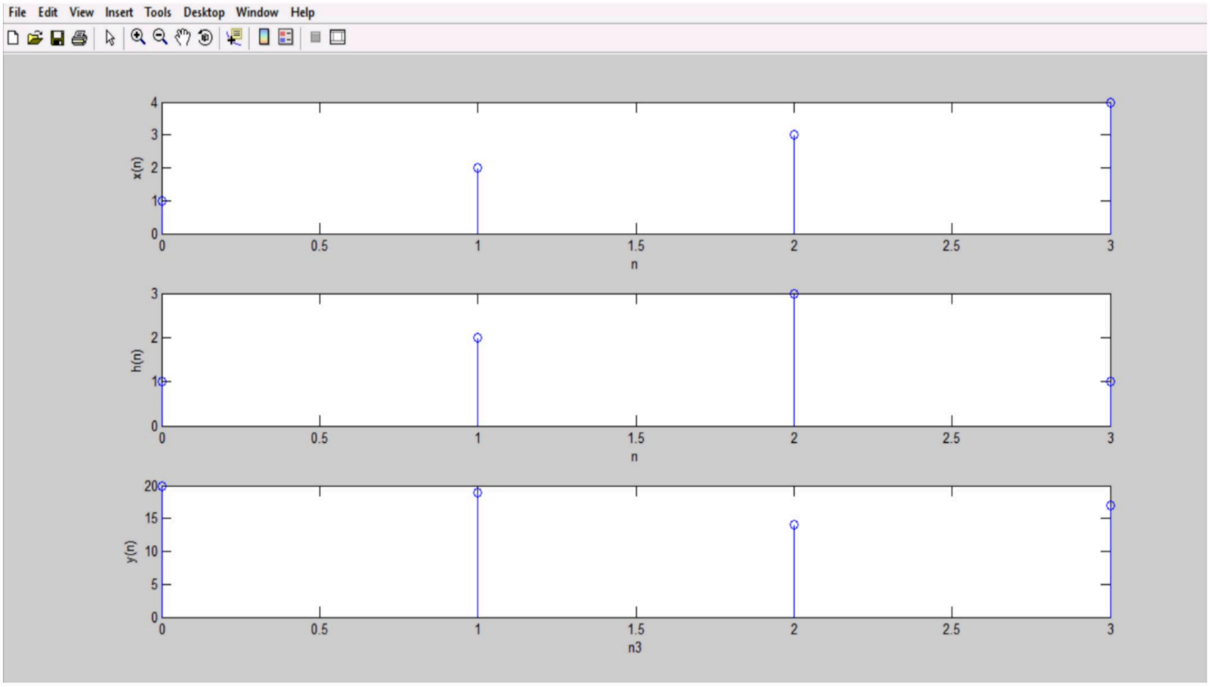
//dft using C

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
 void main()
{
     float k,x[10],sum1,sum2,n,y1[10],y2[10];
     int N;
     clrscr();
     printf("/n enter the length of dft sequence:");
     scanf("%d",&N);

     for(n=0;n<N;n++)
        {
         printf("\nEnter the input sequence");
         scanf("%f",&x[n]);
        }
         //Calculation of DFT
     for(k=0;k<N;k++)
           {
             sum1=sum2=0;
             for(n=0;n<N;n++)
           {
             sum1=  sum1+x[n]*cos(2*3.14*k*n/N);
             sum2=sum2-x[n]*sin(2*3.14*k*n/N);
             }
              y1[k]=sum1;
              y2[k]=sum2;

     printf("\n y[%f]= %f + j(%f)",k,y1[k],y2[k]);
     }
     getch();
}
```

Circular Using Matlab

```
clc;
x=input('enter input sig 1')
N1=length(x);
h=input('enter input sig 2')
N2=length(h);
n1=0:1:N1-1;
subplot(3,1,1);
stem(n1,x);
xlabel('n');
ylabel('x(n)');
n2=0:1:N2-1;
subplot(3,1,2);
stem(n2,h);
xlabel('n');
ylabel('h(n)');
y=cconv(x,h,4)
n3=0:1:3;
subplot(3,1,3);
stem(n3,y);
xlabel('n3');
ylabel('y(n)');
title('circular convolution')
```

Output



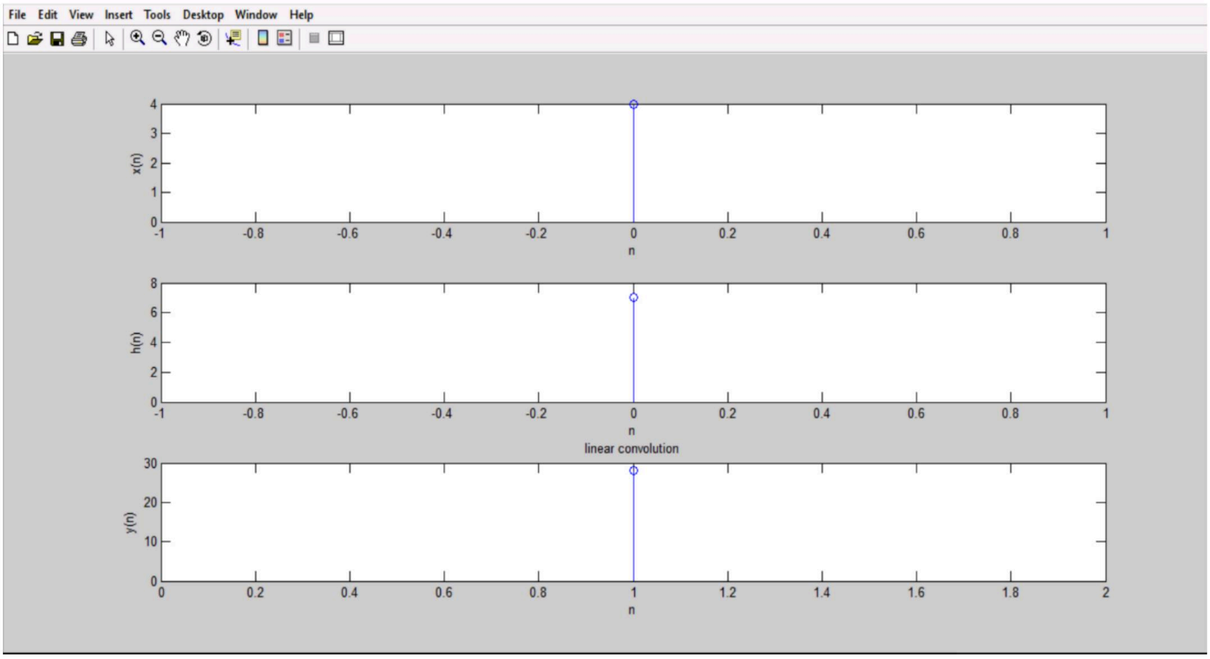
Circular Using C

```
#include<stdio.h>
#include<conio.h>
#define pf printf
#define sf scanf
void main(void)
{
int a,i,j,k,x[10],h[20],c,t[10][10],sum=0,y[10];
clrscr();
pf("\nenter the length of 2 signals");
sf("%d",&a);
pf("\nenter the sequence of x");
for(i=0;i<a;i++)
sf("%d",&x[i]);
pf("\nenter the sequence h");
for(i=0;i<a;i++)
{
sf("%d",&h[i]);
}
for(i=0;i<a;i++)
{
for(j=0;j<a;j++)
t[j][i]=h[j];
c=h[a-1];
for(k=a-2;k>=0;k--)
h[k+1]=h[k];
h[0]=c;
}
pf("Circular convolution is");
for(i=0;i<a;i++)
{
sum=0;
for(j=0;j<a;j++)
sum=sum+(t[i][j]*x[j]);
y[i]=sum;
printf("\n\n%d",y[i]);
}
getch();
}
```

Linear Convolution Using Matlab

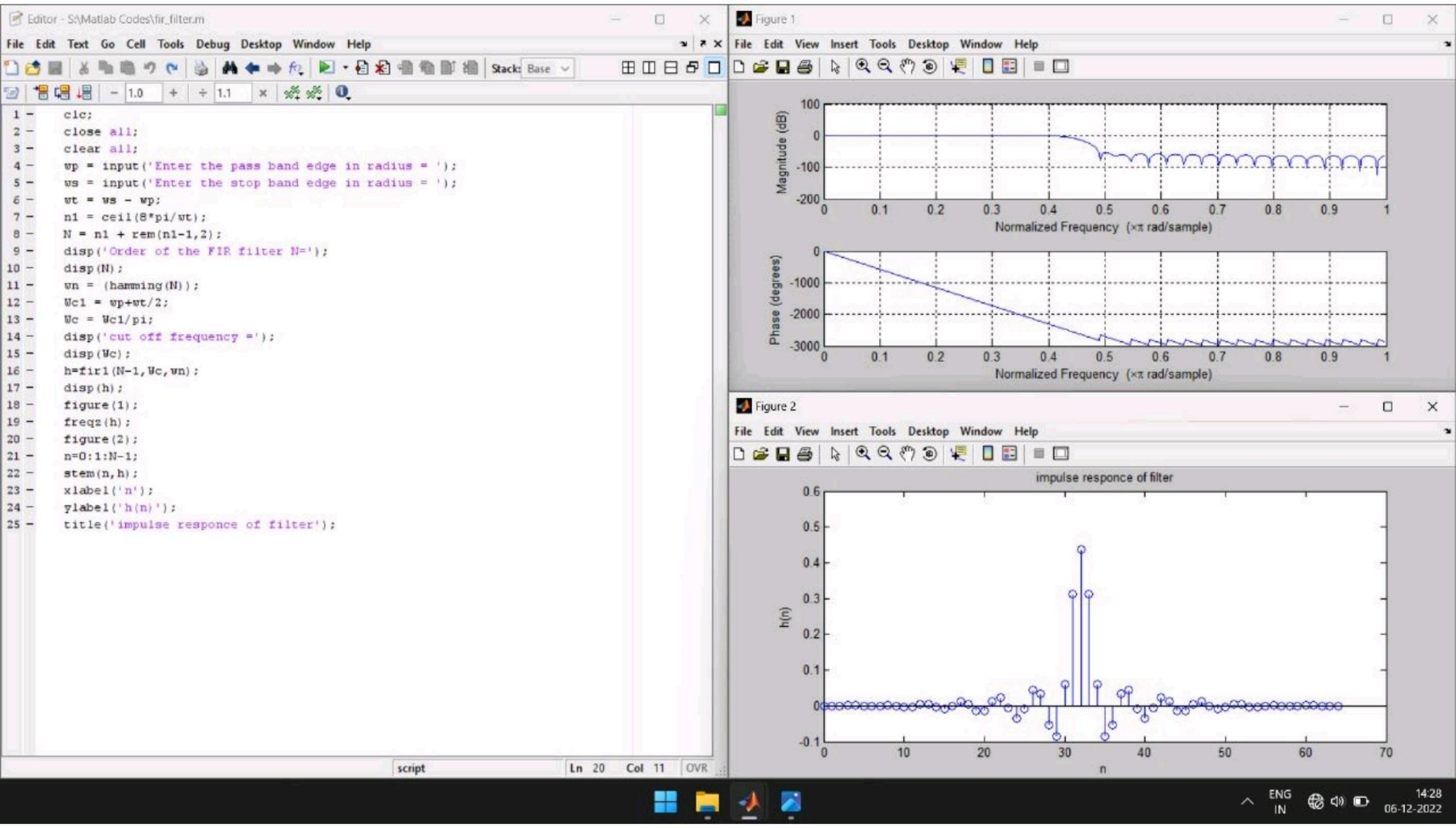
```
clc;
x=input('enter input signal-1[]')
N1=length(x);
n1=0:1:N1-1;
subplot(3,1,1)
stem(n1,x)
xlabel('n')
ylabel('x(n)')
h=input('enter input sig 2')
N2=length(h);
n2=0:1:N2-1;
subplot(3,1,2)
stem(n2,h)
xlabel('n')
ylabel('h(n)')
y=conv(x,h)
n3=1:N1+N2-1;
subplot(3,1,3)
stem(n3,y)
xlabel('n');
ylabel('y(n)');
title('linear convolution')
```

Output



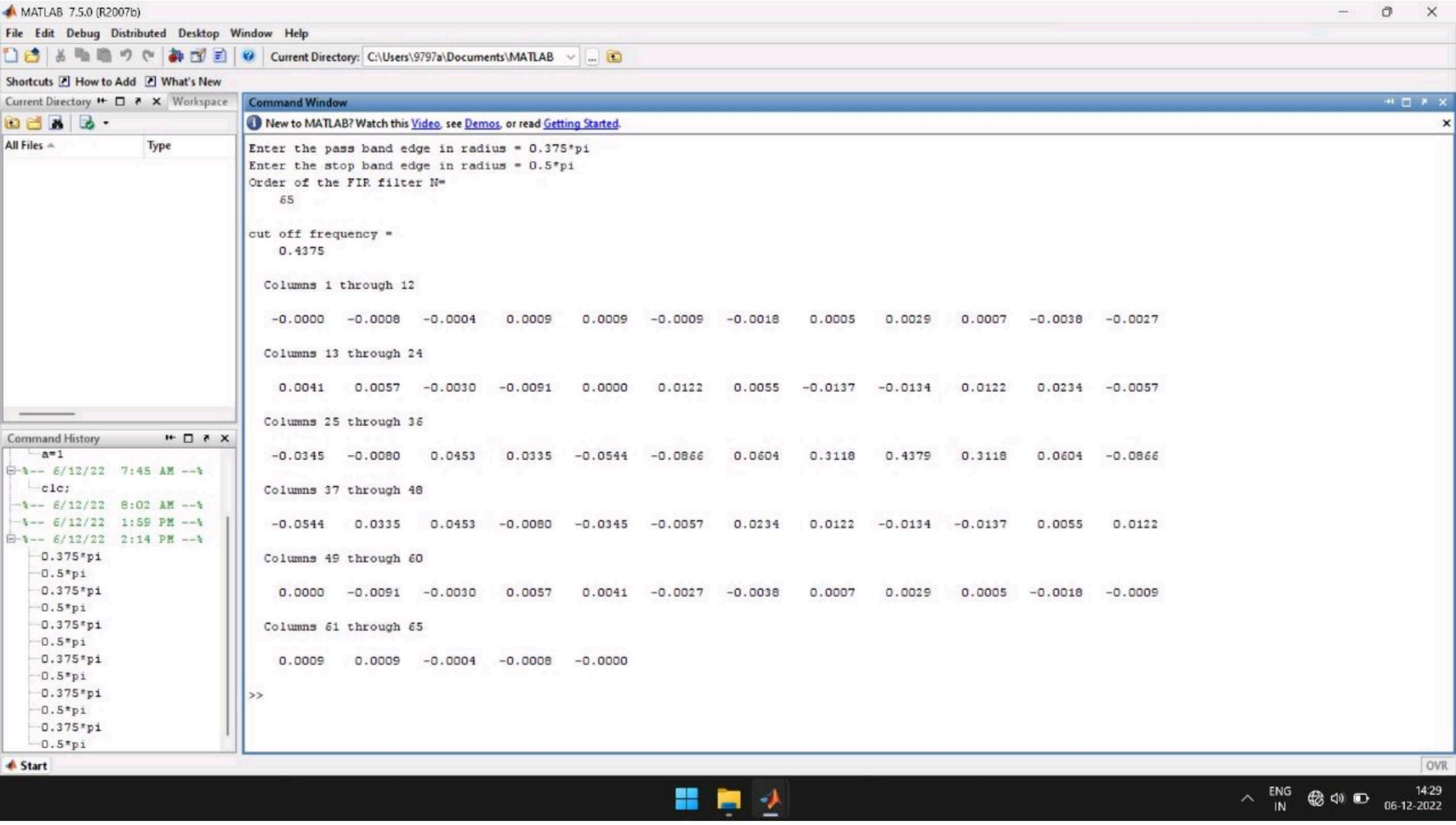
Linear Convolution Using C

```
#include<stdio.h>
#include<conio.h>
#define pf printf
#define sf scanf
void main()
{
int a,sum,b,i,j,k,x[10],y[20],h[10],n;
clrscr();
pf("\nenter the length of 1st signal");
sf("%d",&a);
pf("\nenter the length of 2nd signal");
sf("%d",&b);
pf("\nenter the value of 1st signal");
for(i=0;i<a;i++)
{
sf("%d",&x[i]);
}
pf("\n enter the values of 2nd signal");
for(i=0;i<b;i++)
{
sf("%d",&h[i]);
}
for(i=a;i<(a+b-1);i++) \\ Length of convolved signal
n=a+b-1
{
h[i]=0;
}
for(i=a-1;i>0;i--)
{
x[-i]=x[i];
x[i]=0;
}
for(i=a;i<(a+b-1);i++)
{
x[i]=0;
x[-i]=0;
}
pf("\n\n\n");
sum=0;
pf("\n The convolved signal is");
for(n=0;n<(a+b-1);n++)
{
for(k=0;k<(a+b-1);k++)
{

sum=sum+h[k]*x[-(n-k)];
}
y[n]=sum;


pf("\n%d",y[n]);
sum=0;
}
getch();
}
```



Editor - S:\Matlab Codes\fir_filter.m
File Edit Text Go Cell Tools Debug Desktop Window Help
Stack: Base
1.0
1.1
script
Ln 20 Col 11 OVR
Figure 1
File Edit View Insert Tools Desktop Window Help
Magnitude (dB)
100
0
-100
-200
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
Normalized Frequency (×π rad/sample)
Phase (degrees)
0
-1000
-2000
-3000
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
Normalized Frequency (×π rad/sample)
Figure 2
File Edit View Insert Tools Desktop Window Help
impulse responce of filter
h(n)
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.1
0 10 20 30 40 50 60 70
n
ENG
IN
14:28
06-12-2022

```
clc;
close all;
clear all;
wp = input('Enter the pass band edge in radius = ');
ws = input('Enter the stop band edge in radius = ');
wt = ws - wp;
n1 = ceil(8*pi/wt);
N = n1 + rem(n1-1,2);
disp('Order of the FIR filter N=');
disp(N);
wn = (hamming(N));
Wc1 = wp+wt/2;
Wc = Wc1/pi;
disp('cut off frequency =');
disp(Wc);
h=fir1(N-1,Wc,wn);
disp(h);
figure(1);
freqz(h);
figure(2);
n=0:1:N-1;
stem(n,h);
xlabel('n');
ylabel('h(n)');
title('impulse responce of filter');
```


MATLAB 7.5.0 (R2007b)
File Edit Debug Distributed Desktop Window Help
Current Directory: C:\Users\9797a\Documents\MATLAB
Shortcuts How to Add What's New
Current Directory Workspace
All Files Type
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
Command History
Start
OVR
ENG IN
14:29
06-12-2022

```
Enter the pass band edge in radius = 0.375*pi
Enter the stop band edge in radius = 0.5*pi
Order of the FIR filter N=
    65

cut off frequency =
    0.4375

  Columns 1 through 12

   -0.0000   -0.0008   -0.0004    0.0009    0.0009   -0.0009   -0.0018    0.0005    0.0029    0.0007   -0.0038   -0.0027

  Columns 13 through 24

    0.0041    0.0057   -0.0030   -0.0091    0.0000    0.0122    0.0055   -0.0137   -0.0134    0.0122    0.0234   -0.0057

  Columns 25 through 36

   -0.0345   -0.0080    0.0453    0.0335   -0.0544   -0.0866    0.0604    0.3118    0.4379    0.3118    0.0604   -0.0866

  Columns 37 through 48

   -0.0544    0.0335    0.0453   -0.0080   -0.0345   -0.0057    0.0234    0.0122   -0.0134   -0.0137    0.0055    0.0122

  Columns 49 through 60

    0.0000   -0.0091   -0.0030    0.0057    0.0041   -0.0027   -0.0038    0.0007    0.0029    0.0005   -0.0018   -0.0009

  Columns 61 through 65

    0.0009    0.0009   -0.0004   -0.0008   -0.0000

>>
```

```
  a=1
%-- 6/12/22  7:45 AM --%
  clc;
%-- 6/12/22  8:02 AM --%
%-- 6/12/22  1:59 PM --%
%-- 6/12/22  2:14 PM --%
  0.375*pi
  0.5*pi
  0.375*pi
  0.5*pi
  0.375*pi
  0.5*pi
  0.375*pi
  0.5*pi
  0.375*pi
  0.5*pi
  0.375*pi
  0.5*pi
```