EXPERIMENT NO-3

CODE-

```
#Circular convolution using DFT
import numpy as np
from numpy.fft import fft,ifft
x=[1,2,3,4]
h=[1,2]


N1=len(x)
N2=len(h)


N=max(N1,N2)
m=N-N1
p=N-N2


if N1>N2:
   h=np.pad(h,(0,p),'constant')
if N2>N1:
   x= np.pad(x,(0,m),'constant')
print('\n x1(n)=',x)
print('\n x2(n)=',h)


XX=fft(x)
HX=fft(h)
```

```
YX=XX*HX
y=ifft(YX)

print('\n Circular convolution of x(n) and h(n)=',y)
```

OUTPUT-

```
x1(n)= [1, 2, 3, 4]
x2(n)= [1 2 0 0]
Circular convolution of x(n) and h(n)= [ 9.+0.j  4.+0.j  7.+0.j 10.+0.j]
```

CODE-

```
#Linear convolution using DFT
import numpy as np
from numpy.fft import fft,ifft
x=[1,2,3,4]
h=[1,2]

N1=len(x)
N2=len(h)
N=N1+N2-1

m=N-N1
p=N-N2
```

```
h=np.pad(h,(0,p),'constant')


x= np.pad(x,(0,m),'constant')

print('\n x1(n)=',x)

print('\n x2(n)=',h)


XX=fft(x)

HX=fft(h)

YX=XX*HX

y=ifft(YX)


print('\n Linear convolution of x(n) and h(n)=',y)
```

OUTPUT-

```
x1(n)= [1 2 3 4 0]

x2(n)= [1 2 0 0 0]

Linear convolution of x(n) and h(n)= [ 1.+0.j  4.+0.j  7.+0.j 10.+0.j  8.+0.j]
```