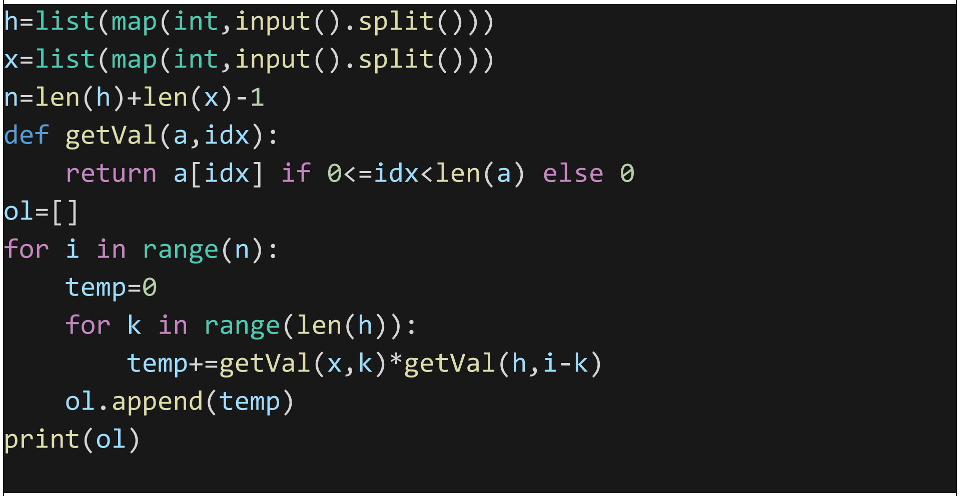
Name:Hayden Cordeiro **BECOMPS(D)** **Roll No:05**

Experiment 2

Title	Discrete Convolution
Aim	The aim of this experiment is to study mathematical operation such as Linear convolution, Circular convolution, Linear convolution using circular convolution.
Objective	1. Develop a function to find Linear Convolution and Circular Convolution 2. Calculate Linear Convolution, Circular Convolution, Linear Convolution using Circular Convolution and verify the results using mathematical formulation. 3. Conclude on aliasing effect in Circular convolution
Input Specifications	1.Length of first Signal L and signal values. 2.Length of second Signal M and signal values.
Problem Definition	1. Find Linear Convolution and Circular Convolution of L point sequence x[n] and M point sequence h[n]. 2. Find Linear Convolution of L point sequence x[n] and M point sequence h[n] using Circular convolution. 3. Give your conclusion about no. of values in linearly convolved signal, and Aliasing effect in Circular Convolution.

Program

1)Linear Convolution



```python
h=list(map(int,input().split()))
x=list(map(int,input().split()))
n=len(h)+len(x)-1
def getVal(a,idx):
    return a[idx] if 0<=idx<len(a) else 0
ol=[]
for i in range(n):
    temp=0
    for k in range(len(h)):
        temp+=getVal(x,k)*getVal(h,i-k)
    ol.append(temp)
print(ol)
```

2) Circular Convolution

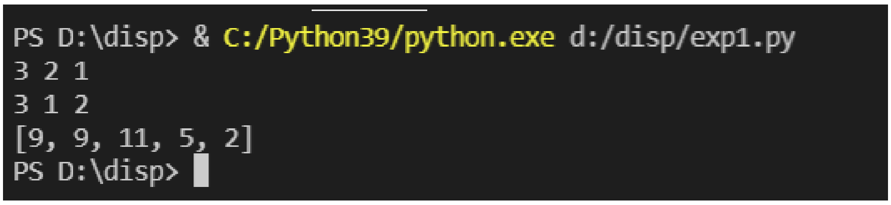
```
x1=[0,0,3,-1,0]
x2=[1,1,1,0,0]
print("Enter Inputs")
# x1=list(map(int,input().split(",")))
# x2=list(map(int,input().split(",")))
x1.extend([0 for i in range(len(x2)-len(x1))])
x2.extend([0 for i in range(len(x1)-len(x2))])
id1=[i for i in range(len(x1))]
id2=[i for i in range(len(x1))]
# initial
id2.reverse()

print("len of output seq",len(x1))
print()
for ik in range(len(id1)):
  temp=id2[-1]
  id2.pop()
  id2.insert(0,temp)
  temp=0
  print("y{}=".format(ik),end=" ")
  for i,j in zip(id1,id2):
    print("{}*{}".format(x1[i],x2[j]),end=" + ")
    temp+=x1[i]*x2[j]
  print(" = {}".format(temp))
```

3)Linear with Circular Convolution

```
x1=[2,1,-2]
x2=[1,2,-1]
print("Enter Inputs")
# x1=list(map(int,input().split(",")))
# x2=list(map(int,input().split(",")))
n=len(x1)+len(x2)-1
x1.extend([0 for i in range(n-len(x1))])
x2.extend([0 for i in range(n-len(x2))])
id1=[i for i in range(len(x1))]
id2=[i for i in range(len(x1))]
# initial
id2.reverse()
print("len of output seq",n)
print()
for ik in range(n):
  temp=id2[-1]
  id2.pop()
  id2.insert(0,temp)
  temp=0
  print("y{}=".format(ik),end=" ")
  for i,j in zip(id1,id2):
    print("{}*{}".format(x1[i],x2[j]),end=" + ")
    temp+=x1[i]*x2[j]
  print(" = {}".format(temp))
```

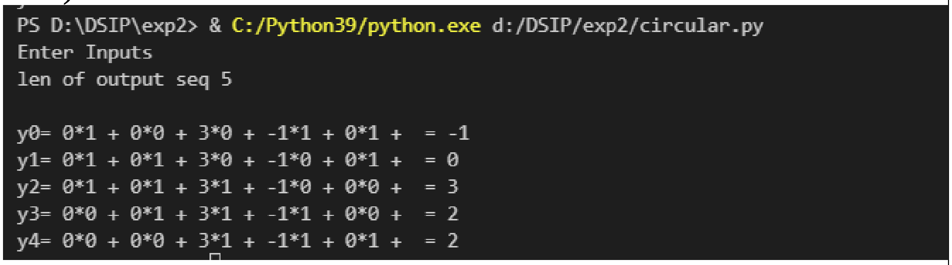
Output

1) Linear Convolution



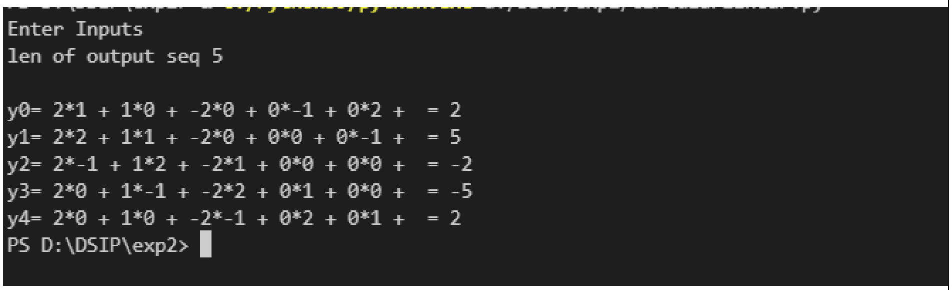
```
PS D:\disp> & C:/Python39/python.exe d:/disp/exp1.py
3 2 1
3 1 2
[9, 9, 11, 5, 2]
PS D:\disp> 
```

2) Circular Convolution



```
PS D:\DSIP\exp2> & C:/Python39/python.exe d:/DSIP/exp2/circular.py
Enter Inputs
len of output seq 5

y0= 0*1 + 0*0 + 3*0 + -1*1 + 0*1 +  = -1
y1= 0*1 + 0*1 + 3*0 + -1*0 + 0*1 +  = 0
y2= 0*1 + 0*1 + 3*1 + -1*0 + 0*0 +  = 3
y3= 0*0 + 0*1 + 3*1 + -1*1 + 0*0 +  = 2
y4= 0*0 + 0*0 + 3*1 + -1*1 + 0*1 +  = 2
```

3) Linear with Circular Convolution



```
Enter Inputs
len of output seq 5

y0= 2*1 + 1*0 + -2*0 + 0*-1 + 0*2 +  = 2
y1= 2*2 + 1*1 + -2*0 + 0*0 + 0*-1 +  = 5
y2= 2*-1 + 1*2 + -2*1 + 0*0 + 0*0 +  = -2
y3= 2*0 + 1*-1 + -2*2 + 0*1 + 0*0 +  = -5
y4= 2*0 + 1*0 + -2*-1 + 0*2 + 0*1 +  = 2
PS D:\DSIP\exp2> 
```

Outcome

Thus, we could implement linear convolution, circular convolution and linear convolution using circular convolution with the help of python.