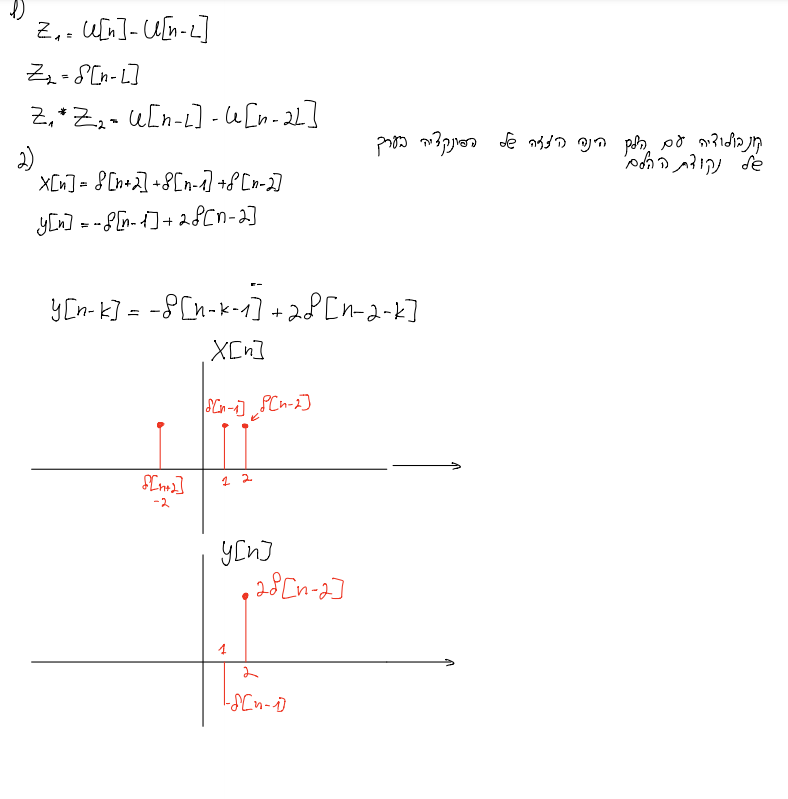
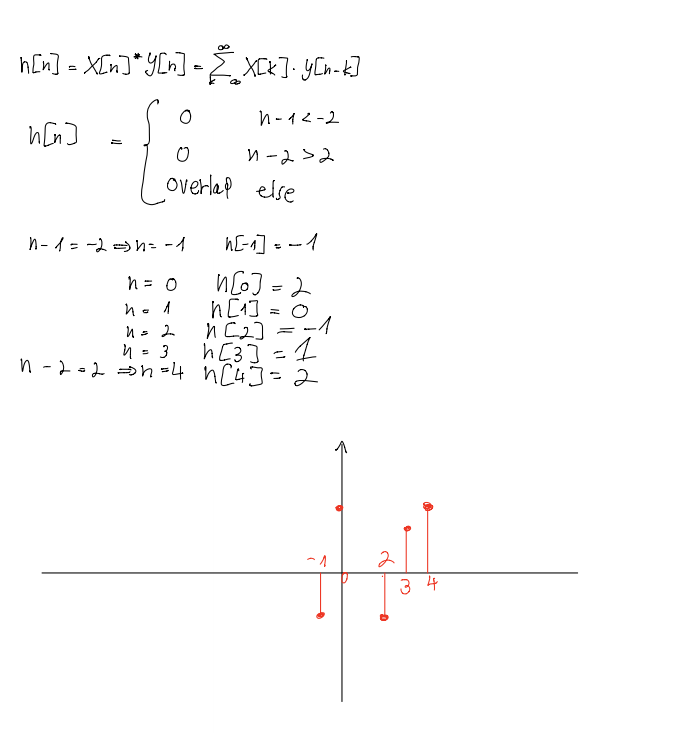
H.W 2

רום הירש – 313288763

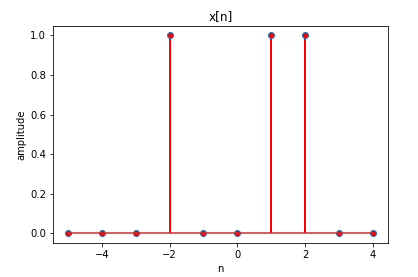
שאלות 1 + 2 :

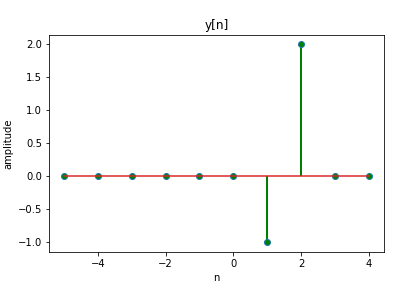


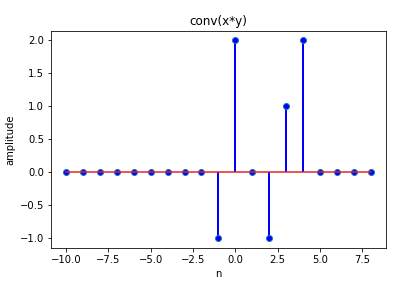


שאלה 3 :

להלן הplot הקוד כתוב למטה במסמך







שאלה 4 :

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

שאלה 6 + 5 :

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

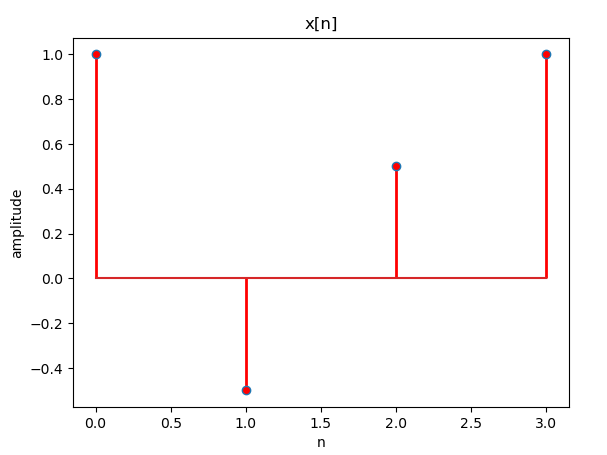
תמונה שמכילה טקסט

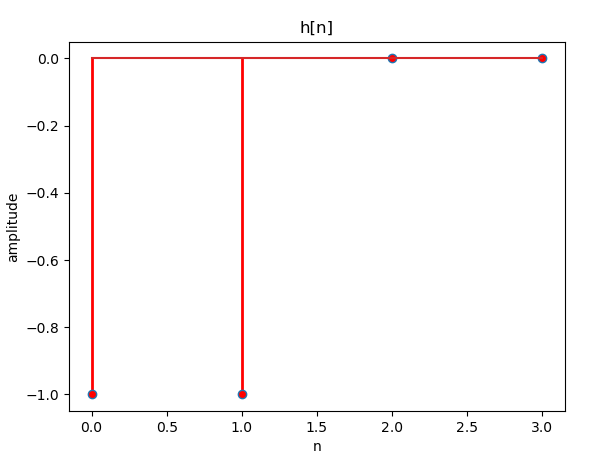
התיאור נוצר באופן אוטומטי

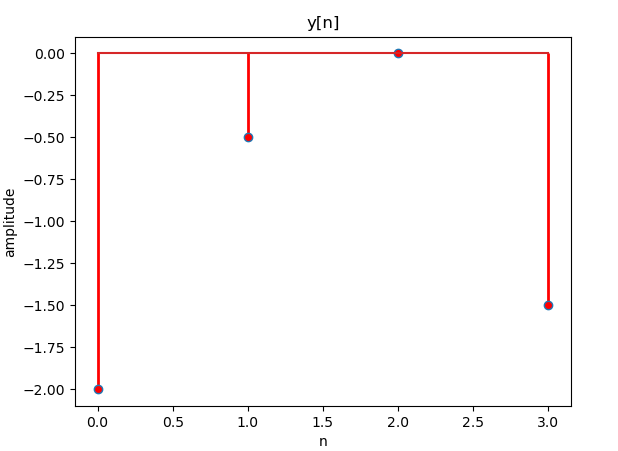
שאלה 7 –

להלן הplot הקוד כתוב למטה במסמך

\*כתבתי את הפונקציה של (ccov) cycle cov כי לא מצאתי פונקציה מוכנה בpython







קוד – python שאלה 3 ו7 –

1 import matplotlib.pyplot as plt

2 from scipy import signal

3 import numpy as np

4 #%%

5 """

6 Q -

7

8 x[n]=delta[n+2]+delta[n-1]+delta[n-2]

9 y[n]=-delta[n-1]+2delta[n-2]

10 """

11

12 Vlen = 10 #len vector

13 #plot stem with color

14 def stem\_plot(n,val,color):

15 markerline1, stemlines1, baseline1 = plt.stem(n,val)

16 plt.setp(markerline1, 'markerfacecolor', color)

17 plt.setp(stemlines1, linestyle="-", color=color, linewidth=2 )

18

19

20 #create Delta

21 def dirac(val=0):

22 zero = Vlen/2

23 return signal.unit\_impulse(Vlen,int(zero-val))

24

25 def plotStem(title,ylabel,xlabel,color,x,y):

26 plt.figure()

27 plt.title(title)

28 plt.ylabel(ylabel)

29 plt.xlabel(xlabel)

30 stem\_plot(x,y,color)

31 plt.show()

32

33 ncov = np.arange(-10,9,1)

34 n = np.arange(-5,5,1)

35 delta = signal.unit\_impulse(8)

36 x = dirac(2)+dirac(-1)+dirac(-2) # x[n]=delta[n+2]+delta[n-1]+delta[n-2]

37 y = -dirac(-1)+2\*dirac(-2) # y[n]=-delta[n-1]+2delta[n-2]

38 plotStem("x[n]","amplitude","n",'red',n,x)

39 plotStem("y[n]","amplitude","n",'green',n,y)

40 convXY = np.convolve(x,y)

41 plotStem("conv(x\*y)","amplitude","n",'blue',ncov,convXY)

42 #%%

43 """

44 Q - 7

45

46 x=[1 -0.5 0.5 1];

47 h=[-1 -1 0 0];

48

49 """

50 #function

51

52 def padded\_zeros(arr,N): # padded zeros

53 padded\_array = np.zeros(N)

54 padded\_array[:len(arr)]+=arr

55 return padded\_array

56

57

58 #cycle Covolation

59 def ccov(x,y):

60 #check which array more large and pandded with zeros the others

61 N=len(x)

62 if len(x)>len(y):

63 N=len(x)

64 y = padded\_zeros(y,N)

65 else:

66 N=len(y)

67 x = padded\_zeros(x,N)

68 h = np.zeros(N) # check h[n] vector

69 for n in range(N):

70 for k in range(N):

71 h[n]+=y[k]\*x[(n-k)%N]

72 return h

73

74 n = np.arange(0,4,1)

75 x=np.array([1,-0.5,0.5,1])

76 h=np.array([-1,-1,0,0])

77 y = ccov(x,h) # cycle convolation

78 plotStem("x[n]","amplitude","n",'red',n,x)

79 plotStem("h[n]","amplitude","n",'red',n,h)

80 plotStem("y[n]","amplitude","n",'red',n,y)