HW 4

2-24

$$x_1[n]=(\frac{1}{2})^nu[n],y_1[n]=2^nu[n]-(\frac{1}{2})^nu[n]$$
$$x_2[n]=2(\frac{1}{2})^nu[n],y_2[n]=3\cdot 2^nu[n]-2(\frac{1}{2})^nu[n]$$

(1)

$$y[n]=y_{zi}[n]+y_{zs}[n]$$
$$x_1[n]=2x_2[n]$$
$$y_1[n]=y_{zi}[n]+y_{1zs}[n]$$
$$y_2[n]=y_{zi}[n]+y_{2zs}[n]$$
$$y_{zi}[n]=2y_1[n]-y_2[n]=-2^nu[n]$$

(2)

$$x_3[n]=0.5(\frac{1}{2})^nu[n]$$
$$y_3[n]=y_{zi}[n]+y_{3zs}[n]$$
$$y_{3zs}[n]=\frac{y_2[n]-y_1[n]}{2}=2^nu[n]-(\frac{1}{2})^{n+1}u[n]$$
$$y_3[n]=-(\frac{1}{2})^{n+1}u[n]$$

3-1

(1):$x(t)=E$

$$x(t)=E=E\cdot e^{jw0}$$
$$y(t)=E$$

(2)

$$x(t)=\sum_{k=-10}^{10}a_ke^{jkw_0t},w_0=\pi$$
$$\text{由于}w>5\text{以及}w<-5\text{时}H(jw)=0$$
$$\text{仅考虑}k\in[-4,4]\text{时的情况}$$
$$H(jw)=1-\frac{|w|}{5\pi}$$
$$w=k\pi$$
$$y(t)=H(jk\pi)a_k=\sum_{k=-4}^{4}(1-\frac{|k|}{5})a_k$$

3-2

(1):$x(t) = cos2t + sin4t$

$$cos(2t)\text{的傅里叶变换}:x(t) = \frac{e^{j2t}+e^{-j2t}}{2}$$

$$sin(4t)\text{的傅里叶变换}:x(t) = \frac{e^{j4t}-e^{-j4t}}{2j}$$

$$sin(2t)+cos(4t)\text{的傅里叶变换}:x(t) = \frac{e^{j2t}+e^{-j2t}}{2}+\frac{e^{j4t}-e^{-j4t}}{2j}$$

(2)

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jkw_0t}$$

$$w_0 = \frac{2\pi}{T_0}$$

$$a_k = \frac{1}{T_0}\int_{T_0} x(t)e^{-jkw_0t}dt$$

$$T_0 = T, w_0 = \frac{2\pi}{T}$$

$$a_k = \int_{-\frac{T_0}{2}}^{\frac{T_0}{2}} x(t)e^{-jkw_0t}dt$$

$$a_k = \frac{1}{T}[\int_{-\frac{T}{2}}^{0} -\frac{E}{2}e^{-jkw_0t}dt + \int_{0}^{\frac{T}{2}} \frac{E}{2}e^{-jkw_0t}dt]$$

$$k \neq 0\text{时}, a_k = \frac{E[1-(-1)^k]}{j2k\pi}$$

$$k = 0\text{时}, a_k = 0$$

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\frac{2\pi}{T_0}t}$$

(3)

$$a_k = \frac{1}{T}[\int_{-\frac{T}{4}}^{\frac{T}{4}} \frac{E}{2}e^{-jkw_0t}dt + \int_{\frac{T}{4}}^{\frac{3T}{4}} -\frac{E}{2}e^{-jkw_0t}dt$$

$$a_k = \frac{1}{T}[\frac{E}{2}\cdot\frac{1}{-jkw_0}(2e^{-jkw_0\frac{T}{4}} - e^{jkw_0\frac{T}{4}} - e^{-jkw_0\frac{3T}{4}})]$$

$$k \neq 0\text{时}, a_k = \frac{E}{-4jk\pi}(2e^{-jk\frac{\pi}{2}} - e^{jk\frac{\pi}{2}} - e^{-jk\frac{3\pi}{2}})$$

$$k = 0\text{时}, a_k = 0$$

(4)

$$a_k = \frac{1}{T}[\int_{-\frac{T}{2}}^{\frac{T}{2}} \frac{E}{T}te^{-jkw_0t}dt]$$

$$k \neq 0\text{时}, a_k = \frac{j(-1)^kE}{2k\pi}$$

$$k = 0\text{时}, a_k = 0$$

(5)

$$a_k = \frac{1}{T}\int_0^2 x(t)e^{-jkw_0t}dt$$

$$T = 4, w_0 = \frac{2\pi}{T} = \frac{\pi}{2}$$

$$a_k = \frac{1}{4}[\int_0^1 2e^{-jk\frac{\pi}{2}t}dt + \int_1^2 e^{-jk\frac{\pi}{2}t}dt]$$

$$k \neq 0\text{时}, a_k = \frac{2 - e^{-jk\frac{\pi}{2}} - e^{-jk\pi}}{j2k\pi}$$

$$k = 0\text{时}, a_k = \frac{3}{4}$$

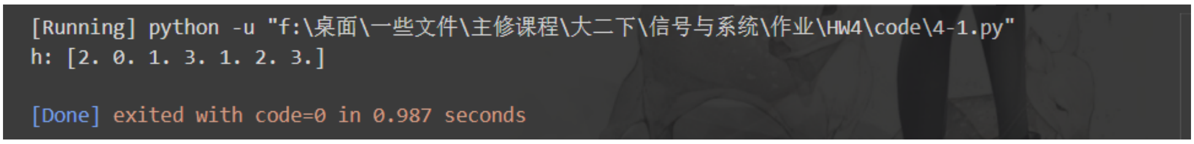
编程作业

1

使用以下程序进行计算

```
from random import randrange
import numpy as np

def calculateInpuseResponse(x,y):
    h = np.zeros(len(y) - len(x) + 1)
    xMatrix = np.zeros((len(h),len(y)))
    for i in range(len(h)):
        xMatrix[i,i:i+len(x)] = x

    xMatrix1 = np.zeros((len(h),len(h)))
    for i in range(len(h)):
        for j in range(len(h)):
            xMatrix1[i][j] = xMatrix[i][j]

    y1 = np.zeros(len(h))
    for i in range(len(h)):
        y1[i] = y[i]
    try:
        xInv = np.linalg.inv(xMatrix1)
    except:
        print("No sequence satisfies this condition")
    #xInv * y1
    for i in range(len(h)):
        for j in range(len(h)):
            h[i] += xInv[j][i] * y1[j]
    return h

def main():
    h = np.array([2,0,1,3,1,2,3])
    x = np.array([3,2,1,3])
    y = np.convolve(x,h)
    res = calculateInpuseResponse(x,y)
    print("h:",res)

main()
```

运算结果如图：



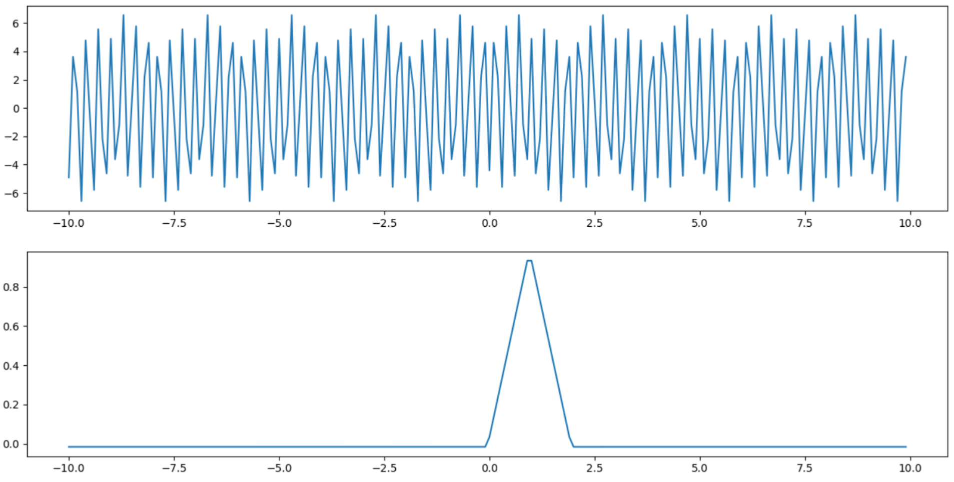
```
[Running] python -u "f:\桌面\一些文件\主修课程\大二下\信号与系统\作业\HW4\code\4-1.py"
h: [2. 0. 1. 3. 1. 2. 3.]

[Done] exited with code=0 in 0.987 seconds
```

2

使用如下代码进行计算

```
import numpy as np
import matplotlib.pyplot as plt
from sympy import *
import math

def myIntegrate(t):
    sum = 0
    for j in np.arange(-1000,0,0.1):
        if(j == 0):
            y = 0.25
        else:
            y = (2/math.pi)*(math.sin(j*0.5)**2/(j*math.sin(j)))*math.cos(j*t)
        sum += y*0.1
    return sum

def approximate():
    x = np.arange(-10,10,0.1)
    y = []
    for i in np.arange(-10,10,0.1):
        y.append(myIntegrate(i))
    return x,y

def convTest():
    y = []
    for i in np.arange(-10,10,0.1):
        sum = 0
        for j in np.arange(0,2,0.1):
            sum += myIntegrate(i-j)*0.1
        y.append(sum)
    return y

def main():
    x,y = approximate()
    xTest = np.arange(-10,10,0.1)
    yTest = convTest()
    fig = plt.figure()
    ax1 = fig.add_subplot(211)
    ax2  = fig.add_subplot(212)
    ax1.plot(x,y)
    ax2.plot(xTest,yTest)
    plt.show()

main()
```

运算结果如图：


6
4
2
0
−2
−4
−6
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
0.8
0.6
0.4
0.2
0.0
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0