Signal & System 2022 期末考模擬

⼿做答案可以⼿寫拍照，

程式考題寫出程式碼以及列印執⾏結果。

⼀並彙整為⼀個pdf檔上傳。

將請助教開⼀個 E-Learning 作業於考試當天(6⽉17⽇)限時(2.5⼩時)之內上傳。

正式考試為第17周上課時間(6⽉17⽇15:00~18:00)

(1)

What is a linear chirp signal? Write down its mathematical function.

答案：

(2)

Write a function called myChirp(), it can generate a chirp signal.

input:

a time sequence: ts= [0, 0.0001, 0.0002, ..., T] in (sec)

a start frequency f0 (Hz),

a ending frequency f1 (Hz).

output:

the chirp sequence: ys= myChirp(ts)

def myChirp(ts, f0=100, f1=400):

#

# your code ...

#

return ys

# 1. generate a time sequence from 0 sec to 10 sec

# 2. test your function.

# Get thinkdsp.py

答案：

def myChirp(ts, f0=100, f1=400):

signal = Chirp(start=100, end=400)

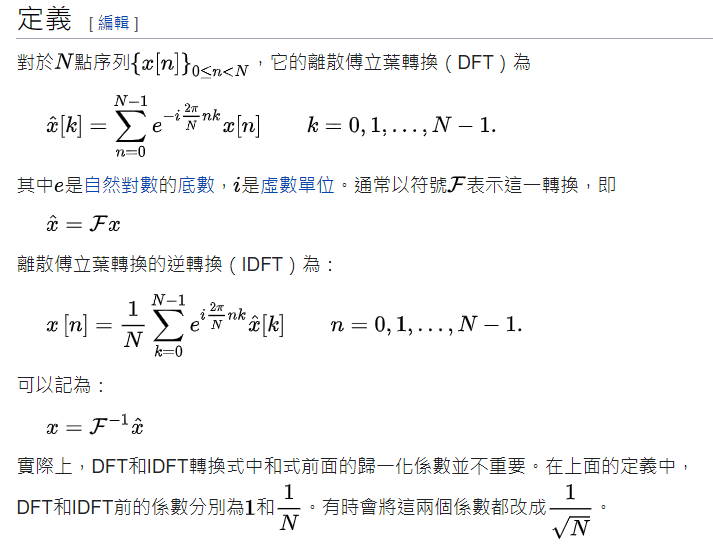
ys = signal.make\_wave(duration=10)

ys.make\_audio()

return ys

(3) what is DFT (Discrete Fourier Transform), and IDFT (Inverse Discrete Fourier Transform), write

down their mathematical definition.



(4) Given a time-domian sequence x[n]= [1,1,1,1], please calculate the DFT X[k]. and also

compute y[n], which is the IDFT of X[k].

(5)

Write a function called myDft(), it can compute the DFT of any signal x[n] or the IDFT of any

spectrum X[k.

input:

a time-domain signal: x= [x0, x1,... x[N-1]]

or a frequency-domain spectrum: X= [X0, X1,... X[N-1]]

output:

a spectrum X[k] or a signal x[n]

def myDft(x, inverst= False):

#

# your code ...

#

#

#

return X

# 1. generate a time-domain signal x[n] or a time-domain spectrum X[k]

# 2. test your function.

(6) If x is a Cosine signal with frequency = 100 Hz, generate a discrete signal with sampling

frequency, Fs = 1000 samples/sec, for 10 sec. compute the spectrum X[k] of the the signal x[n]

and plot X[k], what does it look like?

(7) what is convolution of 2 signal x[n] and y[n], please write down their mathematical definition.

(8) if x= [1,1,1,1], y=[1,2,3,4,5], what is their convolution z = x\*y?

z=[1,2,3]

z=60

(9) write a function called myConvlution(), which can compute the convolution of two signal.

def myConvlution(x,y):

#

# your code to implement

# z = x \* y

#

return z

答案：

def myConvlution(x,y):

z=0;

lenx=len(x);

leny=len(y);

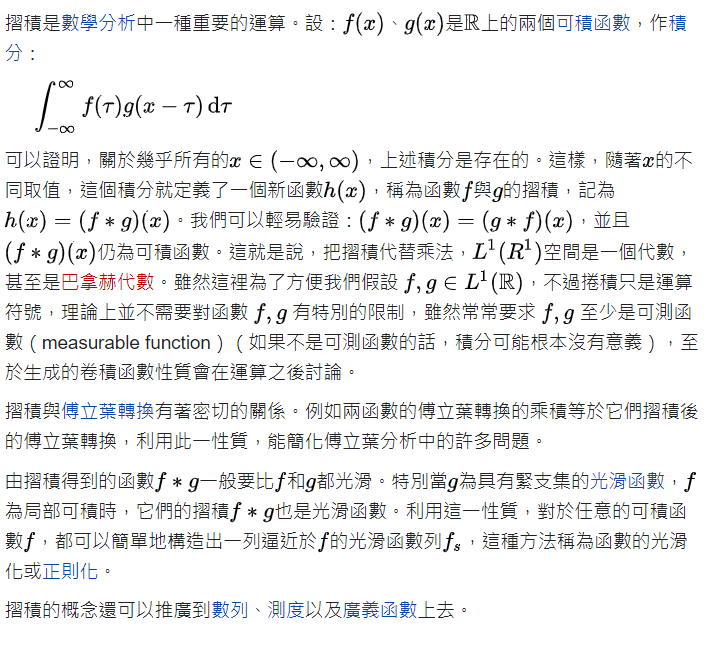
for i in len(x):

for j in len(y):

z=z+i\*j;

return z

(10) what is the convolution theorem?



(11) Verify the convolution theorem by using myDft() and myConvolution()