B0829011 王紹丞 資工­二

錯誤題目從4.1～4.4

心得:

要好好看考卷才不會大小寫搞錯，且變數的命名的規則要熟悉，一般class皆為大寫開頭。

(1.1)

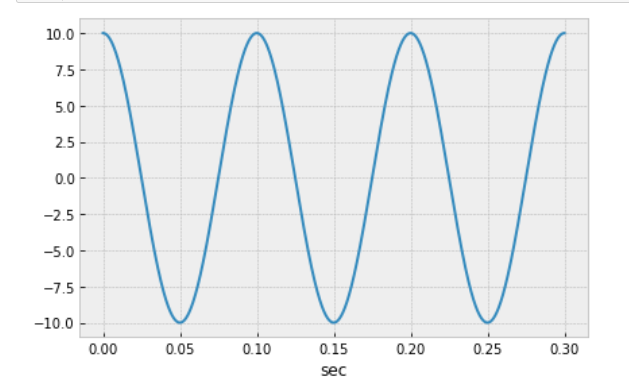
cos1 = CosSignal(freq=10, amp=10.0, offset=0)

from thinkdsp import decorate

plt.grid(True)

cos1.plot()

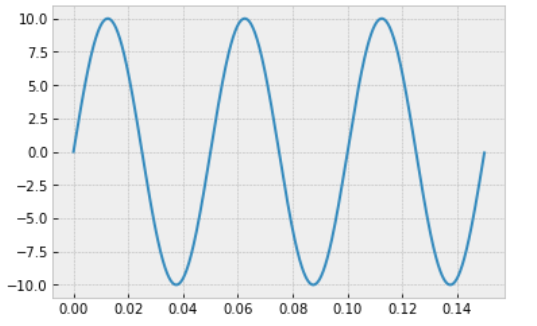
decorate(xlabel='sec')



(1.2)

sin1 = SinSignal(freq=20, amp=10, offset=0)

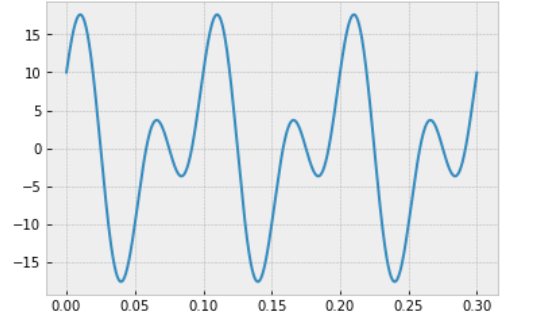
sin1.plot()



(1.3)

mix = sin1 + cos1

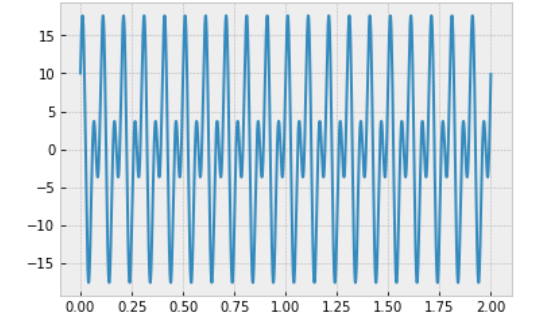
mix.plot()



(1.4)

wave = mix.make\_wave(duration=2, start=0, framerate=11025)

wave.plot()



(1.5)

會有1000\*2=2000個資料點

因為framerate 是指每秒幾個 frame，也就是每秒幾個 sample 的意思，而這張圖的範圍為2s(duration=2)故1000\*2 因為sample rate 設為1000。

wave.duration\*wave.framerate

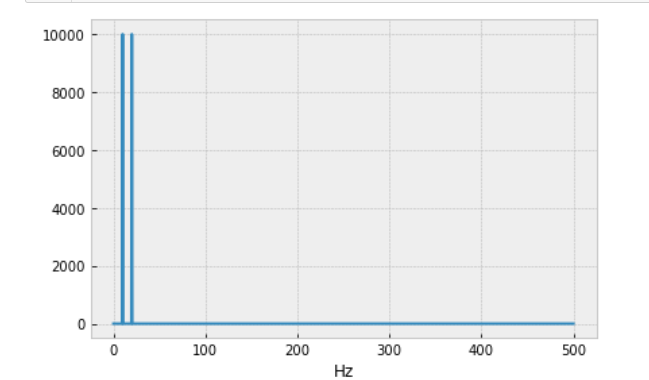


(1.6)

spectrum1 = wave.make\_spectrum()

spectrum1.plot()

decorate(xlabel='Hz')



(1.7)

sin1 = SinSignal(freq=1000, amp=2, offset=0)

sin2 = SinSignal(freq=2000, amp=2, offset=0)

sin3 = SinSignal(freq=3000, amp=2, offset=0)

sin4 = SinSignal(freq=4000, amp=2, offset=0)

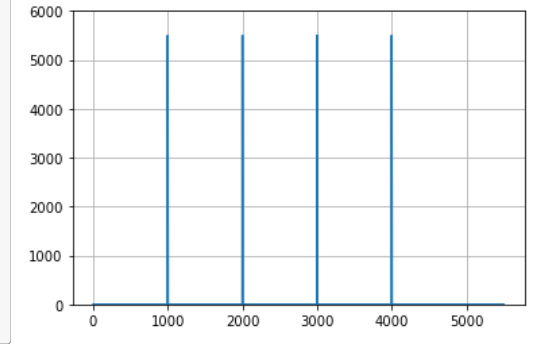
mix=sin1+sin2+sin3+sin4

wave = mix.make\_wave(duration=0.5, framerate=11000)

spectrum= wave.make\_spectrum()

spectrum.plot()

plt.grid(True)

plt.ylim((0, 6000)) 

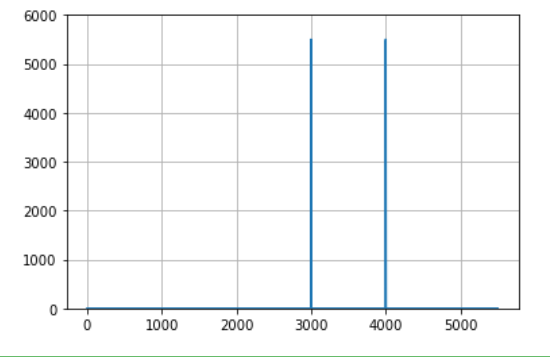
(1.8.1)

spectrum.high\_pass(3000)

spectrum.plot(high=6000)

plt.ylim((0, 6000))

plt.grid(True)



(1.8.2)

sin1 = SinSignal(freq=1000, amp=2, offset=0)

sin2 = SinSignal(freq=2000, amp=2, offset=0)

sin3 = SinSignal(freq=3000, amp=2, offset=0)

sin4 = SinSignal(freq=4000, amp=2, offset=0)

mix=sin1+sin2+sin3+sin4

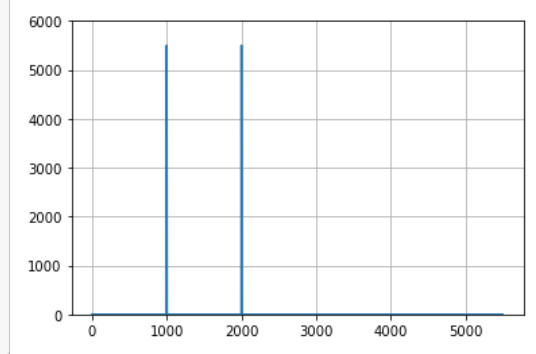
wave = mix.make\_wave(duration=0.5, framerate=11000)

spectrum= wave.make\_spectrum()

spectrum.low\_pass(2001)

spectrum.plot()

plt.grid(True)

plt.ylim((0, 6000)) 

(2.1)

from thinkdsp import SquareSignal

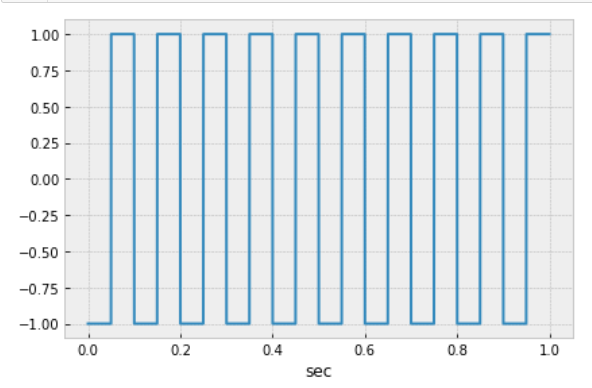
signal = SquareSignal(10)

duration = signal.period\*10

segment = signal.make\_wave(duration, framerate=10000)

segment.plot()

decorate(xlabel='sec')



(2.2)

def rect\_square\_wave(origin=0, size=20, ratio=0.5, x\_unit=1, y0=0, y1=1):

d = []

for start in np.arange(origin, origin + x\_unit \* size, x\_unit):

d.append((start, y0))

middle = start + x\_unit \* ratio

d.append((middle, y0))

d.append((middle, y1))

end = start + x\_unit

d.append((end, y1))

return d

import numpy as np

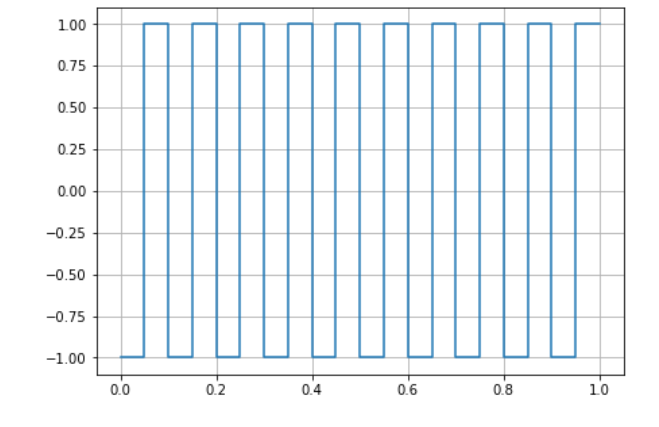
import matplotlib.pyplot as plt

d = np.array(rect\_square\_wave(origin=0, size=10, ratio=0.5, x\_unit=0.1, y0=-1, y1=1))

plt.figure(figsize=(7, 5))

plt.plot(d[:, 0], d[:, 1])

# plt.plot(d[:, 1], d[:, 1])



(3.1)

Aliasing為聲音被折疊，取樣頻率低於2倍最高訊號頻率而導致失真、原先的弦波被錯誤的取樣弦波取代，這個現象稱為aliasing，舉例當你framerate設為10000時，此時對稱點為5000，若頻率超過5000的都會被折疊，像是6000會被折疊到4000因為6000跟5000差1000而5000-1000為4000，若折疊後的值小於0那麼則要再對折，像是11000為-1000再對折為1000。

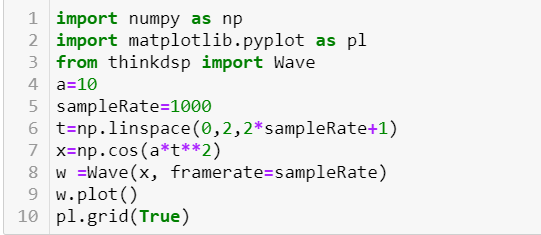
(3.2)

當取樣頻率為原訊號之最高頻率之兩倍以上時，就可以正確的重建原始訊號，進而避免Aliasing

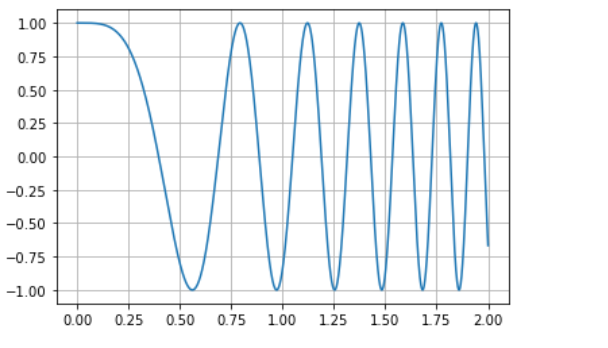
error:

當時在期中考時把thinkdsp的Wave clsss的W打成小寫，因此在from thinkdsp import wave時一直秀出error沒有wave object

(4.1)



(4.2)



(4.3)

第3行:from thinkdsp import Wave

第8行: w =Wave(x, framerate=sampleRate)

(4.4)

spectrogram = w.make\_spectrogram(seg\_length=512)

spectrogram.plot(high=30)

