L-Università ta' Malta
Faculty of Information &
Communication Technology

Assignment Report

Manwel Bugeja

November 19, 2019

Contents

0.1 Question 1

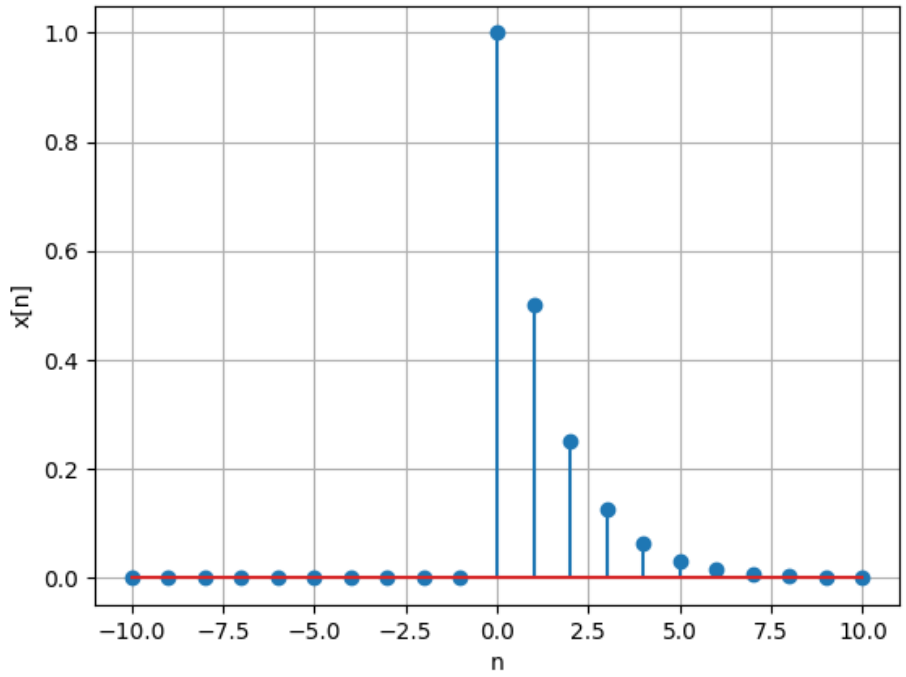


Figure 1: Question 1

0.2 Question 2

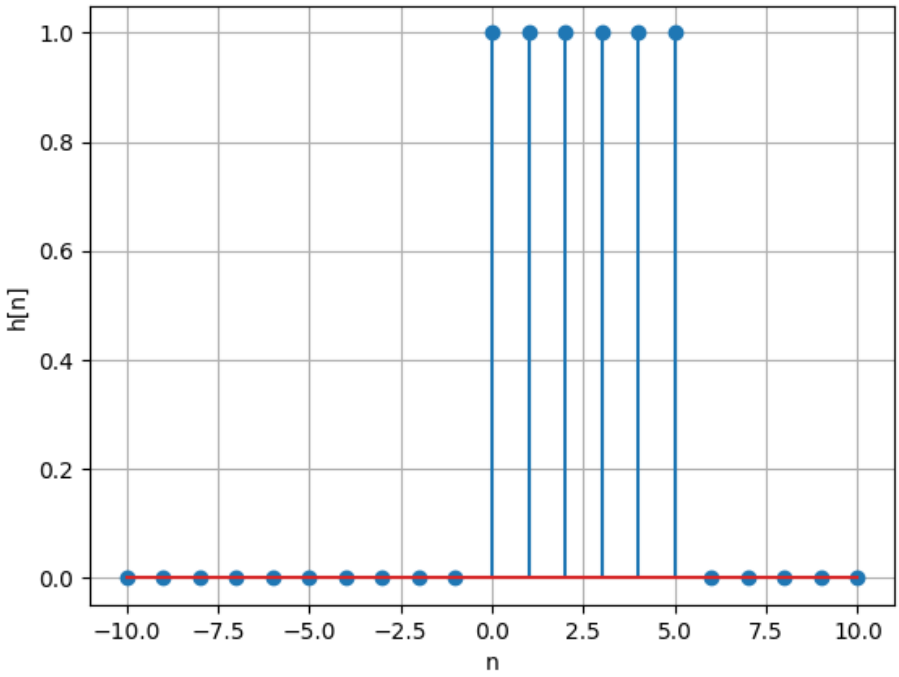


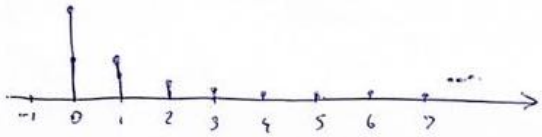
Figure 2: Question 2

Lab 2 question 3

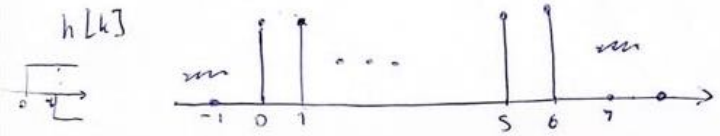
$x[n] = (0.5)^n u[n]$

$h[n] = u[n] - u[n-6]$

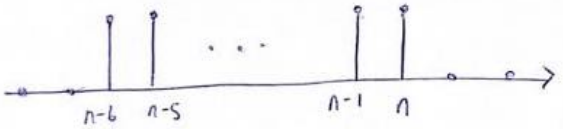
$x[k]$:

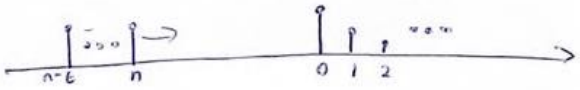


$h[k]$



$h[n-k]$





case 0: $n < 0$

$y[n] = 0$

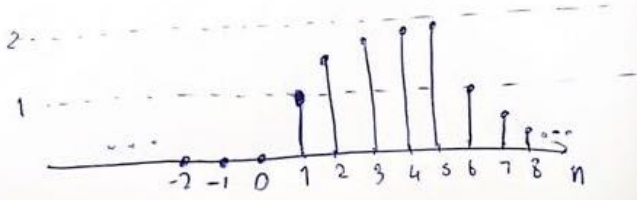
case 1: $n \geq 0 \wedge n-6 < 0$

i.e. $0 \leq n < 6$

$y[n] = \sum_{0}^{n} (0.5)^n$

case 2: $n \geq 6$

$y[n] = \sum_{n-6}^{n} (0.5)^n$



0.3 Question 3

0.4 Question 4

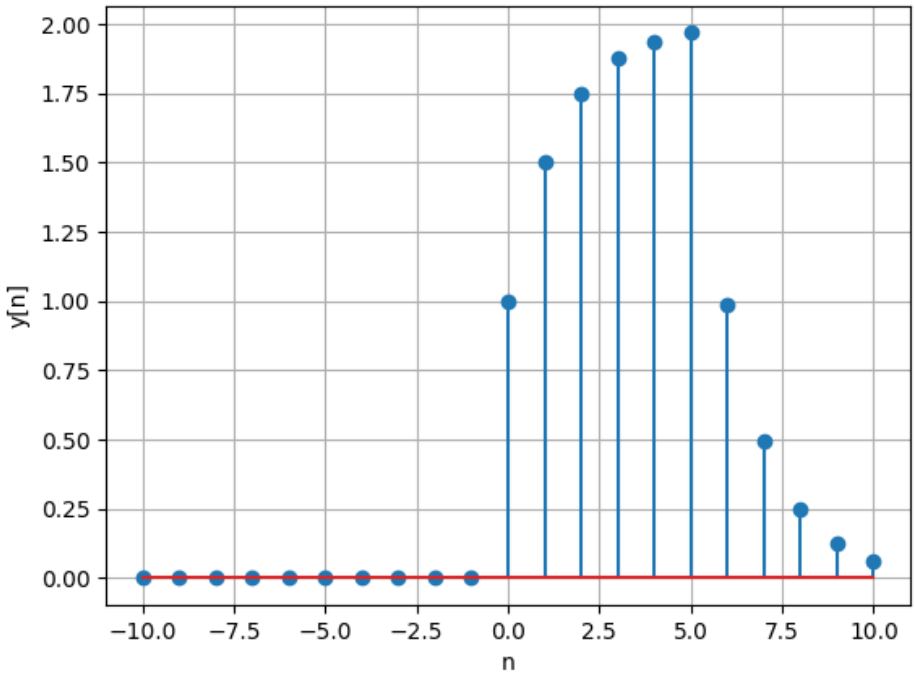


Figure 3: Question 4

0.5 Question 5

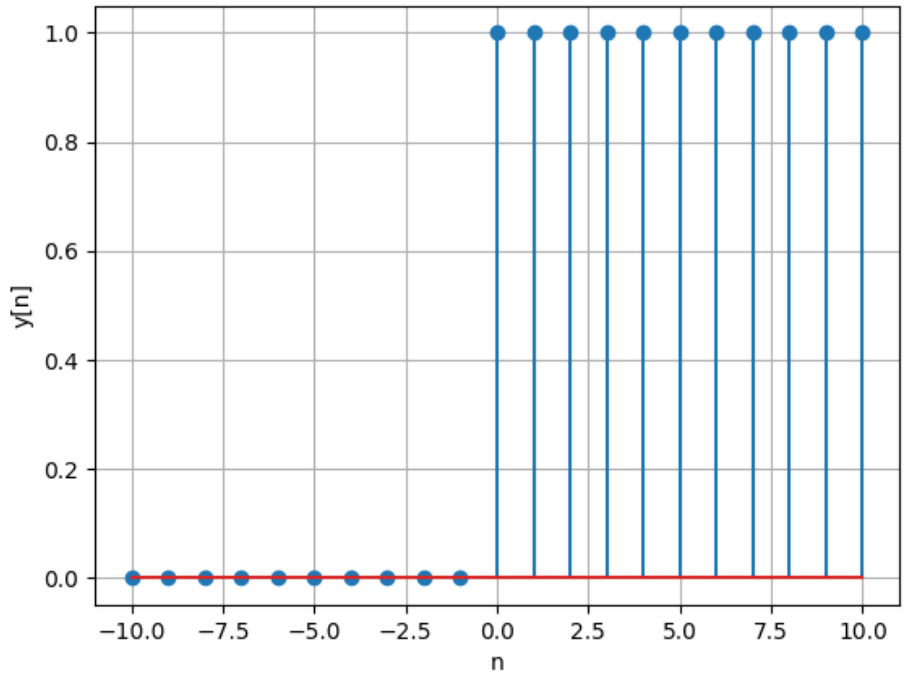


Figure 4: Question 5

0.6 Question 6

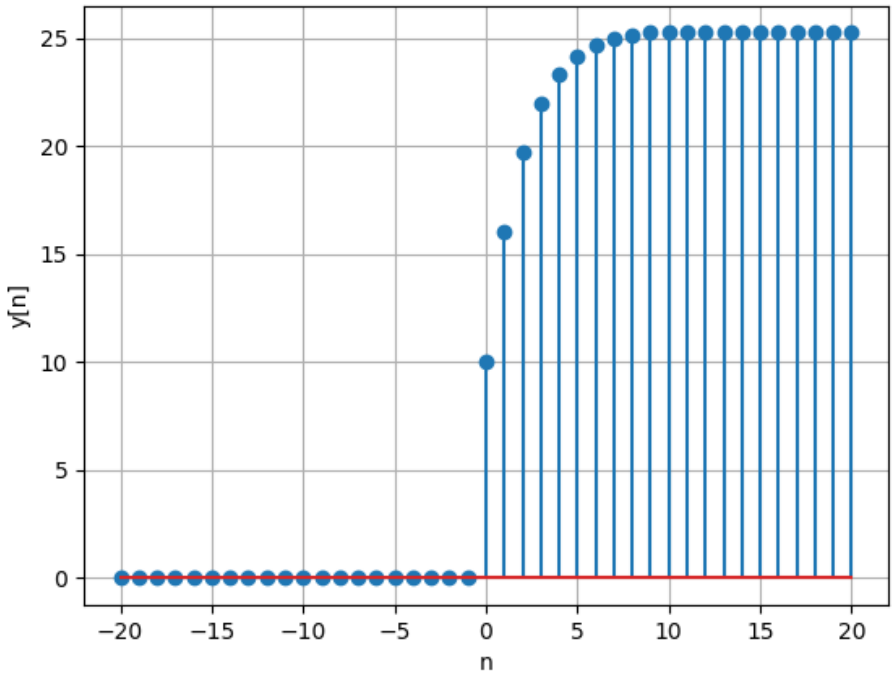


Figure 5: Question 6

0.7 Question 7

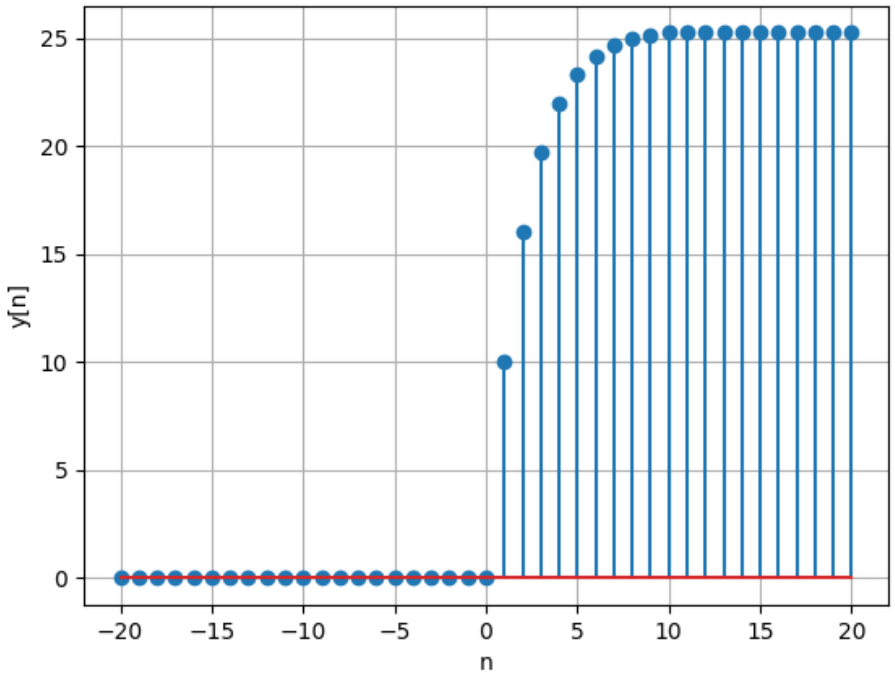


Figure 6: Question 7

0.8 Question 8

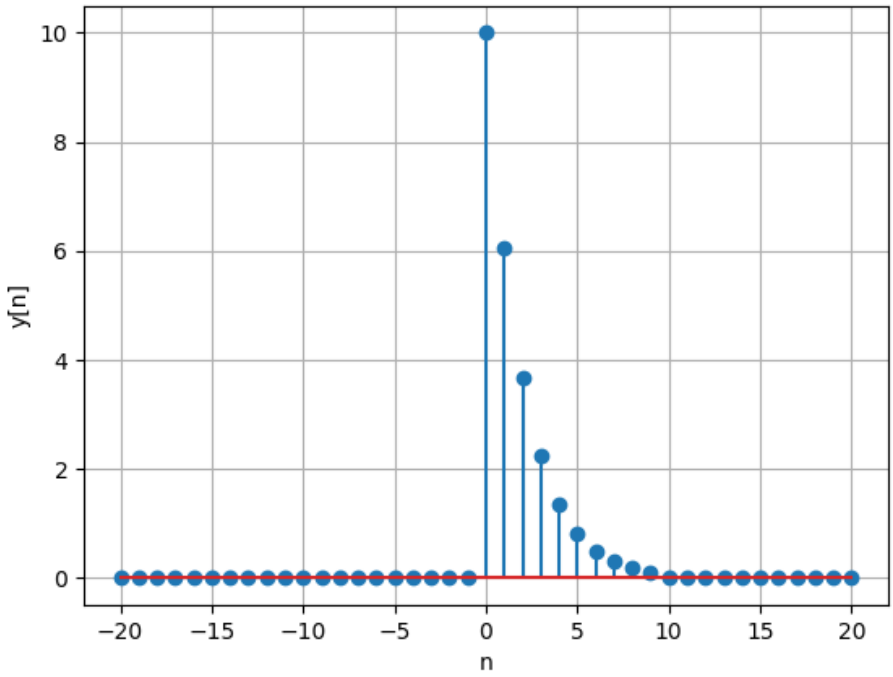


Figure 7: Question 8

0.9 Question 9

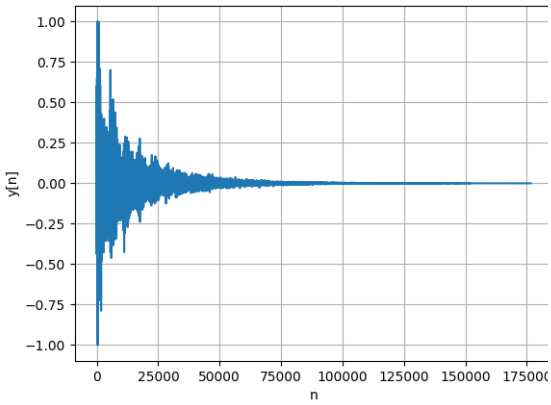


Figure 8: Question 9a

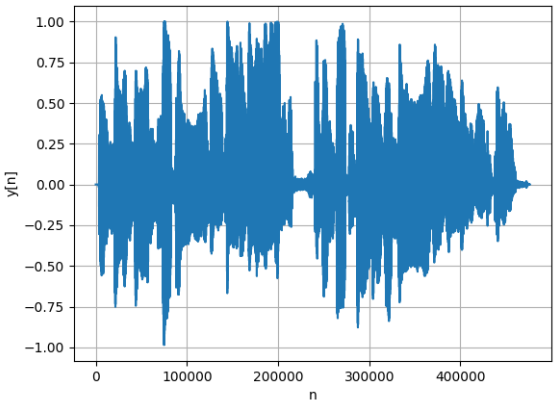


Figure 9: Question 9b

The figures are quite different when plotted as a graph as they contain different sounds.

0.10 Question 10

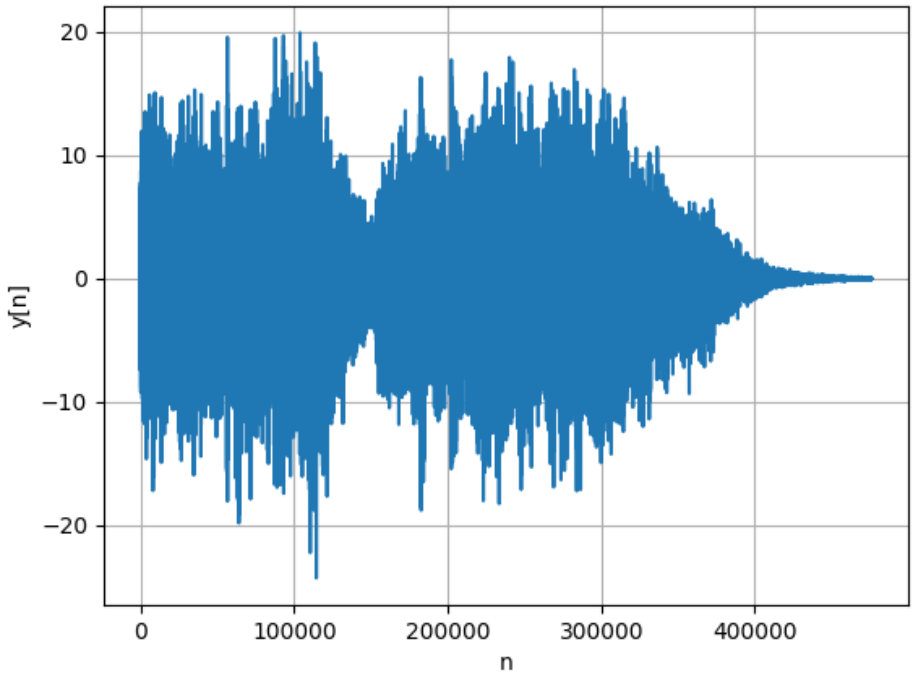


Figure 10: Question 10

This figure contains the convolution of the 'vocal' signal with the 'LongEchoHallIR' signal. The effects of the part with the lower amplitude in Question 9b can be seen in the convoluted signal as that part of the other signal also has a relativly low amplitude.

This was confirmed by writing the signal to a .wav file. This file contained both sounds of the signals in one file.

0.11 Question 12

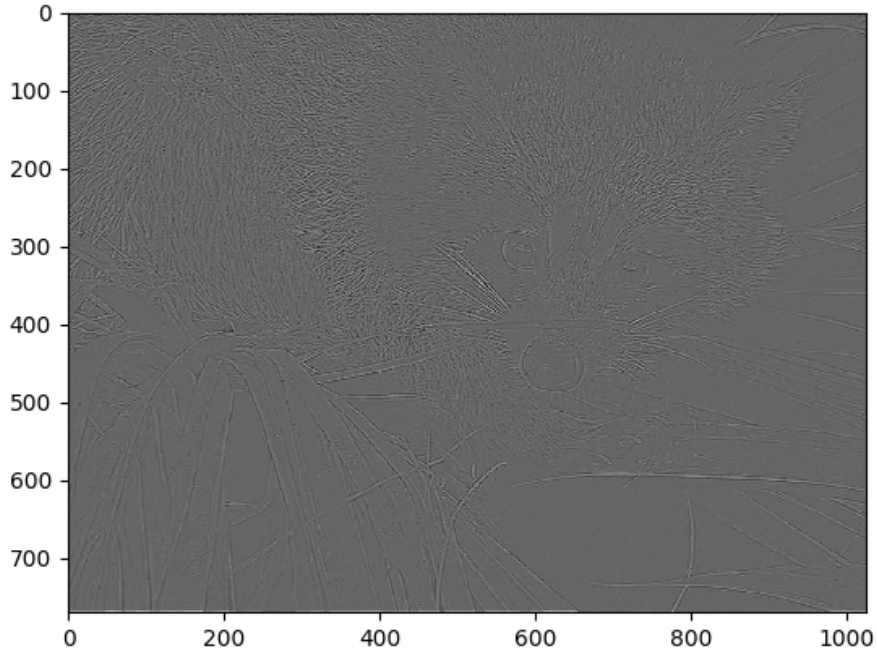


Figure 11: Question 12

Because of the given impulse response, the racoon can hardly be seen in the image.

0.12 Question 13

Figure 12: Original

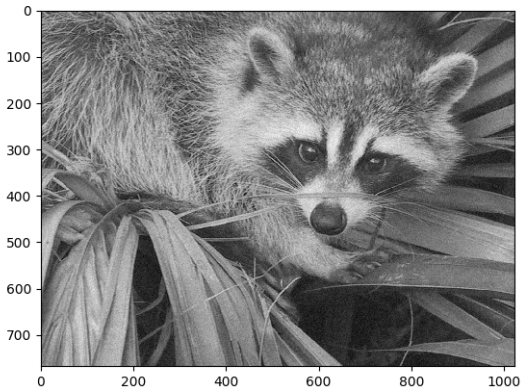


Figure 13: Question 13

Figure 14: Question 14

As expected, the new image has added noise which make the image less clear than the original.

0.13 Question 14

Figure 15: Question 14

When applying the given kernel to the noisy image, the noise is reduced and the clarity on the image is increased and becomes comparable with that of the original image.

0.14 Appendix

Listing 1: Code of part 2

```
from matplotlib import pyplot as plt
import numpy as np
import wavio
from scipy.io import wavfile
from scipy.io import loadmat
from scipy import signal
```

```
from scipy import misc


def make_u(start, length):
    u = np.zeros(length)
    u[n >= start] = 1
    return u


n = np.arange(-10, 11)

# question 1
# plotting x[n]
x = pow(0.5, n) * make_u(0, len(n))
plt.stem(n, x, use_line_collection=True)
plt.xlabel('n')
plt.ylabel('x[n]')
plt.grid()
plt.savefig("Question_01.png")
plt.clf()

# question 2
# plotting h([n]
h = make_u(0, len(n)) - make_u(6, len(n))
plt.stem(n, h, use_line_collection=True)
plt.xlabel('n')
plt.ylabel('h[n]')
plt.grid()
plt.savefig("Question_02.png")
plt.clf()

# question 4
# plotting y[n]
y = np.convolve(x, h, mode='save')
plt.stem(n, y, use_line_collection=True)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_04.png")
plt.clf()
```

Listing 2: Code of part 3

```
from matplotlib import pyplot as plt
import numpy as np
import wavio
from scipy.io import wavfile
from scipy.io import loadmat
from scipy import signal
from scipy import misc


def make_u(start, length):
    u = np.zeros(length)
    u[n >= start] = 1
    return u


n = np.arange(-10, 11)

# question 5
# plot u[n]
plt.stem(n, make_u(0, len(n)), use_line_collection=True
    )
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_05.png")
plt.clf()

# question 6
system = loadmat('unknown-system.MAT')
n = system['n'][0]
h = system['h'][0]

h6 = np.convolve(h, make_u(0, len(n)), mode='save')
plt.stem(n, h6, use_line_collection=True)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_06.png")
plt.clf()
```

```
# question 7
# plotting s[n-1] = h[n]*u[n-1]
s = np.convolve(h, make_u(1, len(n)), mode='save')
plt.stem(n, s, use_line_collection=True)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_07.png")
plt.clf()

# question 8
# plotting impulse response
h8 = h6 - s
plt.stem(n, h8, use_line_collection=True)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_08.png")
plt.clf()

# question 9
ir_filename = 'LongEchoHallIR'
audio_filename = 'vocal'


data = wavio.read("data/" + ir_filename + ".wav")
h = data.data[:,0]
h = np.reshape(h, len(h))
h = h/np.max(np.abs(h))

plt.plot(h)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_09a.png")
plt.clf()


data = wavio.read("data/" + audio_filename + ".wav")
h1 = data.data[:,0]
```

```
h1 = np.reshape(h1, len(h1))
h1 = h1/np.max(np.abs(h1))

plt.plot(h1)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_09b.png")
plt.clf()

# question 10
y = np.convolve(h, h1, mode='save')
plt.plot(y)
plt.xlabel('n')
plt.ylabel('y[n]')
plt.grid()
plt.savefig("Question_10.png")
plt.clf()

import wave
wave_read = wave.open("data/" + audio_filename + ".wav"
    , 'rb')
rate = wave_read.getframerate()
sample_width = wave_read.getsampwidth()

wavio.write("Question_10.wav", y, rate, sampwidth=
    sample_width)
```

Listing 3: Code of part 4

```
from matplotlib import pyplot as plt
import numpy as np
import wavio
from scipy.io import wavfile
from scipy.io import loadmat
from scipy import signal
from scipy import misc


def make_u(start, length):
    u = np.zeros(length)
```

```
    u[n >= start] = 1
    return u


img = misc.face()
img = np.dot(img[...,:3], [0.299, 0.587, 0.114])

plt.figure()
plt.imshow(img, cmap='gray')
plt.savefig('Question_13_original.png')

# question 12
h = [[0,-1,0],[-1,4,-1],[0,-1,0]]

y = signal.convolve2d(img, h)
plt.figure()
plt.imshow(y, cmap='gray')
plt.savefig('Question_12.png')


# question 13
def gauss_noise(image):
    row, col = image.shape
    mean = 0
    sigma = 10
    gauss = np.random.normal(mean, sigma, (row,col))
    gauss = gauss.reshape(row, col)
    noisy = image + gauss
    return noisy

img = gauss_noise(img)
plt.figure()
plt.imshow(img, cmap='gray')
plt.savefig('Question_13.png')

# question 14
val = (1/9)
h = [[val,val,val],[val,val,val],[val,val,val]]

y = signal.convolve2d(img, h)
plt.figure()
```

```
plt.imshow(y, cmap='gray')
plt.savefig('Question_14.png')
```