[redacted] [redacted] **Summary sheet**

Name : Kriuk Fedor Student ID : [redacted]

<u>Objective</u>

- To be familiar with impulse response and convolution [redacted]

<u>Part I</u> **(6)** **Include all useful codes of Part I in (e)**

Let A = the 5th digit of your student ID Let the input sequence $x[n]$ be the same as your student ID.

e.g. Student ID = 20345678 $x[n] = [2, 0, 3, 4, 5, 6, 7, 8]$ for **n = A to A+7**

Write **Matlab code** to define the input x and the index nx of the input.

The impulse responses of two LTI systems are shown below.

System 1 : $h1[n] = \delta[n] + \delta[n-2]$ System 2 : $h2[n] = u[n-A] - u[n-A-4]$

System 1 and System 2 are connected **in series** to form an overall system $h[n]$.
The input sequence $x[n]$ is applied to the overall system $h[n]$ to give the output $y[n]$.

a) Write down the mathematical expression of the overall impulse response $h[n]$ in terms of $h1[n]$ and $h2[n]$.

(1) $\boldsymbol{h[n]}$ = **h1[n] * h2[n]**

Write **Matlab code** to define h1. Write **Matlab code** to define h2 using "**ones**". **DO NOT use "dirac" and "heaviside".**
Write **Matlab code** to obtain the overall impulse response h according to the answer provided in (a).
Write **Matlab code** to define the index nh of the overall impulse response.

b) Fill in the following table after running your Matlab code.

(1) **Only show non-zero value of h and corresponding nh**

nh	**5**	**6**	**7**	**8**	**9**	**10**						
h	**1**	**1**	**2**	**2**	**1**	**1**						

Write **Matlab code** to obtain the output y and define the index ny of the output.
c) Fill in the following table after running your Matlab code.

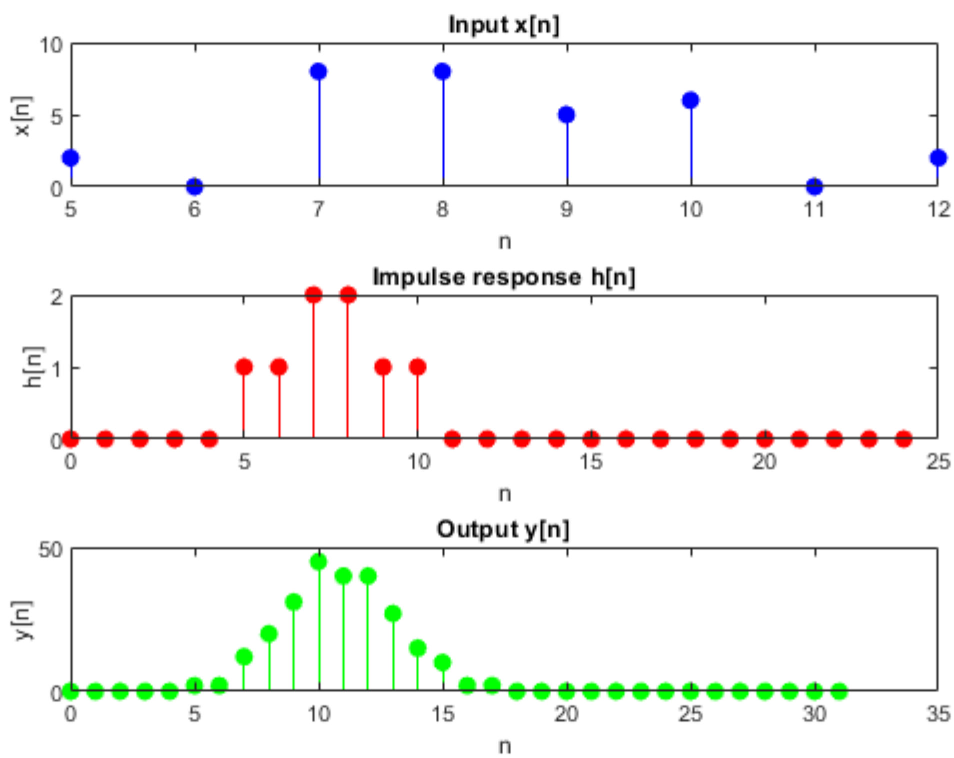
(1) **Only show non-zero value of y and corresponding ny**

ny	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
y	**2**	**2**	**12**	**20**	**31**	**45**	**40**	**40**	**27**	**15**	**10**	**2**

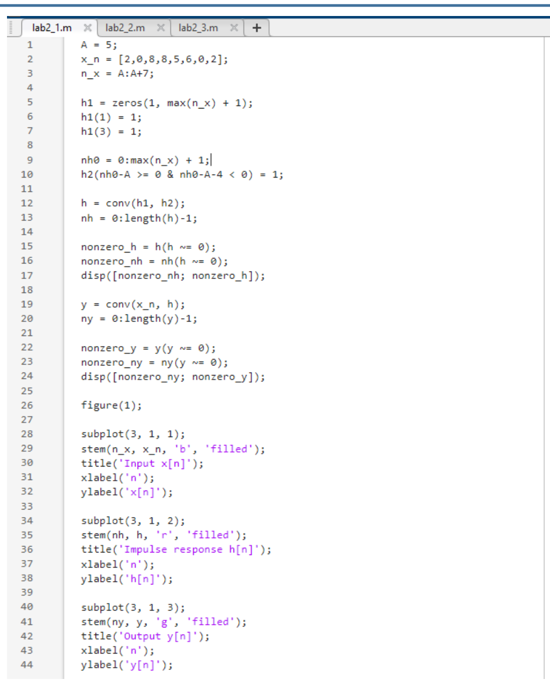
ny	**17**											
y	**2**											

d) Write **Matlab code** to plot the input in **BLUE**, the overall impulse response in **RED** and the output in **GREEN** using "**subplot**" and "**stem**". Show the plot in **figure (1)**.

(1) **figure (1)**



e) **(2)** **Screenshot of Matlab code for Part I**



lab2_1.m | lab2_2.m | lab2_3.m

```
A = 5;
x_n = [2,0,8,8,5,6,0,2];
n_x = A:A+7;

h1 = zeros(1, max(n_x) + 1);
h1(1) = 1;
h1(3) = 1;

nh0 = 0:max(n_x) + 1;
h2(nh0-A >= 0 & nh0-A-4 < 0) = 1;

h = conv(h1, h2);
nh = 0:length(h)-1;

nonzero_h = h(h ~= 0);
nonzero_nh = nh(h ~= 0);
disp([nonzero_nh; nonzero_h]);

y = conv(x_n, h);
ny = 0:length(y)-1;

nonzero_y = y(y ~= 0);
nonzero_ny = ny(y ~= 0);
disp([nonzero_ny; nonzero_y]);

figure(1);

subplot(3, 1, 1);
stem(n_x, x_n, 'b', 'filled');
title('Input x[n]');
xlabel('n');
ylabel('x[n]');

subplot(3, 1, 2);
stem(nh, h, 'r', 'filled');
title('Impulse response h[n]');
xlabel('n');
ylabel('h[n]');

subplot(3, 1, 3);
stem(ny, y, 'g', 'filled');
title('Output y[n]');
xlabel('n');
ylabel('y[n]');
```

<u>Part II</u> **(7)** Include all useful codes of Part II in (f)

Write **Matlab code** to read the audio file (**song2a.wav**).

a) What is the sampling frequency (*fs*) in Hz ?

(1) **The sampling frequency = 96000**

The impulse responses of two CT LTI systems are given below.

System 1 : $h1(t) = \delta(t)$

System 2 : $h2(t) = 0.9\ \delta(t - 0.3)$

These two CT LTI systems are converted into DT representation using the sampling frequency (*fs*).
b) Write **Matlab code** to calculate the required number of zeros for representing a time delay of 0.3 seconds.

(1) **Number of zeros = 0.3 * fs = 28800**

c) Write down the DT mathematical expressions of System 1 and System 2.

(1) $\boldsymbol{h1[n]} = \delta[n]$

$\boldsymbol{h2[n]} = \mathbf{0.9}\ \delta[n - 28800]$

These two DT LTI systems are connected **in parallel** and added together to form a new DT LTI system $h[n]$.
d) Write down the DT mathematical expression of the impulse response $h[n]$.

(1) $\boldsymbol{h[n]} = \delta[n] + \mathbf{0.9}\ \delta[n - 28800]$

Write **Matlab code** to define h1 and h2.
Write **Matlab code** to generate the impulse response h using h1 and h2.
Write **Matlab code** to generate the output y using "**conv**" if the audio file is applied to the new DT LTI system h.
Write **Matlab code** to hear the original audio file and the output y using "**sound**".

e) What does the new DT LTI system h do?

(1) It delays the original signal, and the sound is quieter, less hearable.

f) **(2)** **Screenshot of Matlab code for Part II**

lab2_1.m lab2_2.m

```
[x, fs] = audioread("song2a.wav");
x = x';

delay = 0.3*fs;
h = [1 zeros(1,delay) 0.9];

y = conv(x, fs);
sound(x, fs);
sound(y, fs);

```

<u>Part III</u> **(7)** **Include all useful codes of Part III in (d)**

Write **Matlab code** to read the image file (**image2a.jpg**) and show the image file in figure (2).
No need to place figure (2) on the summary sheet.

The image file is applied to a 2-D filter $h1[n] = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$.

Write **Matlab code** to define h1.
Write **Matlab code** to obtain the output y1 using "**convn**".
Write **Matlab code** to show y1 in **figure (3)** using "**uint8**" and "**imshow**".

(1) **figure (3)**

a) What does the 2-D filter $h1[n]$ do ?

(1) **Detects edges.**

The image file is applied to a 2-D filter $h2[n] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$.

Write **Matlab code** to define h2.
Write **Matlab code** to obtain the output y2 using "**convn**".
Write **Matlab code** to show y2 in figure (4) using "**uint8**" and "**imshow**". No need to place figure (4) below.

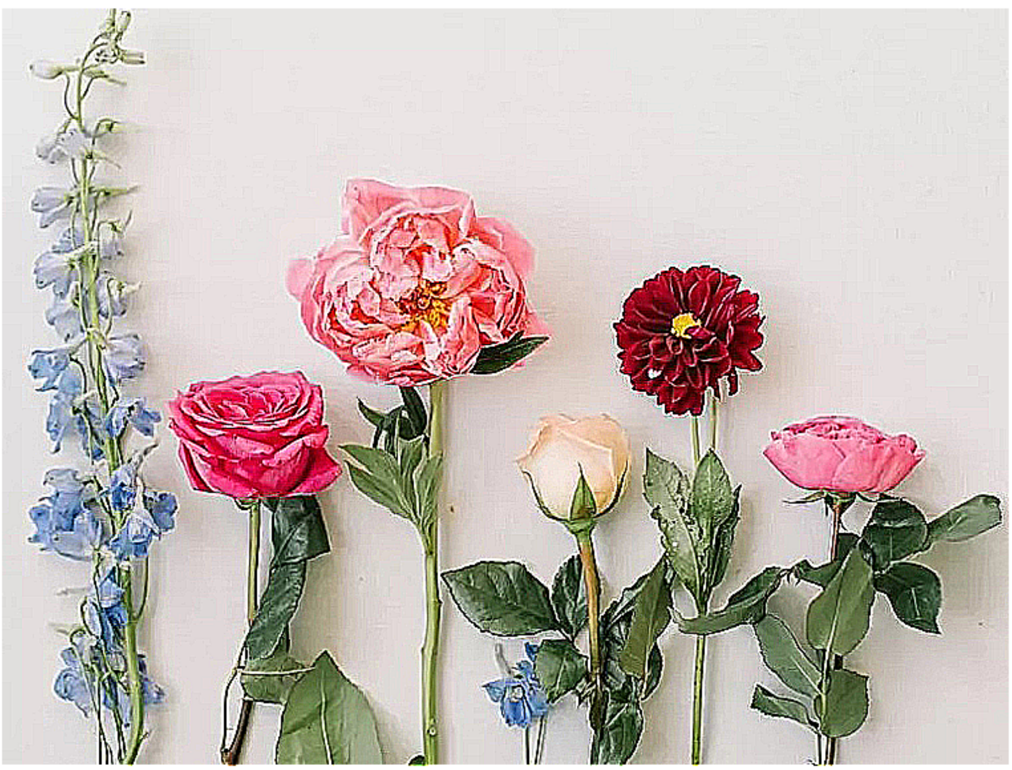
b) What does the 2-D filter $h2[n]$ do ?

(1) **Nothing.**

The 2-D filters $h1[n]$ and $h2[n]$ are connected in parallel and added together to form a new 2-D filter $h3[n]$.

Write **Matlab code** to generate h3 using the defined h1 and h2.
Write **Matlab code** to obtain the output y3 using "**convn**" if the image file is applied to h3.
Write **Matlab code** to show y3 in **figure (5)** using "**uint8**" and "**imshow**".

(1) **figure (5)**



c) What does the new 2-D filter $h3[n]$ do ?

(1) **Makes colors more expressive, sharper.**

d) **(2)** **Screenshot of Matlab code for Part III**

Editor - C:\Users\fkriuk\Desktop\lab2_3.m

lab2_1.m | lab2_2.m | lab2_3.m | +

```
i1 = imread('image2a.jpg');
figure(2);
imshow(i1);

h1 = [-1 -1 -1; -1 8 -1; -1 -1 -1];
y1 = convn(i1, h1, 'same');
figure(3);
imshow(uint8(y1));

h2 = [0 0 0; 0 1 0; 0 0 0];
y2 = convn(i1, h2, 'same');
figure(4);
imshow(uint8(y2));

h3 = h1 + h2;
y3 = convn(i1, h3, 'same');
figure(5);
imshow(uint8(y3));
```