CENG 384 - Signals and Systems for Computer Engineers
Spring 2023
Homework 2

Mehrabi, Sogol
e2547321@ceng.metu.edu.tr

Qaryouti, Hashem
e265281@ceng.metu.edu.tr

April 15, 2023

---

1. (a)

$$y(t) = \int_{-\infty}^{t} [x(\tau) - 5y(\tau)] d\tau$$

(b)

$$\begin{aligned} y(t) &= \int_{-\infty}^{t} [x(\tau) - 5y(\tau)] d\tau \\ &= \int_{-\infty}^{t} [(e^{-\tau} + e^{-3\tau})u(t) - 5y(\tau)] d\tau \\ &= \int_{0}^{t} [e^{-\tau} + e^{-3\tau} - 5y(\tau)] d\tau \end{aligned}$$

Differentiating both sides,

$$\frac{dy(t)}{dt} = e^{-t} + e^{-3t} - 5y(t)$$

$$\frac{dy(t)}{dt} + 5y(t) = e^{-t} + e^{-3t}$$

$$y(t) = y_h(t) + y_p(t)$$

$$\frac{dy_h(t)}{dt} + 5y_h(t) = 0$$

$$r + 5 = 0, r = -5$$

$$y_h(t) = c_1 e^{-5t}$$

$$y_p(t) = c_2 e^{-t} + c_3 e^{-3t}$$

$$-c_2 e^{-t} - 3c_3 e^{-3t} + 5c_2 e^{-t} + 5c_3 e^{-3t} = e^{-t} + e^{-3t}$$

$$c_2 = \frac{1}{4}, c_3 = \frac{1}{2}$$

$$y(t) = c_1 e^{-5t} + \frac{1}{4} e^{-t} + \frac{1}{2} e^{-3t}$$

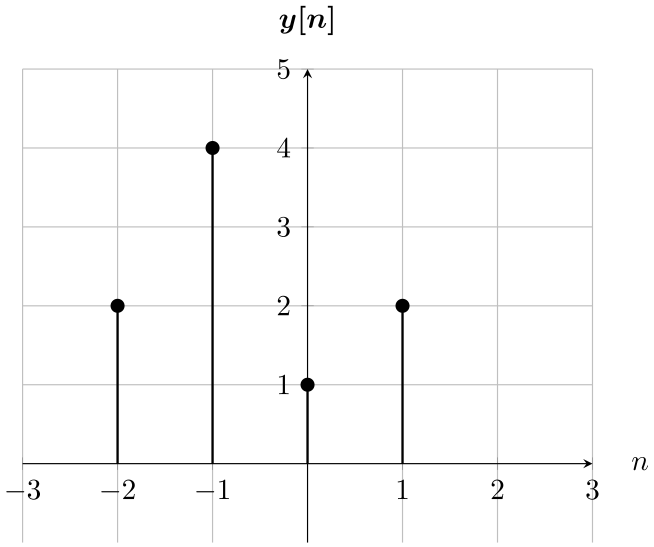
Since the system is initially at rest, $y(0) = 0$. So,

$$c_1 + \frac{1}{4} + \frac{1}{2} = 0, c_1 = -\frac{3}{4}$$

$$y(t) = -\frac{3}{4} e^{-5t} + \frac{1}{4} e^{-t} + \frac{1}{2} e^{-3t}$$

2. (a)

$$
\begin{aligned}
x[n] * h[n] &= \sum_{k=-\infty}^{\infty} h[k]x[n-k] \\
&= h[1]x[n-1] + h[-1]x[n+1] \\
&= x[n-1] + 2x[n+1] \\
&= (2\delta[n-1] + \delta[n]) + 2(2\delta[n+1] + \delta[n+2]) \\
&= 2\delta[n-1] + \delta[n] + 4\delta[n+1] + 2\delta[n+2]
\end{aligned}
$$



(b)

$$\frac{dx(t)}{dt} = \delta(t-1) + \delta(t+1) \tag{1}$$

$$
\begin{aligned}
y(t) = \frac{dx(t)}{dt} * h(t) &= \int_0^{\infty} e^{-\tau} sin(\tau)\delta(t-1-\tau)d\tau + \int_0^{\infty} e^{-\tau} sin(\tau)\delta(t+1-\tau)d\tau \\
&= e^{-(t-1)} sin(t-1)u(t-1) + e^{-(t+1)} sin(t+1)u(t+1)
\end{aligned}
$$

3. (a)

$$
\begin{aligned}
x(t) * h(t) &= \int_0^t e^{-\tau} e^{-2(t-\tau)} d\tau \\
&= e^{-2t} \int_0^t e^{\tau} d\tau \\
&= e^{-2t}(e^t - 1)
\end{aligned}
$$

(b)

$$
\begin{aligned}
x(t) * h(t) &= \int_{-\infty}^{\infty} h(\tau)x(t-\tau) \\
&= \int_0^{\infty} e^{3\tau}(u(t-\tau) - u(t-1-\tau)d\tau
\end{aligned}
$$

For $t \leq 2$, the convolution evaluates to 0.
For $0 < t \leq 1$,

$$y(t) = \int_0^t e^{3\tau} d\tau = \frac{1}{3}(e^{3t} - 1)$$

For $1 < t$,

$$y(t) = \int_{t-1}^t e^{3\tau} d\tau = \frac{1}{3}(e^{3t} - e^{3t-3})$$

4. (a)

$$y[n] - y[n-1] - y[n-2] = 0$$
$$r^2 - r - 1 = 0$$
$$r_1 = \frac{1+\sqrt{5}}{2}, r_2 = \frac{1-\sqrt{5}}{2}$$
$$y[n] = A(\frac{1+\sqrt{5}}{2})^n + B(\frac{1-\sqrt{5}}{2})^n$$
$$y[0] = 1, A + B = 1$$
$$y[1] = 1, A(\frac{1+\sqrt{5}}{2}) + B(\frac{1-\sqrt{5}}{2}) = 1$$
$$A + A\sqrt{5} + B - B\sqrt{5} = 2$$
$$(A - B)\sqrt{5} = 1$$
$$A = \frac{5+\sqrt{5}}{10}, B = \frac{5-\sqrt{5}}{10}$$
$$y[n] = (\frac{5+\sqrt{5}}{10})(\frac{1+\sqrt{5}}{2})^n + (\frac{5-\sqrt{5}}{10})(\frac{1-\sqrt{5}}{2})^n$$

(b)

$$r^3 - 6r^2 + 13r - 10 = 0$$
$$(r-2)(r^2 - 4r + 5) = 0$$
$$r_1 = 2, r_{2,3} = 2 \pm i$$
$$y(t) = Ae^{2t} + e^{2t}(Bcos(t) + Csin(t))$$
$$y(0) = 1, A + B = 1$$
$$y'(t) = 2Ae^{2t} + 2e^{2t}(Bcos(t) + Csin(t)) + e^{2t}(-Bsin(t) + Ccos(t))$$
$$y'(0) = \frac{3}{2}, 2A + 2(Bcos(0) + Csin(0)) + e^0(-Bsin(0) + Ccos(0)) = \frac{3}{2}$$
$$2A + 2B + C = \frac{3}{2}$$
$$y''(0) = 3$$
$$y''(t) = 4Ae^{2t} + e^{2t}(-Bcos(t) - Csin(t)) + 2e^{2t}(-Bsin(t) + Ccos(t)) + 2e^{2t}(-Bsin(t) + Ccos(t)) + 4e^{2t}(Bcos(t) + Csin(t))$$
$$4A + 3B + 4C = 3$$
$$A = 2, B = -1, C = -\frac{1}{2}$$
$$y(t) = 2e^{2t} + e^{2t}(-1cos(t) - \frac{1}{2}sin(t))$$

5. (a) We can assume that y_p has the form $Acos(5t) + Bsin(5t)$.

$$y_p(t) = Acos(5t) + Bsin(5t)$$
$$y_p'(t) = -5Asin(5t) + 5Bcos(5t)$$
$$y_p''(t) = -25Acos(5t) - 25Bsin(5t)$$
$$-25Acos(5t) - 25Bsin(5t) - 25Asin(5t) + 25Bcos(5t) + 6Acos(5t) + 6Bsin(5t) = cos(5t)$$
$$-19A + 25B = 1$$
$$-25A - 19B = 0$$
$$A = -\frac{19}{986}, B = \frac{25}{986}$$
$$y_p(t) = -\frac{19}{986}cos(5t) + \frac{25}{986}sin(5t)$$

(b)

$$y_h''(t) + 5y_h'(t) + 6y_h(t) = 0$$
$$r^2 + 5r + 6 = 0$$
$$(r+2)(r+3) = 0$$
$$r_1 = -2, r_2 = -3$$
$$y_h(t) = c_1 e^{-2t} + c_2 e^{-3t}$$

(c)

$$y(t) = y_h(t) + y_p(t) = c_1 e^{-2t} + c_2 e^{-3t} - \frac{19}{986} cos(5t) + \frac{25}{986} sin(5t)$$
$$y(0) = 0$$
$$c_1 + c_2 - \frac{19}{986} = 0$$
$$c_1 + c_2 = \frac{19}{986}$$
$$y'(t) = -2c_1 e^{-2t} - 3c_2 e^{-3t} + \frac{95}{986} sin(5t) + \frac{125}{986} cos(5t)$$
$$y'(0) = 0$$
$$-2c_1 - 3c_2 + \frac{125}{986} = 0$$
$$2c_1 + 3c_2 = \frac{125}{986}$$
$$c_1 = -\frac{-2}{29}, c_2 = \frac{3}{34}$$
$$y(t) = -\frac{-2}{29} e^{-2t} + \frac{3}{34} e^{-3t} - \frac{19}{986} cos(5t) + \frac{25}{986} sin(5t)$$

6. (a)

$$h_0[n] - \frac{1}{2} h_0[n-1] = \delta[n]$$
$$h_0[n] = \frac{1}{2} h_0[n-1] + \delta[n]$$

Since the system is initially at rest, $h_0[n] = 0$ for $n < 0$. $\delta[n] = 1$ at $n = 0$, $\delta[n] = 0$ for $n \neq 0$.

$$h_0[0] = \frac{1}{2} h_0[-1] + 1 = 1$$
$$h_0[1] = \frac{1}{2} h_0[0] + 0 = \frac{1}{2}$$
$$h_0[2] = \frac{1}{2} h_0[1] + 0 = \frac{1}{4}$$
$$\vdots$$
$$h_0[n] = \frac{1}{2^n} u[n]$$

(b)

$$\begin{aligned} h[n] = h_0[n] * h_0[n] &= \sum_{k=-\infty}^{\infty} \frac{1}{2^k} u[k] \frac{1}{2^{n-k}} u[n-k] \\ &= \sum_{k=0}^{n} \frac{1}{2^k} \frac{1}{2^{n-k}} \\ &= \sum_{k=0}^{n} \frac{1}{2^n} \\ &= \frac{n}{2^n} \end{aligned}$$

$$h[n] = \frac{n}{2^n} u[n]$$

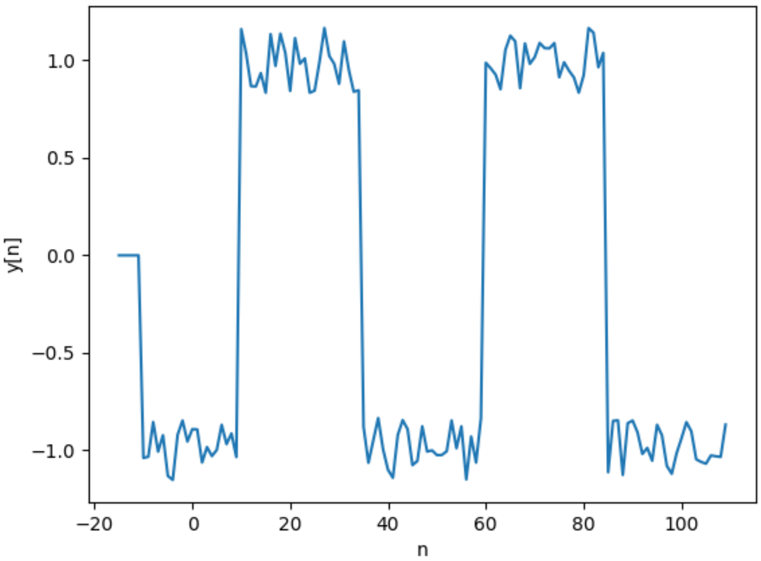
(c)

$$w[n] = y[n] - \frac{1}{2}y[n-1]$$
$$w[n-1] = y[n-1] - \frac{1}{2}y[n-2]$$
$$x[n] = w[n] - \frac{1}{2}w[n-1] = y[n] - \frac{1}{2}y[n-1] - \frac{1}{2}y[n-1] + \frac{1}{4}y[n-2]$$
$$= y[n] - y[n-1] + \frac{1}{4}y[n-2]$$
$$y[n] - y[n-1] + \frac{1}{4}y[n-2] = x[n]$$

7. (a)
```
import numpy as np
import matplotlib.pyplot as plt

def convolution(x,h,xi,hi):
    lx = len(x)
    lh = len(h)
    ly = lx + lh -1
    y = np.zeros(ly)
    for n in range(ly):
        for k in range(lx):
            if n - k >= 0 and n - k < lh:
                y[n] += x[k] * h[n - k]
    yi = xi+hi
    return y, yi

filename = '/Users/blake/Downloads/hw2/hw2_signal.csv'
with open(filename,'r') as f:
    data = f.read().split(',')
xi = int(data[0])
x = [float(val) for val in data[1:]]
h = np.zeros(6)
h[5] = 1
hi = 0
y, yi = convolution(x,h,xi,hi)
plt.figure()
plt.plot([yi + i for i in range(len(y))], y)
plt.xlabel("n")
plt.ylabel("y[n]")
plt.show()
```



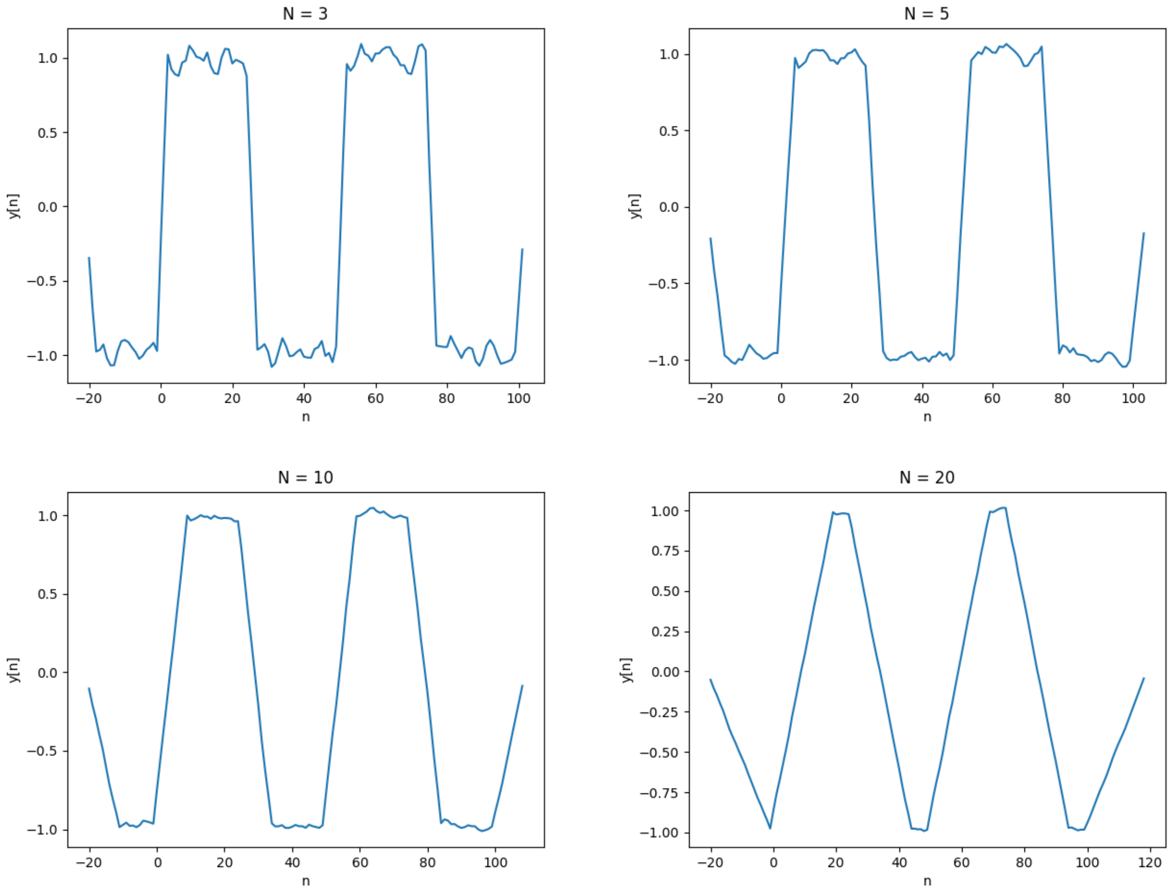
Convolution of the signal with $\delta[n-5]$ time shifts the signal 5 units to the right.

(b)
```
import numpy as np
import matplotlib.pyplot as plt

def convolution(x,h,xi,hi):
    lx = len(x)
    lh = len(h)
    ly = lx + lh -1
    y = np.zeros(ly)
    for n in range(ly):
        for k in range(lx):
            if n - k >= 0 and n - k < lh:
                y[n] += x[k] * h[n - k]
    yi = xi+hi
    return y, yi

filename = '/Users/blake/Downloads/hw2/hw2_signal.csv'
with open(filename,'r') as f:
    data = f.read().split(',')
xi = int(data[0])
x = [float(val) for val in data[1:]]
nvals = [3,5,10,20]
for n in nvals:
    h = np.ones(n) / n
    hi = 0
    y, yi = convolution(x, h, xi, hi)

    plt.figure()
    plt.plot([yi + i for i in range(len(y))], y)
    plt.title(f"N = {n}")
    plt.xlabel("n")
    plt.ylabel("y[n]")
plt.show()
```



This convolution applies some noise reduction to the signal and the higher N gets the more this noise reduction becomes, i.e. we see a smoother signal but with fewer details, and we only see the general trend.