



Faculty of Engineering, Architecture and Science

Department of Electrical and Computer Engineering

Course Number	ELE532
Course Title	Signals and Systems I
Semester/Year	Fall 2021
Instructor	Dr. Javad Alirezaie

ASSIGNMENT No.

2

Assignment Title	System Properties and Convolution
------------------	-----------------------------------

Submission Date	October 29, 2021
Due Date	October 31, 2021

Student Name	Reza Aablu
Student ID	500966944
Signature*	R.A.

Student Name	Rendel Abrasia
Student ID	500942743
Signature*	R.A.

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a “0” on the work, an “F” in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: www.ryerson.ca/senate/current/pol60.pdf.

Problem A:

A.1

```
1 %Rendel Abrasia, Reza Aabluu
2 %500942743, 500966944
3 %Section 4
4
5 % Problem A.1
6
7 % Setting component values.
8 R = [1e4, 1e4, 1e4];
9 C = [1e-6, 1e-6];
10 % Determining the coefficients for characteristic equation.
11 A1 = [1, (1/R(1)+1/R(2)+1/R(3))/C(2), 1/(R(1)*R(2)*C(1)*C(2))];
12 % Determining characteristic equation's roots.
13 lambda = roots(A1);
14
15 % The poly command takes the matrix of the roots and return the original
16 % polynomial equation.
17 poly(lambda);
```

Command Window

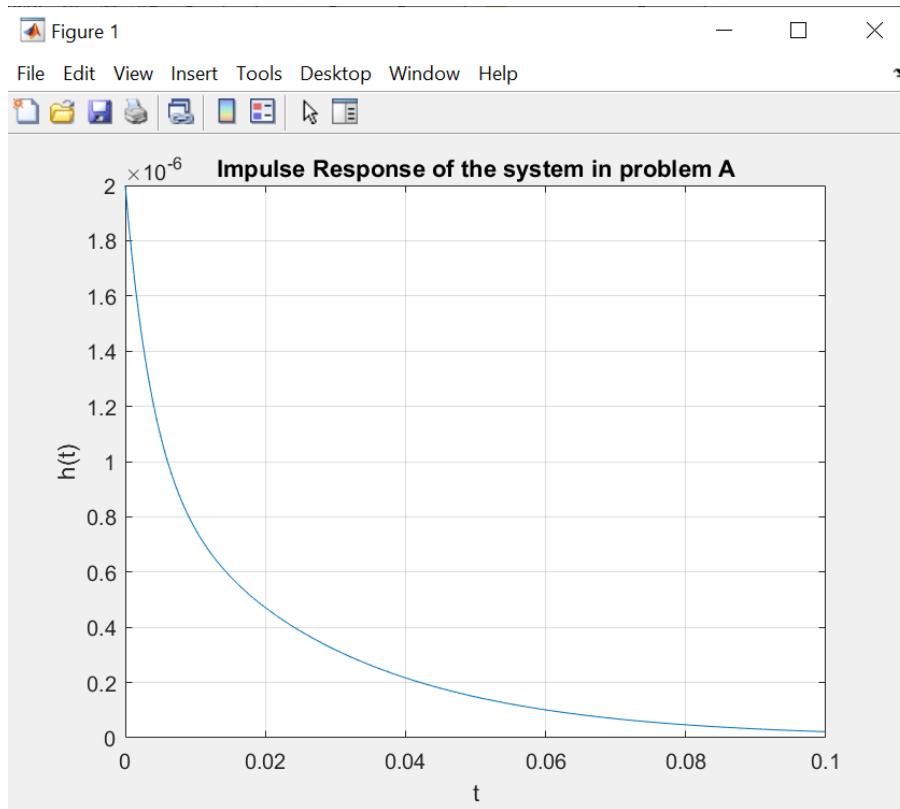
```
>> Lab2PartA
>> lambda

lambda =

-261.8034
-38.1966
```

A.2

```
19 % Problem A.2
20 |
21 % Set up time interval and step size, along with u(t) step function.
22 - t = (0:0.0005:0.1);
23 - u = @(t) 1.0*(t>=0);
24
25 - h = @(t) (C(1).* exp(lambda(1).* t) + C(2).* exp(lambda(2).*t)).*(u(t));
26 % Case 1 of the 2nd order DE solution applies for our h(t) function.
27
28 - plot(t,h(t)) % Plots h(t) function.
29
30 % Setting up x and y-labels and title for graph.
31 - xlabel('t');
32 - ylabel('h(t)');
33 - title('Impulse Response of the system in problem A');
34 - grid;
```



A.3

```
1 %Rendel Abrasia, Reza Aabue
2 %500942743, 500966944
3 %Section 4
4
5 % Problem A.3
6
7 function [lambda] = ProblemA3(R,C)
8 % CH2MP2.m : Chapter 2, MATLAB Program 2
9 % Function M-file finds characteristic roots of op-amp circuit.
10 % INPUTS: R = length-3 vector of resistances
11 % C = length-2 vector of capacitances
12 % OUTPUTS: lambda = characteristic roots
13 % Determine coefficients for characteristic equation:
14 - A = [1, (1/R(1)+1/R(2)+1/R(3))/C(2), 1/(R(1)*R(2)*C(1)*C(2))];
15 % Determine characteristic roots:
16 lambda = roots(A);
17
18 % To test this function, just use this command in the command window with your desired "R" and
19 % "C" values to get the eigenvalues:
20 % ie. lambda = ProblemA3([1e4, 1e4, 1e4],[1e-9, 1e-6])
```

Command Window

```
>> Lab2PartA
>> lambda = ProblemA3([1e4, 1e4, 1e4],[1e-9, 1e-6])
lambda =
1.0e+03 *
-0.1500 + 3.1587i
-0.1500 - 3.1587i
fx >> |
```

Problem B

B.1

```
1 %Rendel Abrasia, Reza Aabue
2 %500942743, 500966944
3 %Section 4
4
5 % Problem B.1
6 figure(1) % Create figure window and make visible on screen
7 u = @(t) 1.0*(t>=0); % u(t) step function.
8 x = @(t) 1.5*sin(pi*t).*(u(t)-u(t-1)); % Assigned x(t) function.
9 h = @(t) 1.5*(u(t)-u(t-1.5))-u(t-2)+u(t-2.5); % h(t).
10 dtau = 0.005; tau = -1:dtau:4; % Time intervals and step size for tau.
11 ti = 0; tvec = -.25:.1:2.25; % Time intervals and step size for tvec.
12 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
13 for t = tvec,
14 ti = ti+1; % Time index
15 xh = x(t-tau).*h(tau); lkh = length(xh);
16 y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
17 subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
18 axis([tau(1) tau(end) -2.0 2.5]);
19 patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
20 [zeros(1,lkh-1);xh(1:end-1);xh(2:end);zeros(1,lkh-1)],...
21 [.8 .8 .8], 'edgecolor','none');
22 xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
23 c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
24 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
25 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
26 axis([tau(1) tau(end) -1.0 2.0]); grid;
27 drawnow;
28
29
```

B.2

```
30 % Problem B.2
31 % x(t) = u(t) - u(t-2)
32 % h(t) = (t+1)(u(t+1)-u(t))
33 % y(t) = x(t)*h(t) (Convolution)
34
35 - figure(2) % Create figure window and make visible on screen
36 - u = @(t) 1.0*(t>=0); % u(t) step function.
37 - x = @(t) u(t)-u(t-2); % Assigned x(t) function (Figure 2.4-28 (a)).
38 - h = @(t) (t+1).* (u(t+1)-u(t)); % h(t) = (t+1)(u(t+1)-u(t)).
39 - dtau = 0.005; tau = -1:dtau:4; % Time intervals and step size for tau.
40 - tvec = -1.0:.1:2.25; % Time intervals and step size for tvec.
41 - y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
42 - for t = tvec,
43 - ti = ti+1; % Time index
44 - xh = x(t-tau).*h(tau); lxh = length(xh);
45 - y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
46 - subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
47 - axis([tau(1) tau(end) -2.0 2.5]);
48 - patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
49 - [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
50 - [.8 .8 .8], 'edgecolor', 'none');
51 - xlabel('tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
52 - c = get(gca, 'children'); set(gca, 'children', [c(2);c(3);c(4);c(1)]);
53 - subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
54 - xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
55 - axis([tau(1) tau(end) -1.0 2.0]); grid;
56 - drawnow;
57 - end
58
```

B.3

```
59 % Problem B.3 - part 1 (Assume A=1, B=2)
60 % x1(t) = x(t) = u(t-4)-u(t-6)
61 % x2(t) = h(t) = (2) (u(t+5)-u(t+4))
62 % y(t) = x(t)*h(t)
63
64 figure(3) % Create figure window and make visible on screen
65 A = 1.0; B = 2.0; % Assumptions for A and B values.
66 u = @(t) 1.0*(t>=0); % u(t) step function.
67 x = @(t) A*(u(t-4)-u(t-6)); % Assigned x(t) function (Figure 2.4-28 (a)).
68 h = @(t) B*(u(t+5)-u(t+4)); % h(t) = (t+1)(u(t+1)-u(t)).
69 dtau = 0.005; tau = -6:dtau:2.5; % Time intervals and step size for tau.
70 ti = 0; tvec = -5.0::1:5; % Time intervals and step size for tvec.
71 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
72 for t = tvec,
73 ti = ti+1; % Time index
74 xh = x(t-tau).*h(tau); lxh = length(xh);
75 y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
76 subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
77 axis([tau(1) tau(end) -2.0 2.5]);


---


78 patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
79 [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
80 [.8 .8 .8], 'edgecolor','none');
81 xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
82 c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
83 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
84 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
85 axis([tau(1) tau(end) -1.0 2.0]); grid;
86 drawnow;
87 end
88
```

```

89 % Problem B.3 - part 2 (Assume A=1, B=2)
90 % x1(t) = x(t) = u(t-3)-u(t-5)
91 % x2(t) = h(t) = (2)(u(t+5)-u(t+3))
92 % y(t) = x(t)*h(t)
93
94 figure(4) % Create figure window and make visible on screen
95 A = 1.0; B = 2.0; % Assumptions for A and B values.
96 u = @(t) 1.0*(t>=0); % u(t) step function.
97 x = @(t) A*(u(t-3)-u(t-5)); % Assigned x(t) function (Figure 2.4-28 (a)).
98 h = @(t) B*(u(t+5)-u(t+3)); % h(t) = (t+1)(u(t+1)-u(t)).
99 dtau = 0.005; tau = -6:dtau:2.5; % Time intervals and step size for tau.
100 ti = 0; tvec = -5.0::1:5; % Time intervals and step size for tvec.
101 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
102 for t = tvec,
103 ti = ti+1; % Time index
104 xh = x(t-tau).*h(tau); lkh = length(xh);
105 y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
106 subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
107 axis([tau(1) tau(end) -2.0 2.5]);

```

```

108 patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
109 [zeros(1,lkh-1);xh(1:end-1);xh(2:end);zeros(1,lkh-1)],...
110 [.8 .8 .8], 'edgecolor','none');
111 xlabel('tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
112 c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
113 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
114 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
115 axis([tau(1) tau(end) -1.0 4.0]); grid;
116 drawnow;
117 end
118

```

```

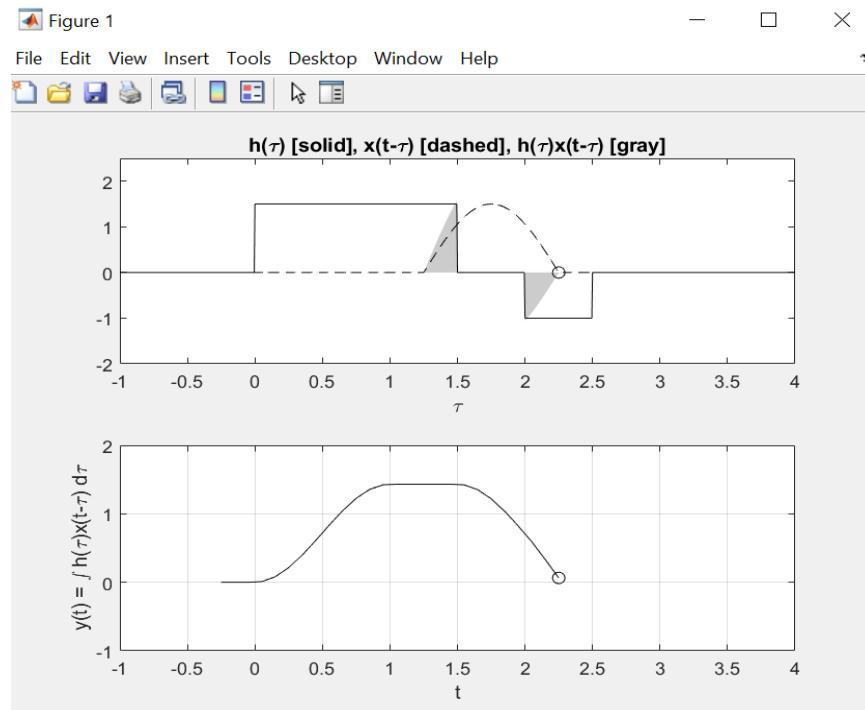
119 % Problem B.3 - part 3
120 % x1(t) = x(t) = e^t(u(t+2)-u(t))
121 % x2(t) = h(t) = e^{-2t}(u(t)-u(t-1))
122 % y(t) = x(t)*h(t)
123
124 figure(5) % Create figure window and make visible on screen
125 u = @(t) 1.0*(t>=0); % u(t) step function.
126 x = @(t) exp(t).*(u(t+2)-u(t)); % Assigned x(t) function (Figure 2.4-28 (a)).
127 h = @(t) exp(-2*t).*(u(t)-u(t-1)); % h(t) = (t+1)(u(t+1)-u(t)).
128 dtau = 0.005; tau = -6:dtau:2.5; % Time intervals and step size for tau.
129 ti = 0; tvec = -5.0::1:5; % Time intervals and step size for tvec.
130 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
131 for t = tvec,
132 ti = ti+1; % Time index
133 xh = x(t-tau).*h(tau); lkh = length(xh);
134 y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
135 subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
136 axis([tau(1) tau(end) -2.0 2.5]);
137 patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...

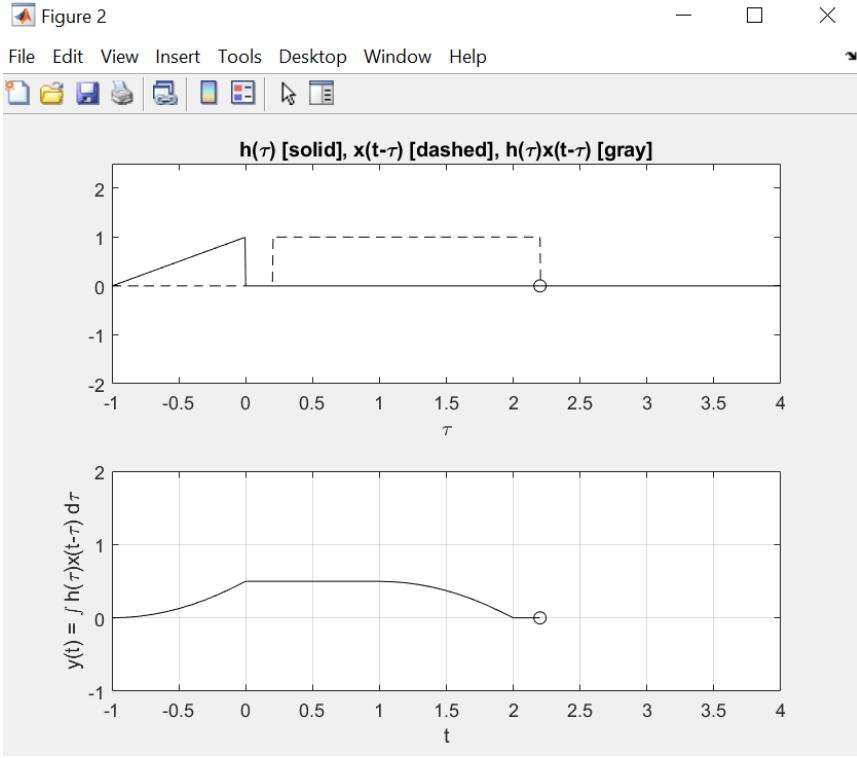
```

```

138 [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
139 [.8 .8 .8], 'edgecolor', 'none');
140 xlabel('tau'); title('h(tau) [solid], x(t-tau) [dashed], h(tau)x(t-tau) [gray]');
141 c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
142 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
143 xlabel('t'); ylabel('y(t) = \int h(tau)x(t-tau) d\tau');
144 axis([tau(1) tau(end) -1.0 2.0]); grid;
145 drawnow;
146 end

```

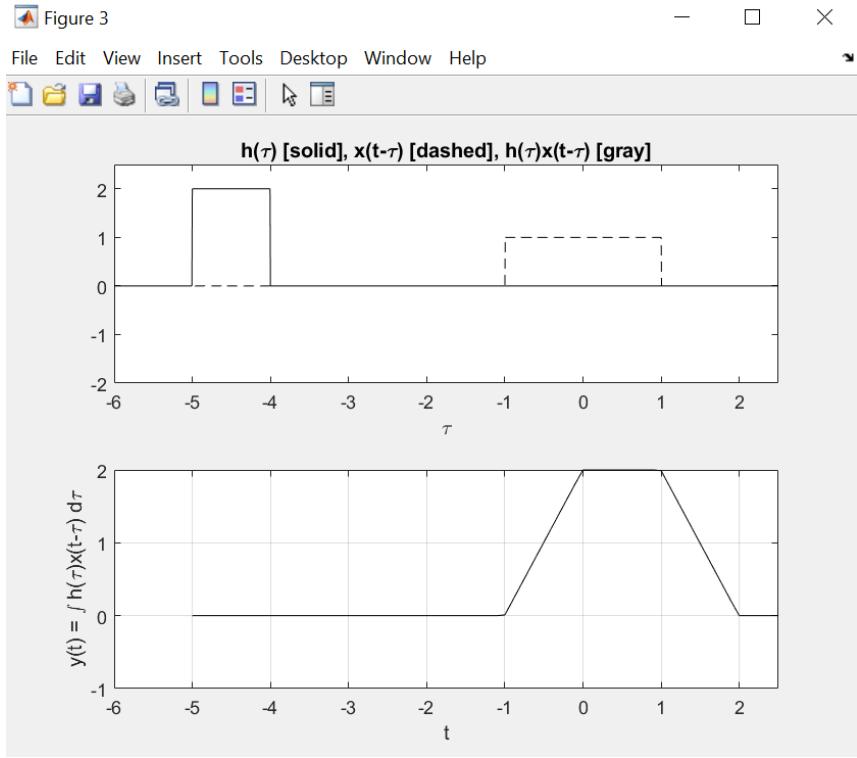
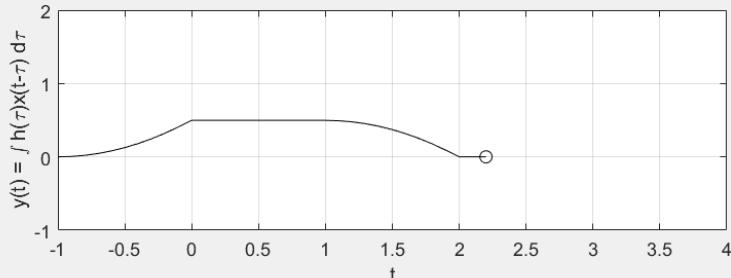
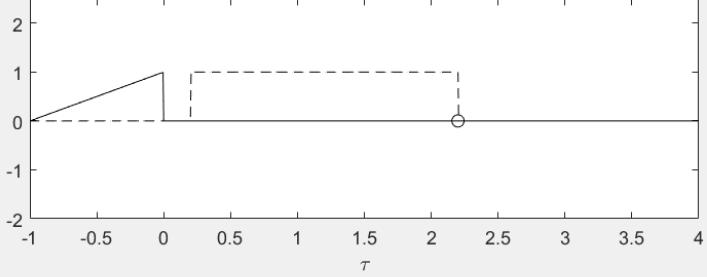




File Edit View Insert Tools Desktop Window Help



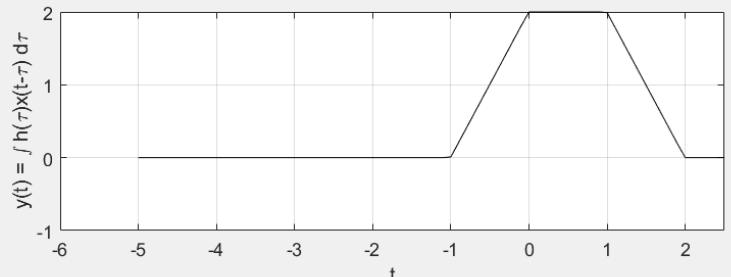
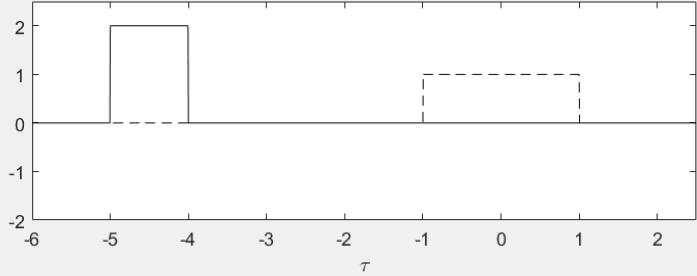
$h(\tau)$ [solid], $x(t-\tau)$ [dashed], $h(\tau)x(t-\tau)$ [gray]

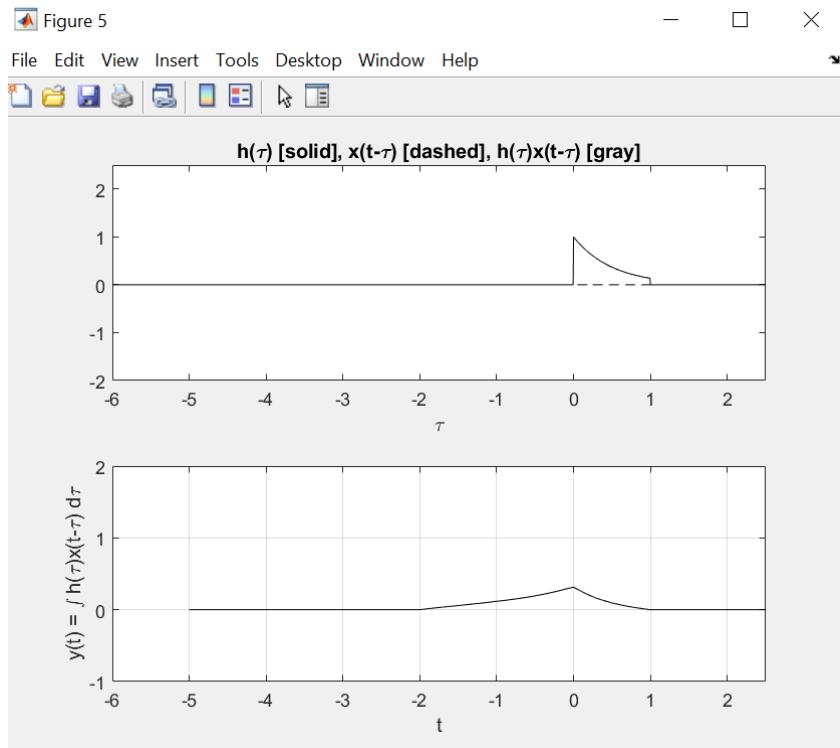
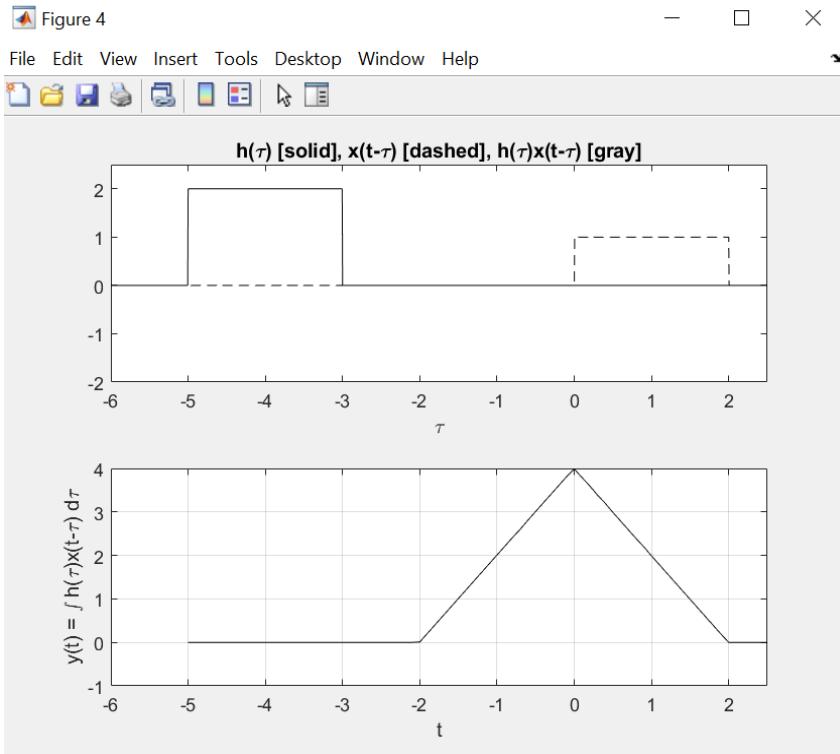


File Edit View Insert Tools Desktop Window Help



$h(\tau)$ [solid], $x(t-\tau)$ [dashed], $h(\tau)x(t-\tau)$ [gray]



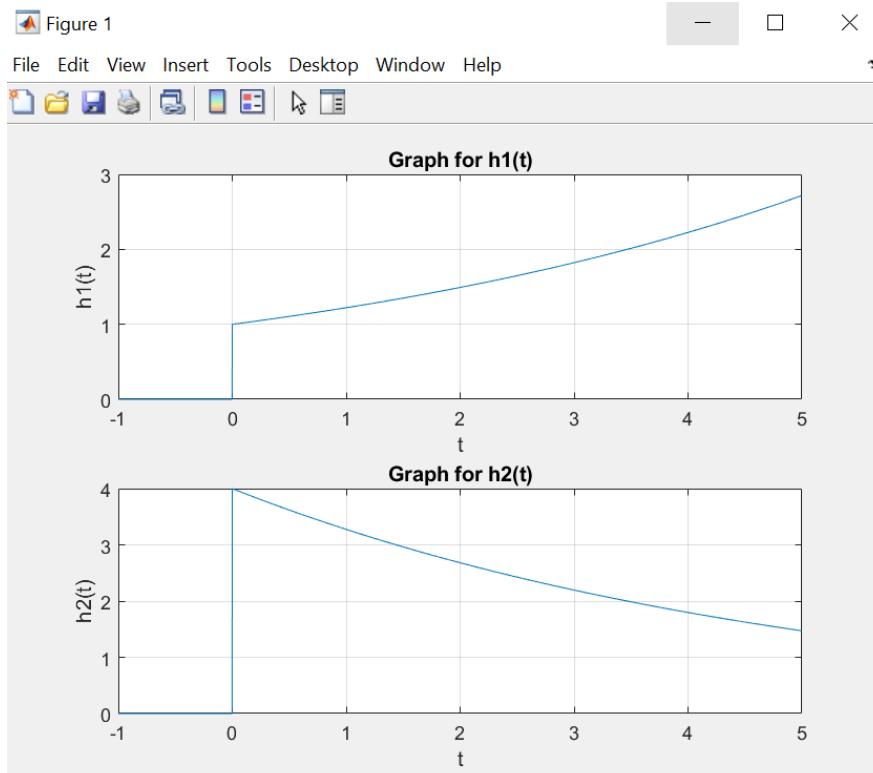


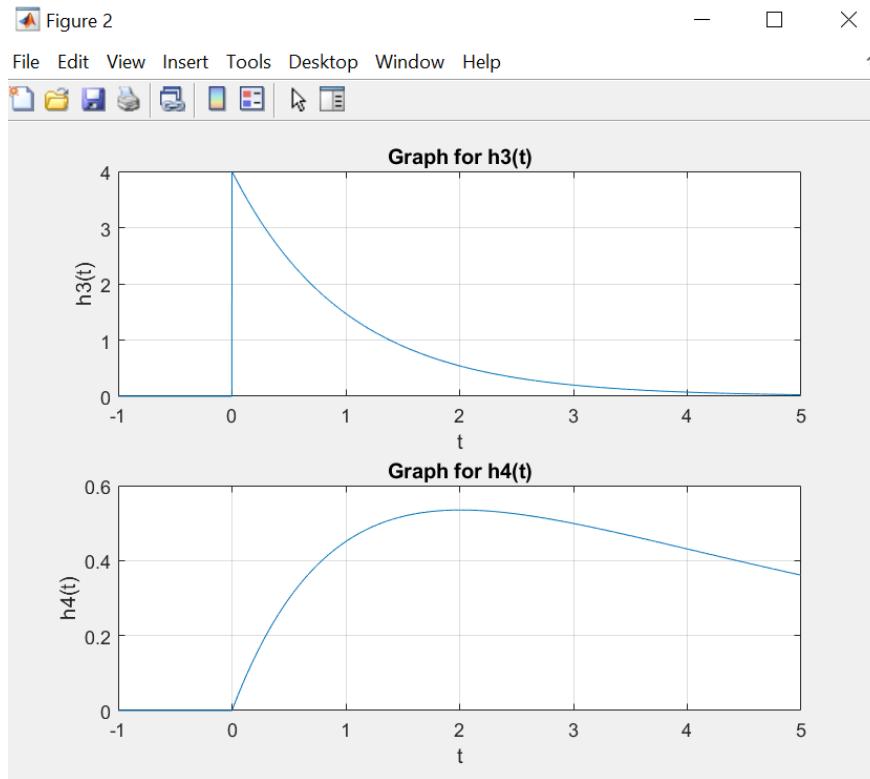
Problem C

C.1

```
1 %Rendel Abrasia, Reza Aabluu
2 %500942743, 500966944
3 %Section 4
4
5 % Problem C.1
6
7 % Set up time interval and step size, along with u(t) step function.
8 t = (-1:0.001:5);
9 u = @(t) 1.0* (t>=0);
10
11 h1 = @(t) (exp((1/5) .* t)).*(u(t)); % Equation for h1(t) function.
12
13 h2 = @(t) 4 .* (exp((-1/5) .* t)).*(u(t)); % Equation for h2(t) function.
14
15 h3 = @(t) 4 .* (exp((-1) .* t)).*(u(t)); % Equation for h3(t) function.
16
17 h4 = @(t) (exp((-1/5) .* t) - exp((-1).*t)).*(u(t)); % Equation for h4(t) function.
18
19 figure (1);
20
21 subplot(211);
22 plot(t,h1(t));
23 xlabel('t');
24 ylabel('h1(t)');
25 title('Graph for h1(t)');
26 grid;
27
28 subplot(212);
29 plot(t,h2(t));
30 xlabel('t');
31 ylabel('h2(t)');
32 title('Graph for h2(t)');
33 grid;
```

```
35 figure (2);
36
37 subplot(211);
38 plot(t,h3(t));
39 xlabel('t');
40 ylabel('h3(t)');
41 title('Graph for h3(t)');
42 grid;
43
44 subplot(212);
45 plot(t,h4(t));
46 xlabel('t');
47 ylabel('h4(t)');
48 title('Graph for h4(t)');
49 grid;
```





C.2

Determine the characteristics values (eigenvalues) of systems S1-S4

For $h_1(t)$, the eigenvalue is $1/5$.

For $h_2(t)$, the eigenvalue is $-1/5$.

For $h_3(t)$, the eigenvalue is -1 .

For $h_4(t)$, the eigenvalues are $-1/5$ and -1 .

C.3

```
57 % Problem C.3 - Part 1
58 % h1(t) = e^(t/5)u(t)
59
60 figure(1) % Create figure window and make visible on screen
61 u = @(t) 1.0*(t>=0); % u(t) step function.
62 x = @(t) sin(5*t).*(u(t)-u(t-3)); % Assigned x(t) function (Figure 2.4-28 (a)).
63 h = @(t) exp(t/5).*(u(t)-u(t-20)); % h(t) = (t+1)(u(t+1)-u(t)).
64 dtau = 0.005; tau = 0:dtau:20; % Time intervals and step size for tau.
65 ti = 0; tvec = 0:.1:20; % Time intervals and step size for tvec.
66 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
67 for t = tvec,
68     ti = ti+1; % Time index
69     xh = x(t-tau).*h(tau); lxh = length(xh);
70     y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
71     subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
72     axis([tau(1) tau(end) -2.0 2.5]);
73     patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
74 [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
75 [.8 .8 .8], 'edgecolor', 'none');
76 xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
77 c = get(gca, 'children'); set(gca, 'children',[c(2);c(3);c(4);c(1)]);
78 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
79 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
80 axis([tau(1) tau(end) -1.0 2.0]); grid;
81 drawnow;
82 end

84 % Problem C.3 - Part 2
85 % h2(t) = 4e^(-t/5)u(t)
86
87 figure(2) % Create figure window and make visible on screen
88 u = @(t) 1.0*(t>=0); % u(t) step function.
89 x = @(t) sin(5*t).*(u(t)-u(t-3)); % Assigned x(t) function (Figure 2.4-28 (a)).
90 h = @(t) 4*exp(-t/5).*(u(t)-u(t-20)); % h(t) = (t+1)(u(t+1)-u(t)).
91 dtau = 0.005; tau = 0:dtau:20; % Time intervals and step size for tau.
92 ti = 0; tvec = 0:.1:20; % Time intervals and step size for tvec.
93 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
94 for t = tvec,
95     ti = ti+1; % Time index
96     xh = x(t-tau).*h(tau); lxh = length(xh);
97     y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
98     subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
99     axis([tau(1) tau(end) -2.0 2.5]);
100    patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
101 [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
102 [.8 .8 .8], 'edgecolor', 'none');
103 xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
104 c = get(gca, 'children'); set(gca, 'children',[c(2);c(3);c(4);c(1)]);
105 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
106 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
107 axis([tau(1) tau(end) -1.0 2.0]); grid;
108 drawnow;
109 end
```

```

111 % Problem C.3 - Part 3
112 % h3(t) = 4e^(-t)u(t)
113
114 figure(3) % Create figure window and make visible on screen
115 u = @(t) 1.0*(t>=0); % u(t) step function.
116 x = @(t) sin(5*t).*(u(t)-u(t-3)); % Assigned x(t) function (Figure 2.4-28 (a)).
117 h = @(t) 4*exp(-t).* (u(t)-u(t-20)); % h(t) = (t+1)(u(t+1)-u(t)).
118 dtau = 0.005; tau = 0:dtau:20; % Time intervals and step size for tau.
119 ti = 0; tvec = 0:.1:20; % Time intervals and step size for tvec.
120 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
121 for t = tvec,
122 ti = ti+1; % Time index
123 xh = x(t-tau).*h(tau); lkh = length(xh);
124 y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
125 subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
126 axis([tau(1) tau(end) -2.0 2.5]);
127 patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
128 [zeros(1,lkh-1);xh(1:end-1);xh(2:end);zeros(1,lkh-1)],...
129 [.8 .8 .8], 'edgecolor', 'none');
130 xlabel('t\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
131 c = get(gca, 'children'); set(gca, 'children',[c(2);c(3);c(4);c(1)]);
132 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
133 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
134 axis([tau(1) tau(end) -1.0 2.0]); grid;
135 drawnow;
136 end

```

```

138 % Problem C.3 - Part 4
139 % h3(t) = 4(e^(-t/5) - e^(-t))u(t)
140
141 figure(4) % Create figure window and make visible on screen
142 u = @(t) 1.0*(t>=0); % u(t) step function.
143 x = @(t) sin(5*t).*(u(t)-u(t-3)); % Assigned x(t) function (Figure 2.4-28 (a)).
144 h = @(t) 4*(exp(-t/5) - exp(-t)).*(u(t)-u(t-20)); % h(t) = (t+1)(u(t+1)-u(t)).
145 dtau = 0.005; tau = 0:dtau:20; % Time intervals and step size for tau.
146 ti = 0; tvec = 0:.1:20; % Time intervals and step size for tvec.
147 y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
148 for t = tvec,
149 ti = ti+1; % Time index
150 xh = x(t-tau).*h(tau); lkh = length(xh);
151 y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
152 subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
153 axis([tau(1) tau(end) -2.0 2.5]);
154 patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
155 [zeros(1,lkh-1);xh(1:end-1);xh(2:end);zeros(1,lkh-1)],...
156 [.8 .8 .8], 'edgecolor', 'none');
157 xlabel('t\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
158 c = get(gca, 'children'); set(gca, 'children',[c(2);c(3);c(4);c(1)]);
159 subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
160 xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
161 axis([tau(1) tau(end) -1.0 2.0]); grid;
162 drawnow;
163 end

```

Figure 3

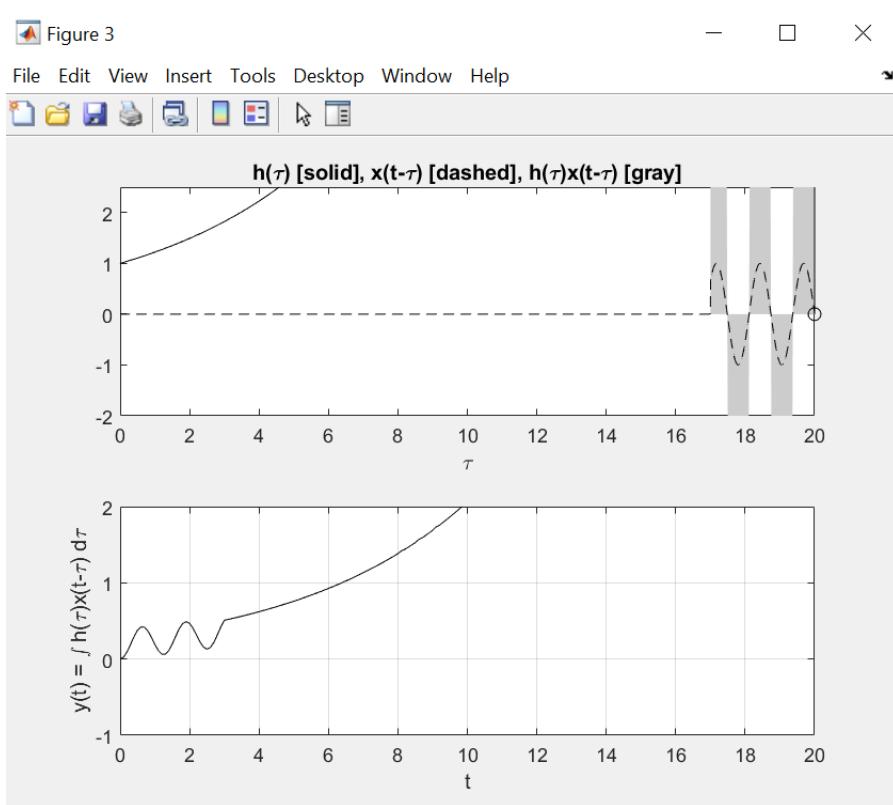
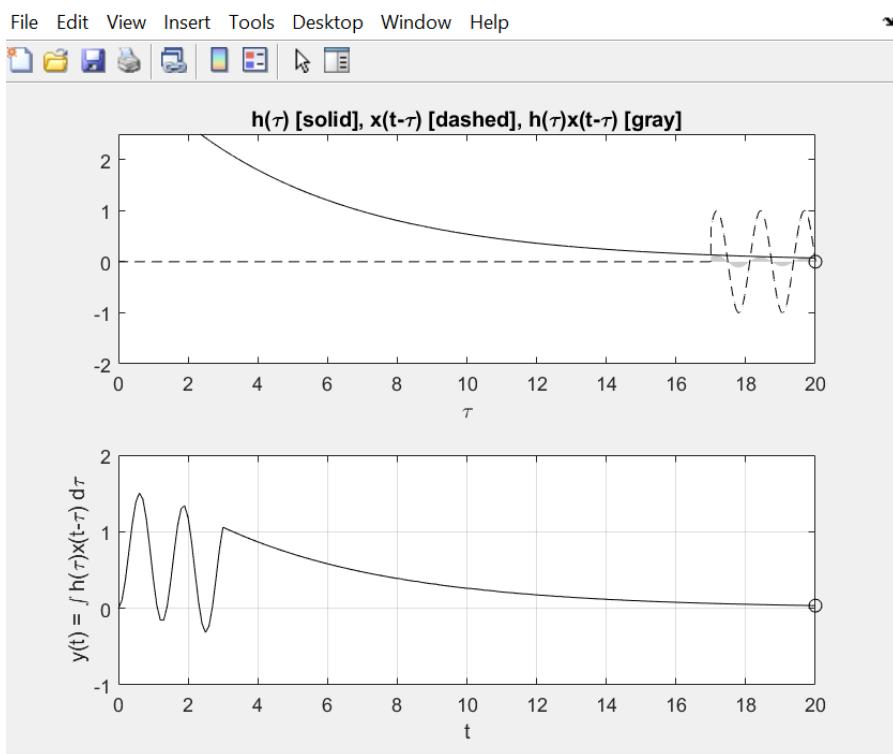
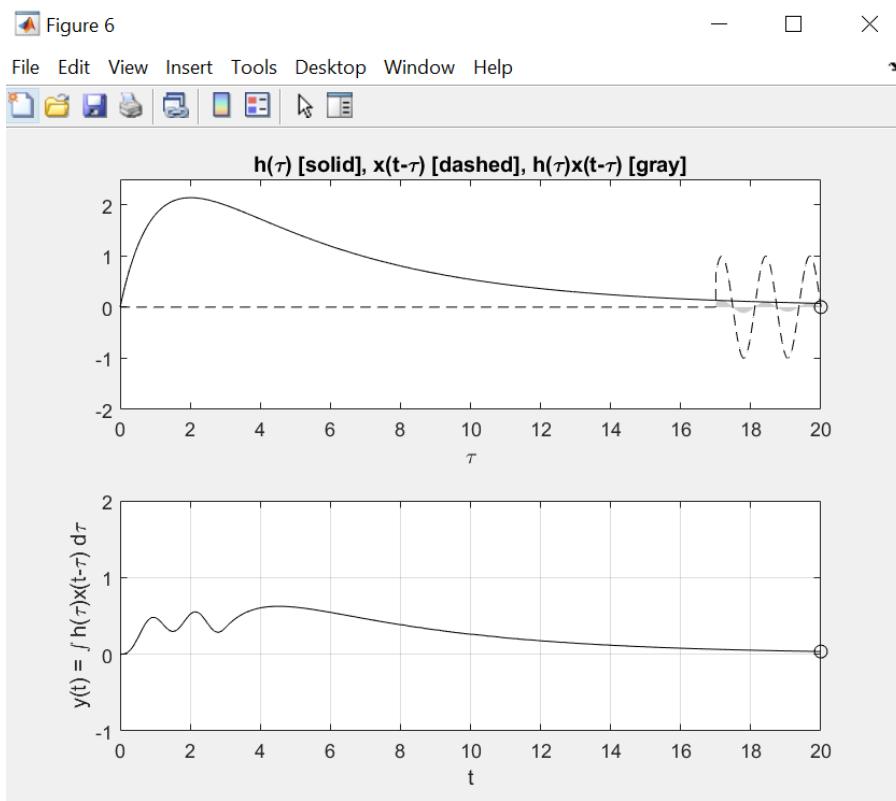
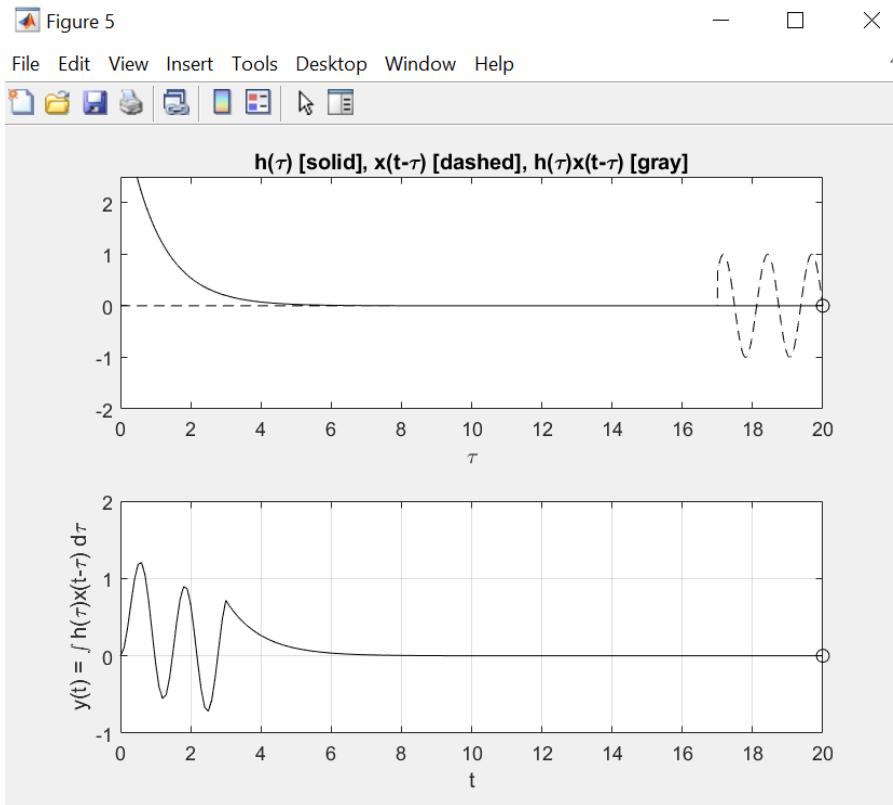


Figure 4



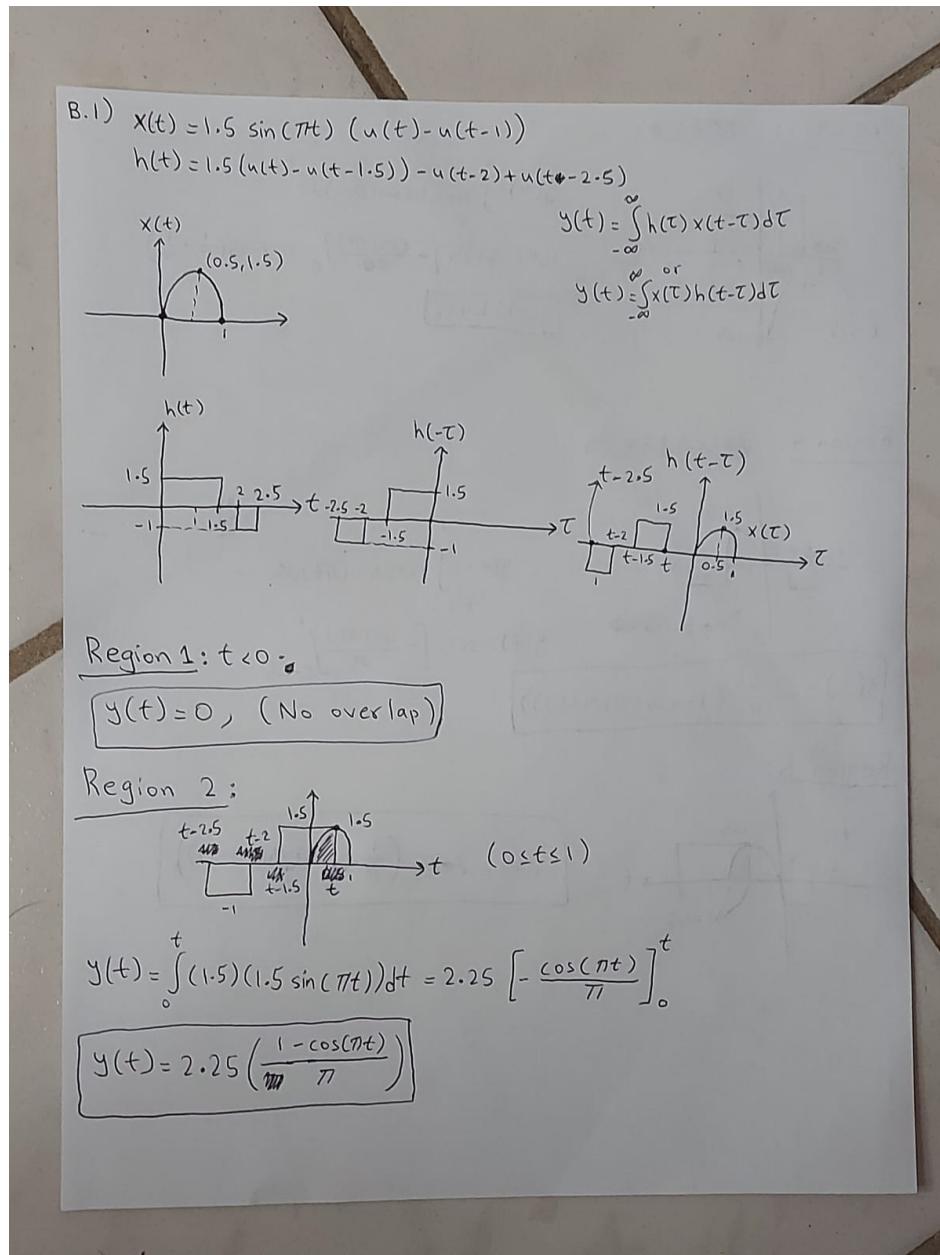


Is there any relationship between the output of systems S2, S3, S4? Explain.

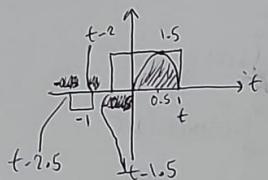
Due to the linearity of the three LTI systems (S2, S3 and S4), the convolutions of S2 and S3 are the exact same shape. As for S4, it is defined as the difference between S2 and S3, and its convolution is fairly similar to S2 and S3. In essence, all three LTI systems among S2, S3 and S4 look fairly similar to each other, with S2 and S3(t) having the same convolution graph, and S4(t) having a similar shape to S2 and S3(t).

Problem D

D.1



Region 3:

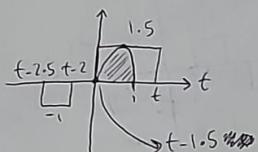


$$y(t) = \int_0^t (1.5)(1.5 \sin(\pi t)) dt$$

$$y(t) = 2.25 \left[-\frac{\cos(\pi t)}{\pi} \right]_0^1 = 2.25 \left(\frac{1}{\pi} + \frac{1}{\pi} \right)$$

$$y(t) = 1.432$$

Region 4:

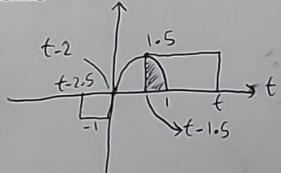


$$y(t) = \int_{-1.5}^t 2.25 \sin(\pi t) dt$$

$$y(t) = 2.25 \left[-\frac{\cos(\pi t)}{\pi} \right]_{-1.5}^1$$

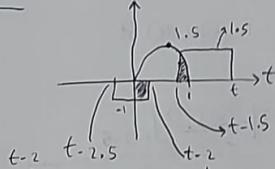
$$y(t) = \frac{2.25}{\pi} (1 + \cos(\pi(t-1.5)))$$

Region 5:



$$y(t) = 2.25 \left(\frac{1}{\pi} + \cos(\pi(t-1.5)) \right)$$

Region 6:

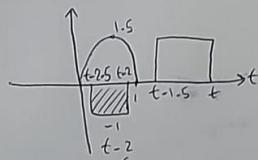


$$y(t) = \int_0^{-1.5} -1.5 \sin(\pi t) dt + \int_{-1.5}^t (1.5)(1.5 \sin(\pi t)) dt$$

$$y(t) = -1.5 \left[-\frac{\cos(\pi t)}{\pi} \right]_0^{-1.5} + 2.25 \left[-\frac{\cos(\pi t)}{\pi} \right]_{t-1.5}^t$$

$$y(t) = -1.5 \left(\frac{-\cos(\pi(t-2)) + 1}{\pi} \right) + 2.25 \left(\frac{1 + \cos(\pi(t-1.5))}{\pi} \right)$$

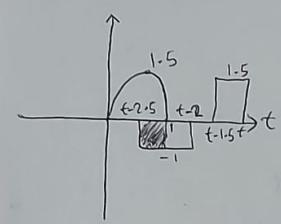
Region 7:



$$y(t) = \int_{t-2.5}^{-1.5} -1.5 \sin(\pi t) dt$$

$$y(t) = -1.5 \left(\frac{-\cos(\pi(t-2)) + \cos(\pi(t-2.5))}{\pi} \right)$$

Region 8:

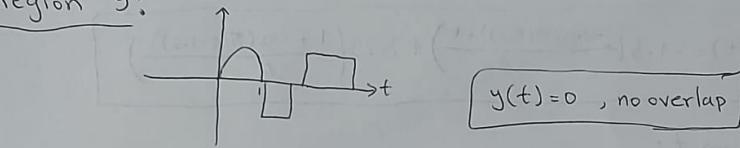


$$y(t) = \int_{t-2.5}^t -1.5 \sin(\pi t) dt$$

$$y(t) = -1.5 \left[-\frac{\cos(\pi t)}{\pi} \right]_{t=2.5}^t$$

$$y(t) = -1.5 \left(\frac{1 + \cos(\pi(t-2.5))}{\pi} \right)$$

Region 9:

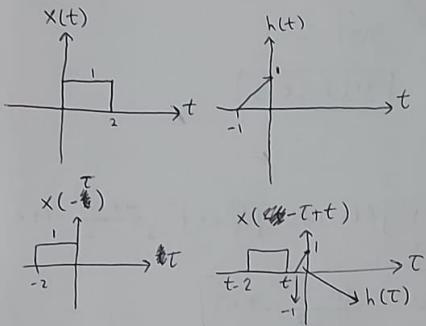


$$y(t) = 0, \text{ no overlap}$$

$$\text{B.2) } x(t) = u(t) - u(t-2)$$

$$h(t) = (t+1)(u(t+1) - u(t))$$

$$y(t) = x(t) * h(t)$$



Region 1: $t \leq -1$

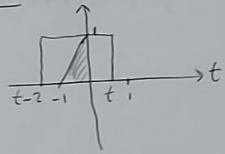
$$\boxed{y(t) = 0 \text{ (no overlap)}}$$

Region 2: $-1 \leq t \leq 0$

$$y(t) = \int_{-1}^t (t+\tau) d\tau = \left[t^2/2 + t \right]_{-1}^t = \frac{t^2}{2} + t - \frac{1}{2} + 1$$

$$\boxed{y(t) = \frac{t^2}{2} + t + \frac{1}{2}}$$

Region 3: $0 \leq t \leq 1$

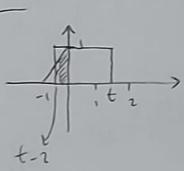


$$y(t) = \int (t+1) dt = \left[\frac{t^2}{2} + t \right]_0^1$$

$$\cdot y(-1) = \frac{1}{2} - 1$$

$$\boxed{y(t) = -0.5}$$

Region 4: $1 \leq t \leq 2$



$$y(t) = \int_{t-2}^0 (t+1) dt = \left[\frac{t^2}{2} + t \right]_{t-2}^0 = \frac{-(t-2)^2}{2} - t + 2$$

$$\boxed{y(t) = -\frac{(t-2)^2}{2} - t + 2}$$

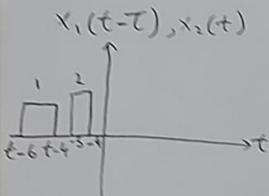
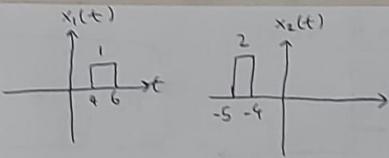
Region 5: $t > 2$

$$\boxed{y(t) = 0, \text{ no overlap}}$$

B.3-Part 1) $A=1, B=2$

$$x_1(t) = u(t-4) - u(t-6)$$

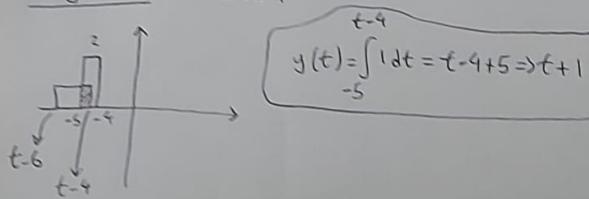
$$x_2(t) = ((u(t+5) - u(t+4)).2$$



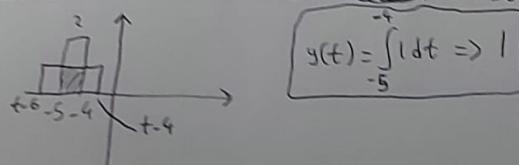
Region 1: $(t-4) \leq -5 \Rightarrow t \leq -1$

$$y(t) = 0 \text{ (no overlap)}$$

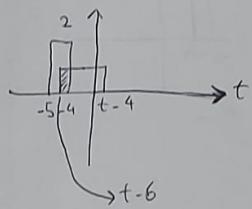
Region 2: $-5 \leq t \leq 4$



Region 3: ~~1234567890~~ $-4 \leq t \leq 0$



Region 4: $1 \leq t \leq 2$



$$y(t) = \int_{t-6}^{-4} 1 dt = -4 - t + 6 \Rightarrow 2 - t$$

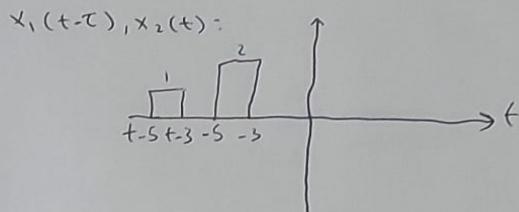
Region 5:

$$t > 2$$

$$y(t) = 0, \text{ no overlap}$$

B=3 - part 2) A=1, B=2

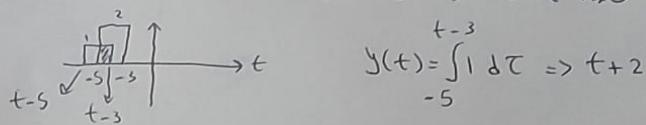
$$x_1(t) = u(t-3) - u(t-5), x_2(t) = 2(u(t+5) - u(t+3))$$



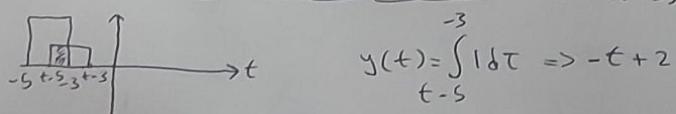
Region 1: $(t-3) < -5 \Rightarrow t < -2$

$$y(t) = 0 \text{ (no overlap)}$$

Region 2: $-5 < (t-3) < -3 \Rightarrow -2 < t < 0$



Region 3: ~~$-3 < (t-3) < 0$~~ $-3 < (t-3) < 0 \Rightarrow 0 < t < 3$

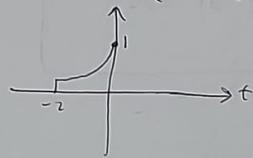


Region 4: ~~$(t-3) > 0$~~ $(t-3) > 0 \Rightarrow t > 3$

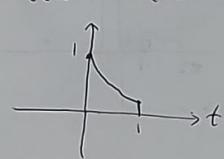
$$y(t) = 0 \text{ (no overlap)}$$

B.3 - Part 3) ~~ANSWER~~

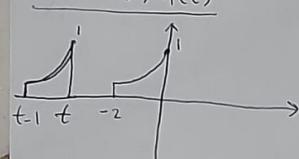
$$x_1(t) = e^t (u(t+2) - u(t))$$



$$x_2(t) = e^{-2t} (u(t) - u(t-1))$$



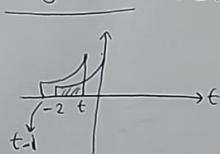
$$x_2(t-T), x_1(t)$$



Region 1: $t \leq -2$

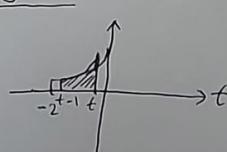
$$y(t) = 0 \text{ (no overlap)}$$

Region 2: $-2 \leq t \leq 0$

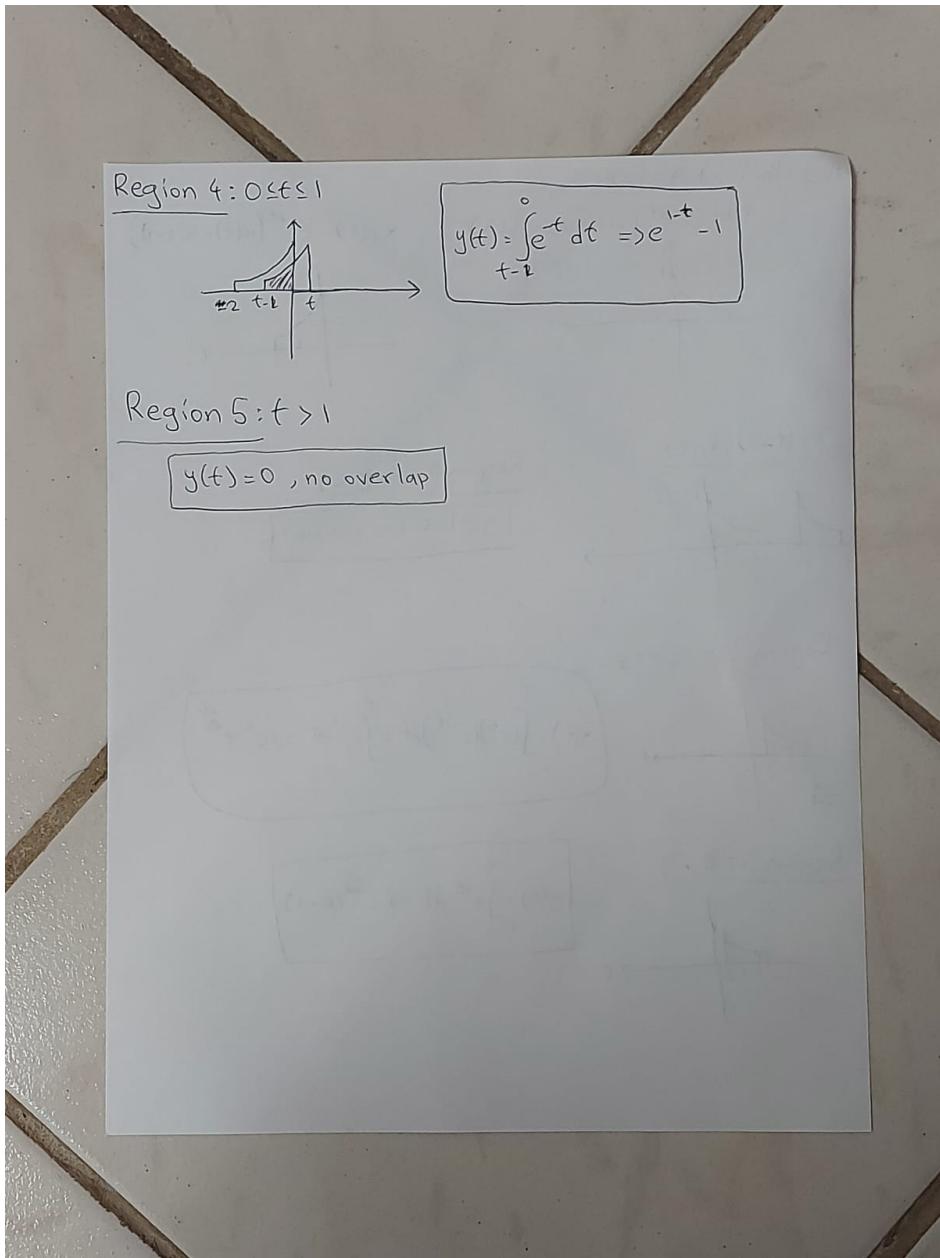


$$y(t) = \int_{-2}^t (e^t)(e^{-2t}) dt = \int_{-2}^t e^{-t} dt \Rightarrow e^2 - e^{-t}$$

Region 3: $t > 0$



$$y(t) = \int_{t-1}^t e^{-t} dt \Rightarrow e^{-t}(e-1)$$



Overall, all convolution calculations performed by hand are very accurate when compared to the MATLAB results and the same results are obtained in both cases.

D.2 What can you say about the width/duration of the signal resulting from the convolution of two signals?

It is observed that the width/duration of the signal resulting from the convolution of the two signals is equal to the sum of the duration of the two functions.