ELE532 Lab 2 Nini Yang

A.1:

Code:

```
% CH2MP1.m : chapter 2, p1

% 1
% script m file determines roots of op amp

% set component values
R = [1e4,1e4,1e4];
C = [1e-6, 1e-6];

% det coeffs for characteristic eqn
A = [1,  (1/R(1)+1/R(2)+1/R(3))/C(2), 1/(R(1)*R(2)*C(1)*C(2))];

% det roots:
lamba = roots(A);
```

Output:

```
>> lambda

lambda =

 -261.8034
  -38.1966
```

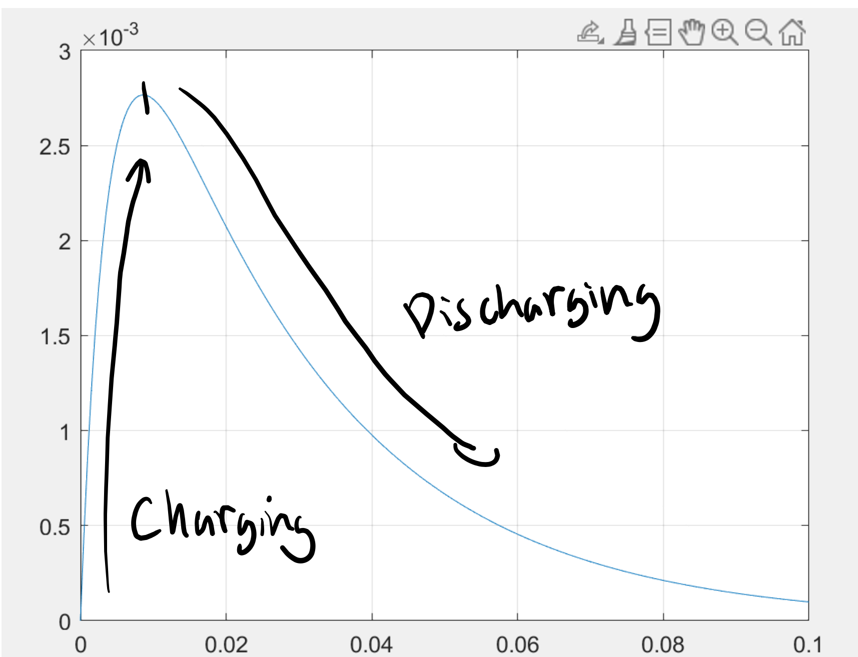
<u>A.2:</u>

Code:

```
Code:

% CH2MP1.m : chapter 2, p1

% 1

% script m file determines roots of op amp

% set component values

t =[0:0.0005:0.1];

h = @(t) (-0.0045*exp(-261.8034*t)+0.0045*exp(-38.1966*t)).*(t>0);

plot(t, h(t));

grid();
```

Report:

The graph represents the voltage output of C1 (see diagram below). When an impulse is delivered to the system, the capacitor charges up. While there is no impulse, the capacitor slowly discharges which is represented by the decreasing exponential graph.



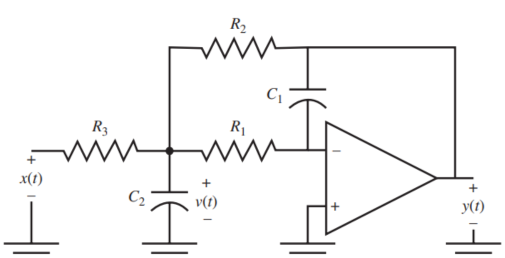


Figure 2.25 Operation-amplifier circuit.

A.3:

Code:

```
% CH2MP1.m : chapter 2, p1

function [lambda] = CH2MP2(R,C);

% CH2MP2.m : chpt 2, p 2
% Function find char roots of op amp circuit
% INPUTS: R = length-3 vector of resistances
% C = length-2 vector of capacitances
% OUTPUTS: lambda = characteristic roots

% Determine coefficients for characteristic equation:
A = [1, (1/R(1)+1/R(2)+1/R(3))/C(2), 1/(R(1)*R(2)*C(1)*C(2))];
% Determine characteristic roots:
lambda = roots(A);
```

Output:

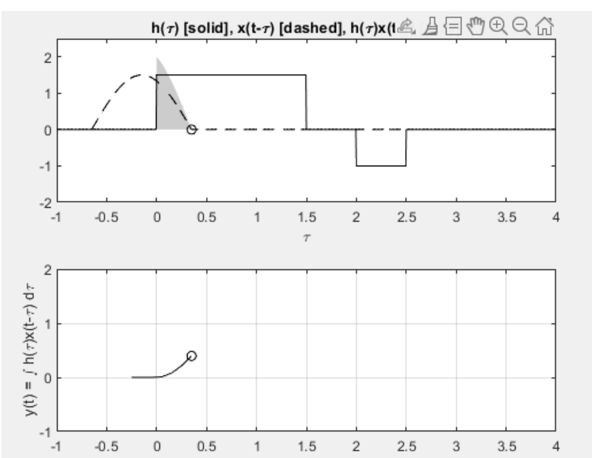
```
>> CH2MP2([1e4, 1e4, 1e4],[1e-9, 1e-6])

ans =

   1.0e+03 *

  -0.1500 + 3.1587i
  -0.1500 - 3.1587i
```

<u>B.1:</u>

Code:

```
% lab 2 B.1

% CH2MP4.m : chpt 2, p 4

% Script M-file graphically demonstrates the convolution process.

figure(1) % Create figure window and make visible on screen

u = @(t) 1.0*(t>=0);

x = @(t) 1.5*sin(pi*t).*(u(t)-u(t-1));

h = @(t) 1.5*(u(t)-u(t-1.5))-u(t-2)+u(t-2.5);

dtau = 0.005; tau = -1:dtau:4;

ti = 0; tvec = -.25:.1:3.75;

y = NaN*zeros(1,length(tvec)); % Pre-allocate memory

for t = tvec

ti = ti+1; % Time index

xh = x(t-tau).*h(tau); lxh = length(xh);

y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral

subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');

axis([tau(1) tau(end) -2.0 2.5]);

patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...

[zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...

[.8 .8 .8],'edgecolor','none');

xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');

c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);

subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');

xlabel('t'); ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');

axis([tau(1) tau(end) -1.0 2.0]); grid;

drawnow; pause (0.01);

end
```

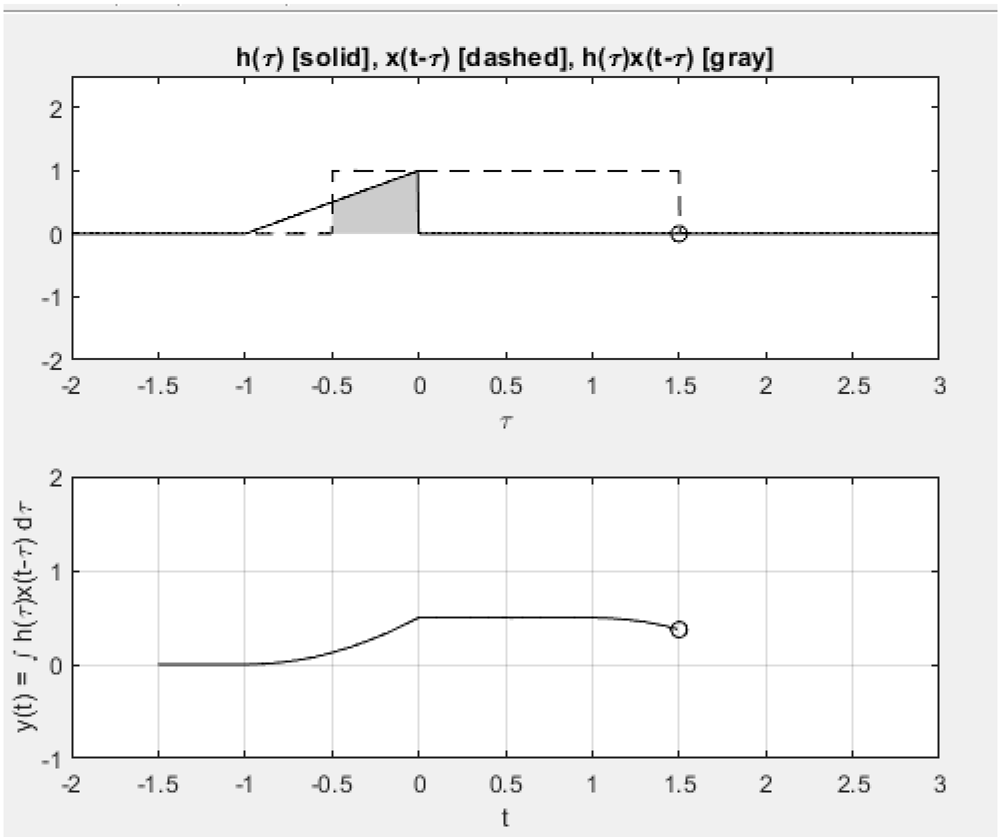
Output:



B.2:

Code:

```
% 2

% Create figure window and make visible on screen

u = @(t) 1.0*(t>=0);

x = @(t) u(t)-u(t-2);

h = @(t) (t+1).*(u(t+1)-u(t));

dtau = 0.005; tau = -2:dtau:3;

ti = 0;

tvec = -1.5:.1:3;

y = NaN*zeros(1,length(tvec)); % Pre-allocate memory

for t = tvec

ti = ti+1; % Time index

xh = x(t-tau).*h(tau); lxh = length(xh);

y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral

subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');

axis([tau(1) tau(end) -2.0 2.5]);

patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...

[zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...

[.8 .8 .8],'edgecolor','none');

xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');

c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);

subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');

xlabel('t');

ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');

axis([tau(1) tau(end) -1.0 2.0]); grid;

drawnow;

end
```


h(τ) [solid], x(t-τ) [dashed], h(τ)x(t-τ) [gray]
τ
y(t) = ∫ h(τ)x(t-τ) dτ
t

B.3:

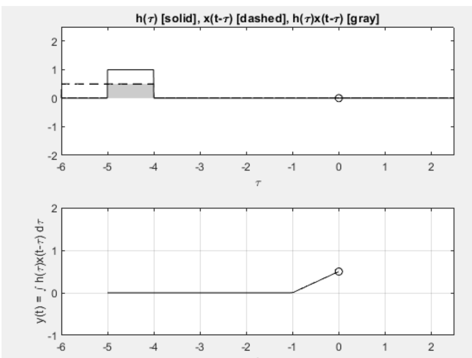
Code:

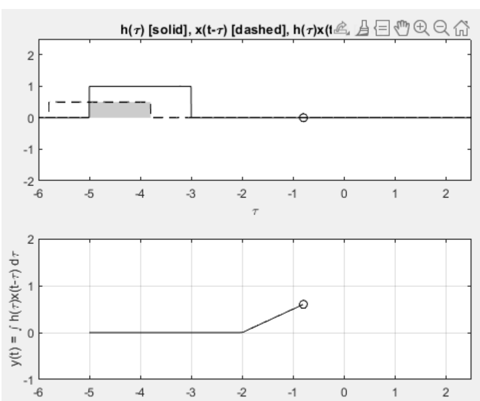
```
% 3 =========================================================
% Create figure window and make visible on screen
u = @(t) 1.0*(t>=0);
x = @(t) 0.5*(u(t-4)-u(t-6));
h = @(t) 1.0*(u(t+5)-u(t+4));
dtau = 0.005; tau = -6:dtau:2.5;
ti = 0;
tvec = -5:.1:5;
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for t = tvec
ti = ti+1; % Time index
xh = x(t-tau).*h(tau); lxh = length(xh);
y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
axis([tau(1) tau(end) -2.0 2.5]);
patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
[zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
[.8 .8 .8],'edgecolor','none');
xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
xlabel('t');
ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
axis([tau(1) tau(end) -1.0 2.0]); grid;
drawnow;
end
%}
% 4 =================================================================
%{
% Create figure window and make visible on screen
u = @(t) 1.0*(t>=0);
x = @(t) 0.5*(u(t-3)-u(t-5));
h = @(t) 1.0*(u(t+5)-u(t+3));
dtau = 0.005; tau = -6:dtau:2.5;
ti = 0;
```

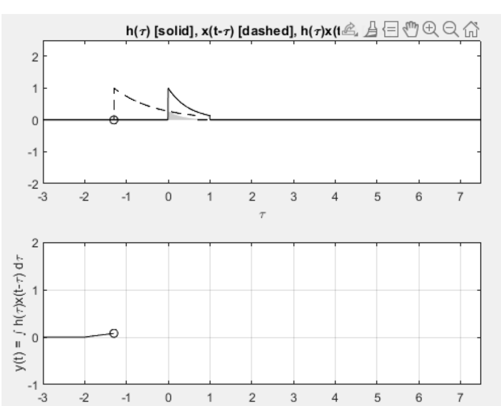
```
tvec = -5:.1:5;

y = NaN*zeros(1,length(tvec)); % Pre-allocate memory

for t = tvec

ti = ti+1; % Time index

xh = x(t-tau).*h(tau); lxh = length(xh);

y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral

subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');

axis([tau(1) tau(end) -2.0 2.5]);

patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...

[zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...

[.8 .8 .8],'edgecolor','none');

xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');

c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);

subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');

xlabel('t');

ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');

axis([tau(1) tau(end) -1.0 2.0]); grid;

drawnow;

end

%}

% 5 ======================================================================

%{

% Create figure window and make visible on screen

u = @(t) 1.0*(t>=0);

x = @(t) exp(t).*(u(t+2)-u(t));

h = @(t) exp(-2*t).*(u(t)-u(t-1));

dtau = 0.005; tau = -3:dtau:7.5;

ti = 0;

tvec = -5:.1:5;

y = NaN*zeros(1,length(tvec)); % Pre-allocate memory

for t = tvec

ti = ti+1; % Time index

xh = x(t-tau).*h(tau); lxh = length(xh);

y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral

subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');

axis([tau(1) tau(end) -2.0 2.5]);

patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...

[zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
```

```
[.8 .8 .8],'edgecolor','none');

xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');

c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);

subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');

xlabel('t');

ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');

axis([tau(1) tau(end) -1.0 2.0]); grid;

drawnow;

end

%}
```

Output:





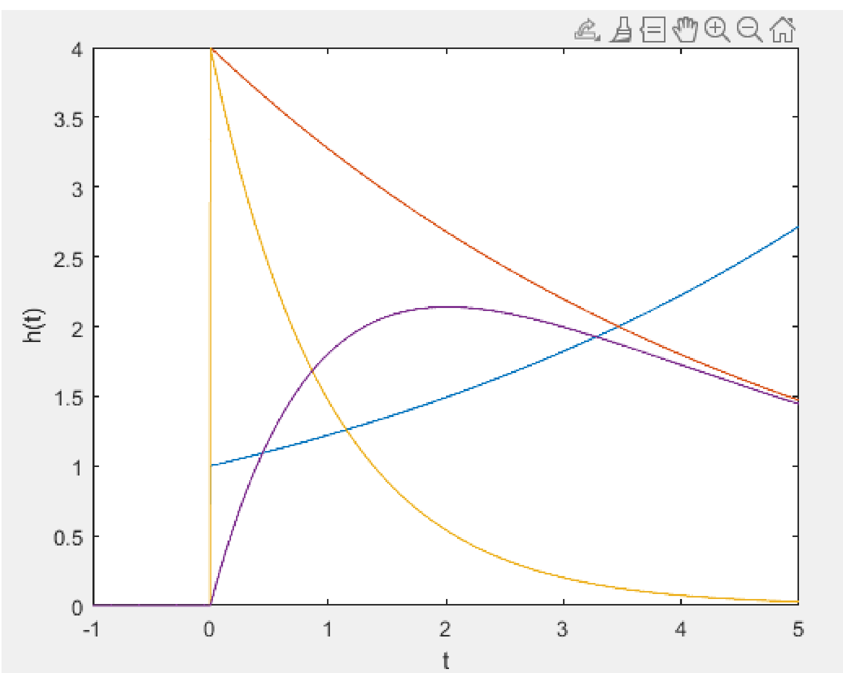


C.1:

Code:

```
t = [-1:0.001:5];

%4 functions
u = @(t) 1.0.*(t>=0);
h1 = @(t)exp(t/5).* u(t);
h2 = @(t)4*exp(-t/5).* u(t);
h3 = @(t)4*exp(-t).* u(t);
h4 = @(t)4*(exp(-t/5)-exp(-t)).* u(t);

% plotting
plot(t,h1(t));
xlabel ("t");
ylabel ("h(t)");
hold on;
plot(t,h2(t));
plot(t,h3(t));
plot(t,h4(t));
```

Output:



C.2:

Report:

Eigenvalues:

$h_1(t) = 1/5$

$h_2(t) = -1/5$

$h_3(t) = -1$

$h_4(t) = -1/5$ and -1

C.3:

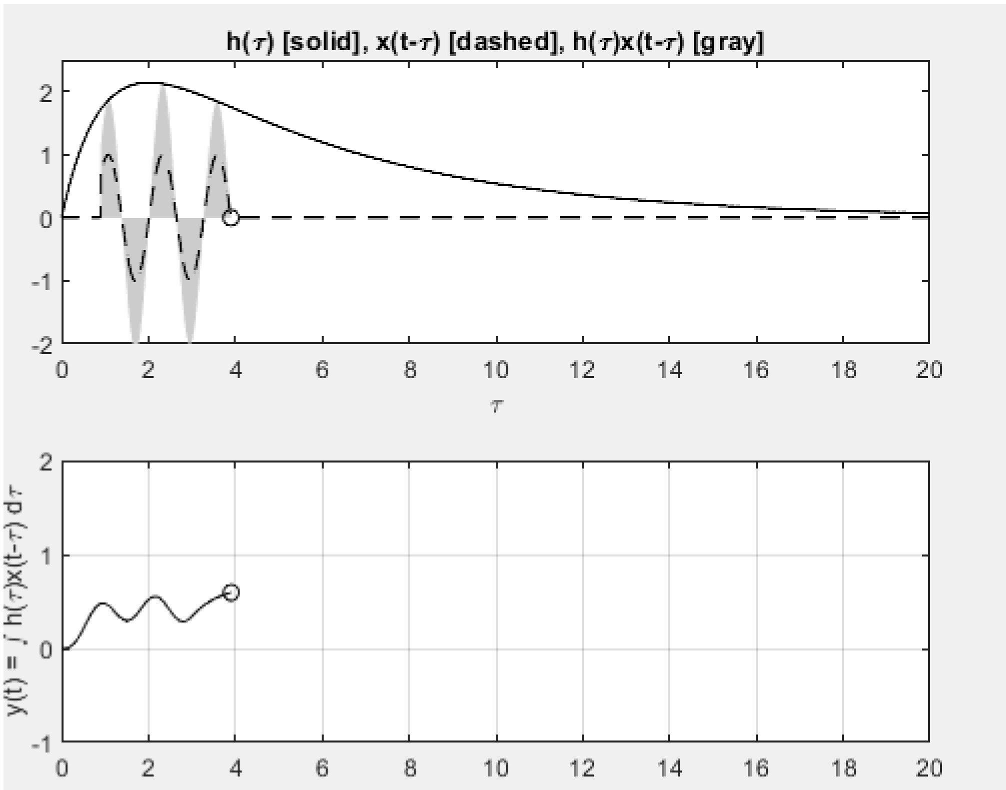
Code:

```
%5
% Create figure window and make visible on screen
u = @(t) 1.0*(t>=0);
x = @(t) sin(5*t).*(u(t)-u(t-3));
h = @(t) 4*(exp(-t/5)-exp(-t)).* u(t);
dtau = 0.005; tau = 0:dtau:20;
ti = 0;
tvec = 0:.1:20;
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for t = tvec
ti = ti+1; % Time index
xh = x(t-tau).*h(tau); lxh = length(xh);
y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
subplot(2,1,1),plot(tau,h(tau),'k-',tau,x(t-tau),'k--',t,0,'ok');
axis([tau(1) tau(end) -2.0 2.5]);
patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
[zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
[.8 .8 .8],'edgecolor','none');
xlabel('\tau'); title('h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-\tau) [gray]');
c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
subplot(2,1,2),plot(tvec,y,'k',tvec(ti),y(ti),'ok');
xlabel('t');
ylabel('y(t) = \int h(\tau)x(t-\tau) d\tau');
axis([tau(1) tau(end) -1.0 2.0]); grid;
drawnow;
end
```

Report:

Relationship between outputs:

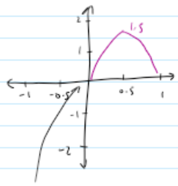
Fourth equation is a combination of equation 2 and 3: $h_4(t) = h_3(t) - h_2(t)$.

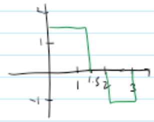


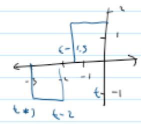
D.1:

1.

$x(t) = 1.5\sin(\pi t)[u(t) - u(t-1)]$ $\quad h(t) = 1.5[u(t) - u(t-1.5)] - u(t-2) + u(t-2.5)$

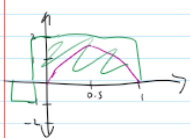






$t < 0$: $y(t) = 0$

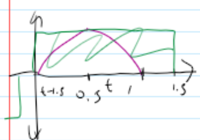
$0 < t < 1$



$$y(t) = \int_0^t 1.5(1.5)\sin\pi\tau \, d\tau$$
$$= 2.255 \int_0^t \sin\pi\tau \, d\tau$$
$$= \frac{2.255}{\pi}\left(-\cos(\pi\tau)\Big|_0^t\right)$$
$$= \frac{2.255}{\pi}\left(-\cos(\pi t) + \cos(0)\right)$$

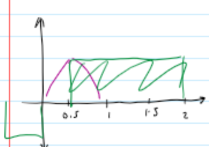
$1 < t < 1.5$

$$y(t) = \frac{2.255}{\pi}\left(1 - \cos(\pi t)\right)$$



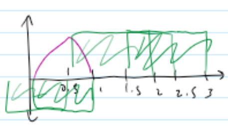
$$y(t) = \int_0^1 (1.5)(1.5\sin(\pi\tau)\,d\tau)$$
$$= -\frac{2.25}{\pi}\cos(\pi\tau)\Big|_0^1$$
$$= -\frac{2.25}{\pi}\left(\cos(\pi) - 1\right)$$
$$y(t) = \frac{4.5}{\pi} = 1.432$$

$1.5 < t < 2$



$$y(t) = \int_{t-1.5}^1 (1.5)(1.5\sin(\pi\tau))\,d\tau$$
$$= \frac{-2.25}{\pi}\cos(\pi\tau)\Big|_{t-1.5}^1$$
$$= \frac{-2.25}{\pi}\left(\cos\pi - \cos(t-1.5)\right)$$

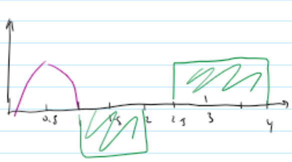
$2 < t < 3$



$$y(t) = \int_{t-1.5}^1 (1.5)(1.5\sin(\pi\tau))\,d\tau + \int_0^{t-2} (-1)(1.5\sin\pi\tau)\,d\tau$$
$$= 2.25\int\sin\pi\tau\,d\tau - 1.5\int\sin\pi\tau\,d\tau$$
$$= \frac{2.25}{\pi}\left(\cos\pi\tau\Big|_0^{t-2}\right) - \frac{1.5}{\pi}\left(\cos\pi\tau\Big|_{t-1.5}^1\right)$$
$$= \frac{2.25}{\pi}\left(\cos\pi(t-2) - \cos 0\right) - \frac{1.5}{\pi}\left(\cos\pi - \cos\pi(t-1.5)\right)$$

$3 < t < 4$

$$y(t) = \int_{t-3}^1 -1.5\sin(\pi\tau)\,d\tau$$

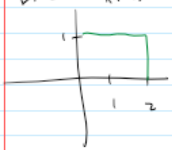


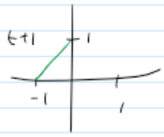
$$= -\frac{1.5}{\pi}\left(\cos\pi\tau\Big|_{t-3}^1\right)$$
$$= \frac{-1.5}{\pi}\left(\cos\pi - \cos(t-3)\right)$$

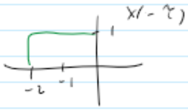
$4 < t$ $y(t) = 0$

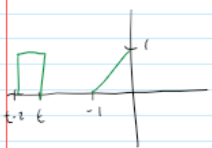
$$y(t) = \begin{cases} 0 & t < 0 \\ \frac{2.25}{\pi}\left(1 - \cos(\pi t)\right) & 0 < t < 1 \\ \frac{4.5}{\pi} & 1 < t < 1.5 \\ -\frac{2.25}{\pi}\left(\cos\pi - \cos(t-1.5)\right) & 1.5 < t < 2 \\ \frac{2.25}{\pi}\left(\cos\pi(t-2) - \cos 0\right) - \frac{1.5}{\pi}\left(\cos\pi - \cos\pi(t-1.5)\right) & 2 < t < 3 \\ \frac{-1.5}{\pi}\left(\cos\pi - \cos(t-3)\right) & 3 < t < 4 \\ 0 & 4 < t \end{cases}$$

B. 2 $x(t)$





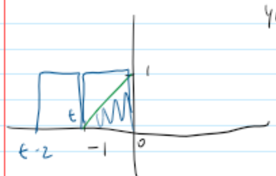




$t < -1$

$y(t) = 0$

$-1 < t < 0$



$$y(t) = \int_{-1}^{t} (\tau+1)\, d\tau$$

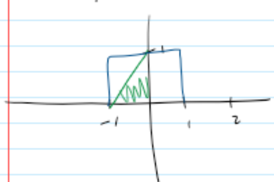
$$= \int \tau + \int 1$$

$$= \frac{\tau^2}{2}\Big|_{-1}^{t} + \tau\Big|_{-1}^{t}$$

$$= \frac{t^2}{2} - \frac{1}{2} + t + 1$$

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

$0 < t < 1$



$$y(t) = \int_{t-2}^{t} (\tau+1)\, d\tau$$

$$= \int \tau + \int 1$$

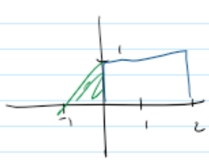
$$= \left[\frac{t^2}{2} - \frac{(t-2)^2}{2}\right] + \cancel{t} - (\cancel{t}-2)$$

$$= \frac{t^2}{2} - \frac{t^2-4t+4}{2} - 2$$

$$= \frac{t^2}{2} - \cancel{\frac{t^2}{2}} + \frac{4t}{2} - \frac{4}{2} - 2$$

$1 < t < 2$

$$y(t) = 2t - 4$$



$$y(t) = \int_{t-2}^{2} (\tau+1)$$

$$= \frac{\tau^2}{2}\Big|_{t-2}^{2} + \tau\Big|_{t-2}^{2}$$

$2 < t$ $y(t) = 0$

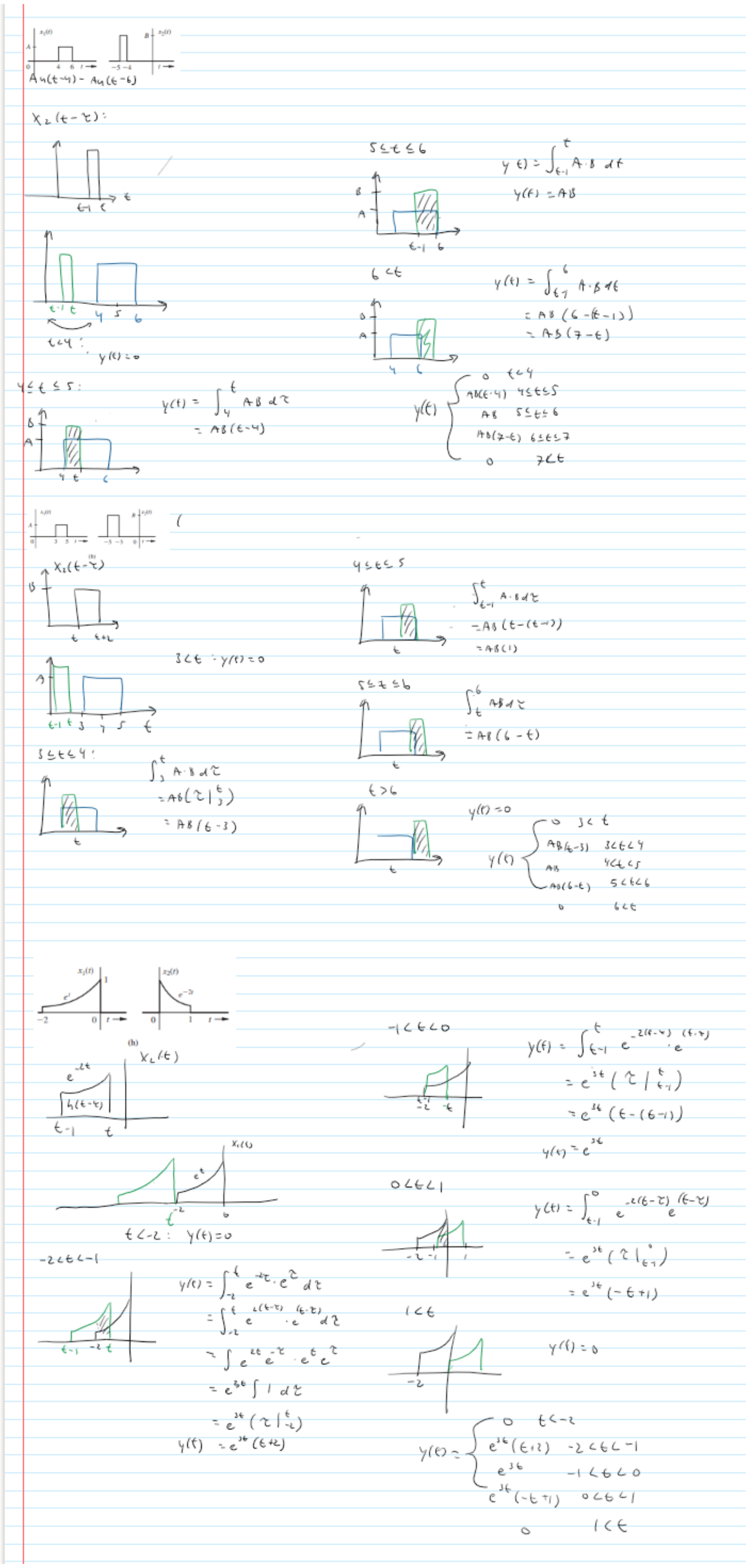
$$= \left(\frac{4}{2} - \frac{t^2-4t+4}{2}\right) + \cancel{2} + t - \cancel{2}$$

$$= \cancel{2} - \frac{t^2}{2} + \frac{4t}{2} - \cancel{\frac{4}{2}} + t$$

$$= -\frac{t^2}{2} + 3t$$

$$y(t) = \begin{cases} 0 & t < -1 \\ \frac{t^2}{2} + t + \frac{1}{2} & -1 < t < 0 \\ 2t - 4 & 0 < t < 1 \\ -\frac{t^2}{2} + 3t & 1 < t < 2 \\ 0 & 2 < t \end{cases}$$

b.3



<u>D.2:</u>

The width of the resulting signal is a combination of the widths of the signals being convoluted, since the left and right end points of the resulting signal equal the sum of the left and right end points of the functions being combined.