Table of Contents

A.1

```
% 3x 10k resistors
R = [1e4, 1e4, 1e4];
% 2x 1uF capacitors
C = [1e-6, 1e-6];

% Characteristic Equation
A1 = [1, (1/R(1)+1/R(2)+1/R(3))/C(2), 1/(R(1)*R(2)*C(1)*C(2))];

% Find roots (lambda)
lambda = roots(A1)

% Find coefficients a0, a1, and a2
poly(lambda)
```

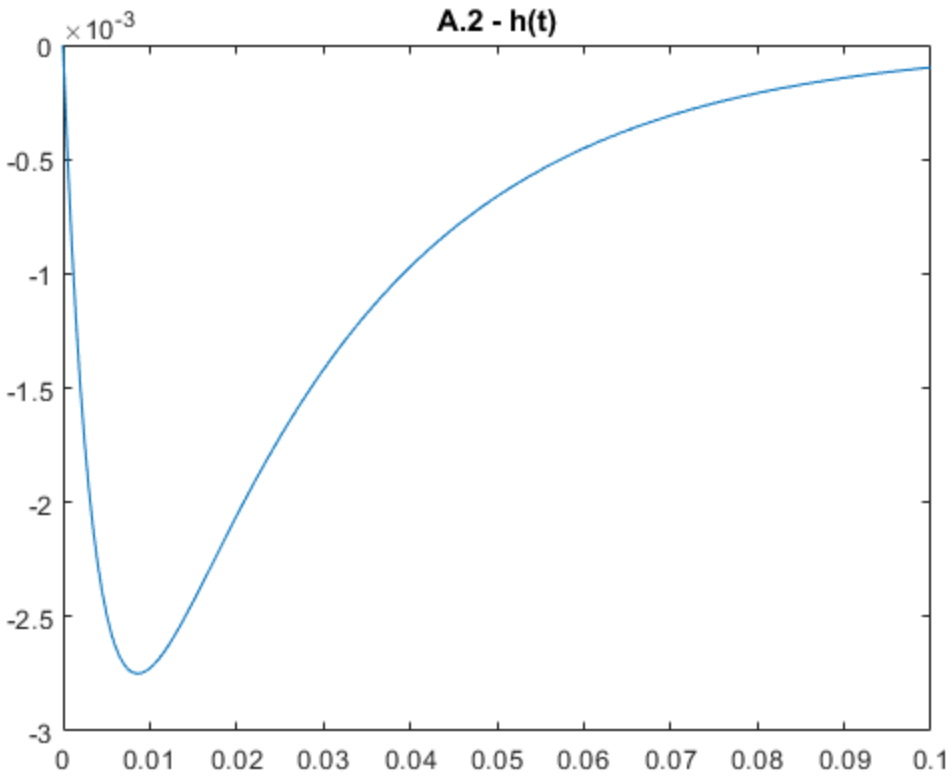
```
lambda =

 -261.8034
  -38.1966


ans =

   1.0e+04 *

    0.0001    0.0300    1.0000
```

A.2

```
t = 0:0.0005:0.1;
```

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

% y(t) =  c1e^(#1t) + c2e^(#2t) = 0
% y(0) => c1e^(#1(0)) + c2e^(#2(0)) = 0
%         c1e^(0) + c2e^(0) = 0
%         c1 + c2 = 0 (Equation 1)

% y'(t) =  #1c1e^(#1t) + #2c2e^(#2t), #1 = -261.8034, #2 = -38.1966
% y'(0) => (-261.8034)c1e^((-261.8034)(0)) + (-38.1966)c2e^((-38.1966)(0)) = 1
%          (-261.8034)c1e^(0) + (-38.1966)c2e^(0) = 1
%          -261.8034c1 - 38.1966c2 = 1 (Equation 2)

% Rewrite equation 1:
% Equation 1 => c1 + c2 = 0
%               c1 = -c2 (Equation 3)

% Sub equation 3 into equation 2:
% -261.8034c1 + 38.1966c1 = 1
% -223.6068c1 = 1
% c1 = 1 / -223.6068 = -4.47e-3
% c2 = -c1 = 4.47e-3

c1 = 1 ./ (lambda(2) - lambda(1))
c2 = -1 .* c1

h = @(t) (c1 .* exp(lambda(1) .* t) + c2 .* exp(lambda(2) .* t)) .* (u(t));

plot(t, h(t)); % Plot second order impulse response
title('A.2 - h(t)')
```

```
c1 =

    0.0045


c2 =

   -0.0045
```



A.3

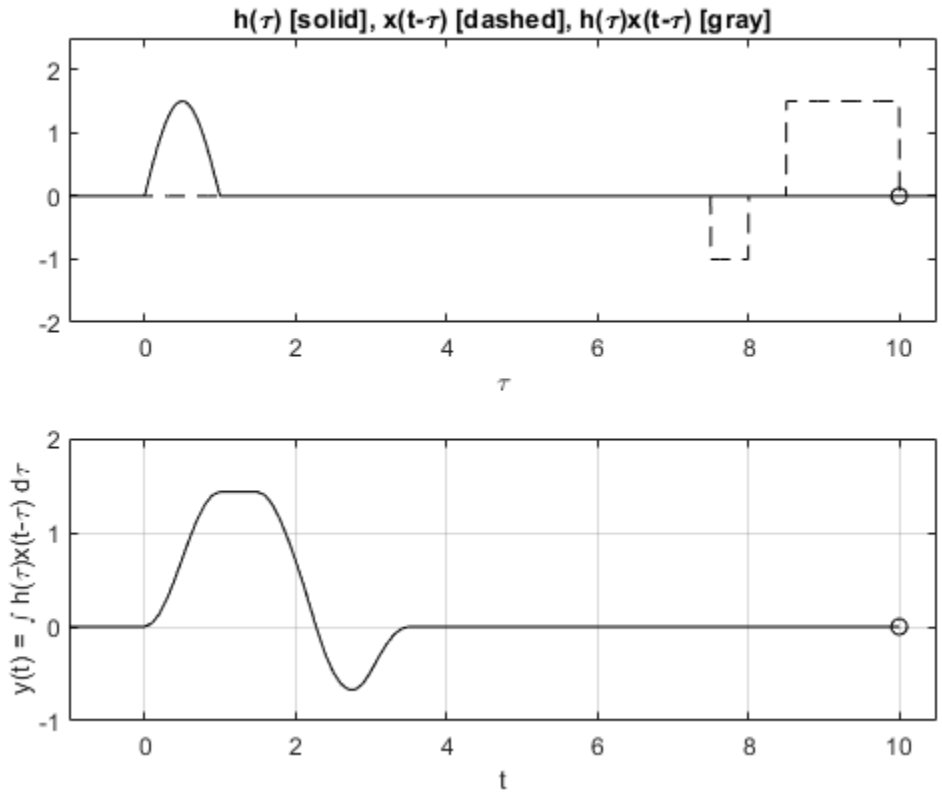
```
% <include>CH2MP2.m</include>
% Use function M-file
lambda = CH2MP2([1e4, 1e4, 1e4],[1e-9, 1e-6])
```

```
lambda =

   1.0e+03 *

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

B.1

```
figure(1) % Create figure window and make visible on screen
u = @(t) 1.0 * (t >= 0);
x = @(t) 1.5 * (u(t) -u(t - 1.5)) - u(t - 2) + u(t - 2.5);
h = @(t) 1.5 * sin(pi * t) .* (u(t) - u(t - 1));
dtau = 0.005; tau = -1:dtau:10.5;
ti = 0; tvec = -1:0.1:10;
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for t = tvec,
```

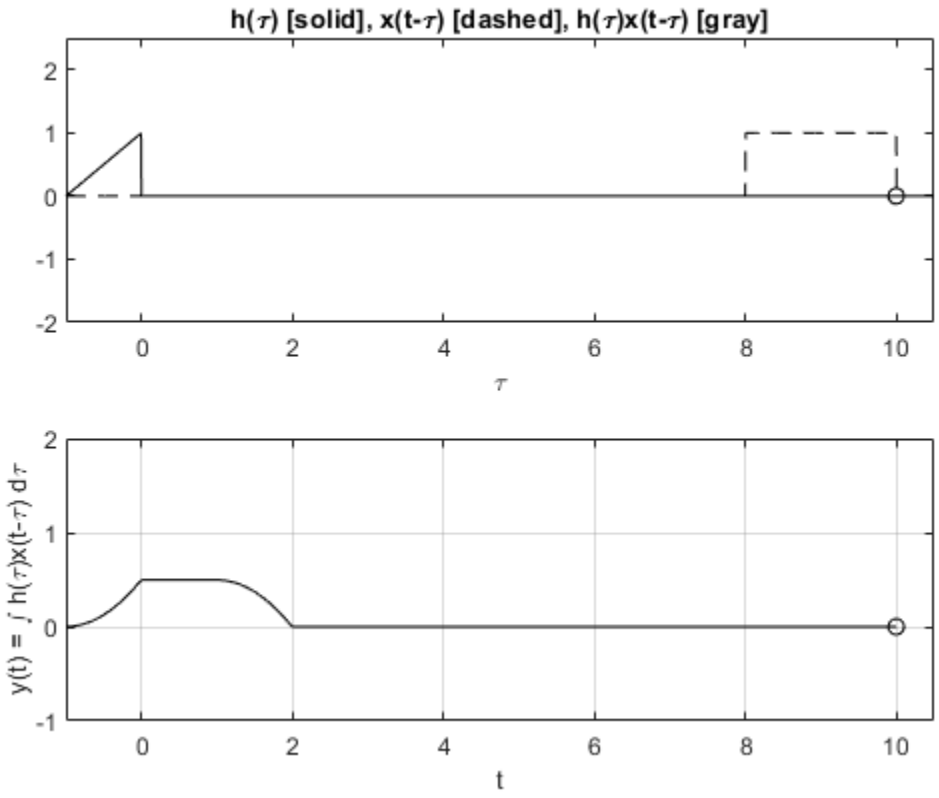
```
    ti = ti+1; % Time index
    xh = x(t-tau).*h(tau);
    lxh = length(xh);
    y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
    subplot(2,1,1),plot(tau,h(tau),"k-",tau,x(t-tau),"k--",t,0,"ok");
    axis([tau(1) tau(end) -2.0 2.5]);
    patch([tau(1:end-1);tau(1:end-1);tau(2:end);tau(2:end)],...
        [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;
    pause;
end
```



B.2

```
figure(1) % 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 = -1:dtau:10.5;
ti = 0; tvec = -1:0.1:10;
```

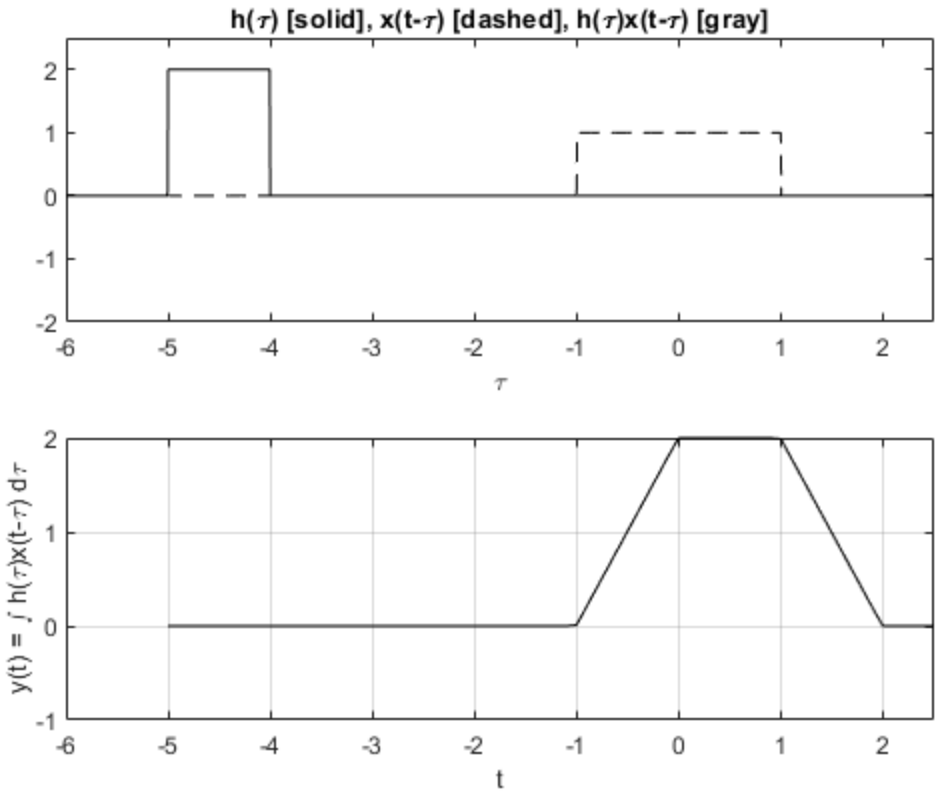
```
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for t = tvec,
    ti = ti+1; % Time index
    xh = x(t-tau).*h(tau);
    lxh = length(xh);
    y(ti) = sum(xh.*dtau); % Trapezoidal approximation of convolution integral
    subplot(2,1,1),plot(tau,h(tau),"k-",tau,x(t-tau),"k--",t,0,"ok");
    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;
    pause;
end
```



B.3.1

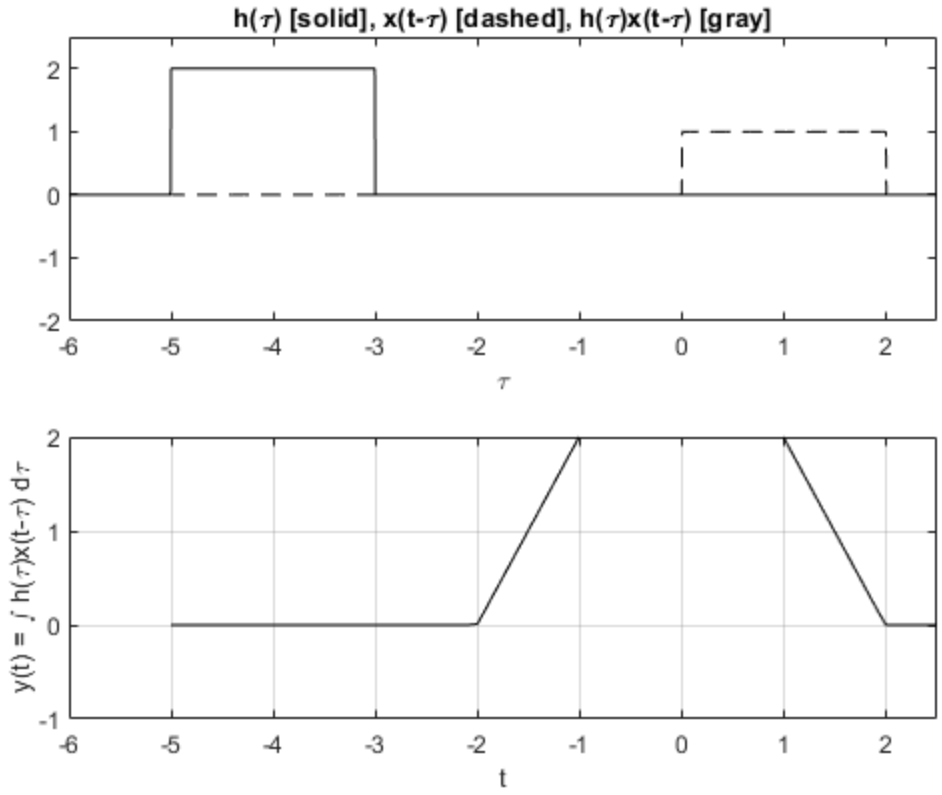
```
figure(1) % Create figure window and make visible on screen
A = 1.0;
B = 2.0;
u = @(t) 1.0 * (t >= 0);
```

```
x = @(t) A * (u(t - 4) - u(t - 6));
h = @(t) B * (u(t + 5) - u(t + 4));
dtau = 0.005; tau = -6:dtau:2.5;
ti = 0; tvec = -5:0.1:5;
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;
    pause;
end
```



B.3.2

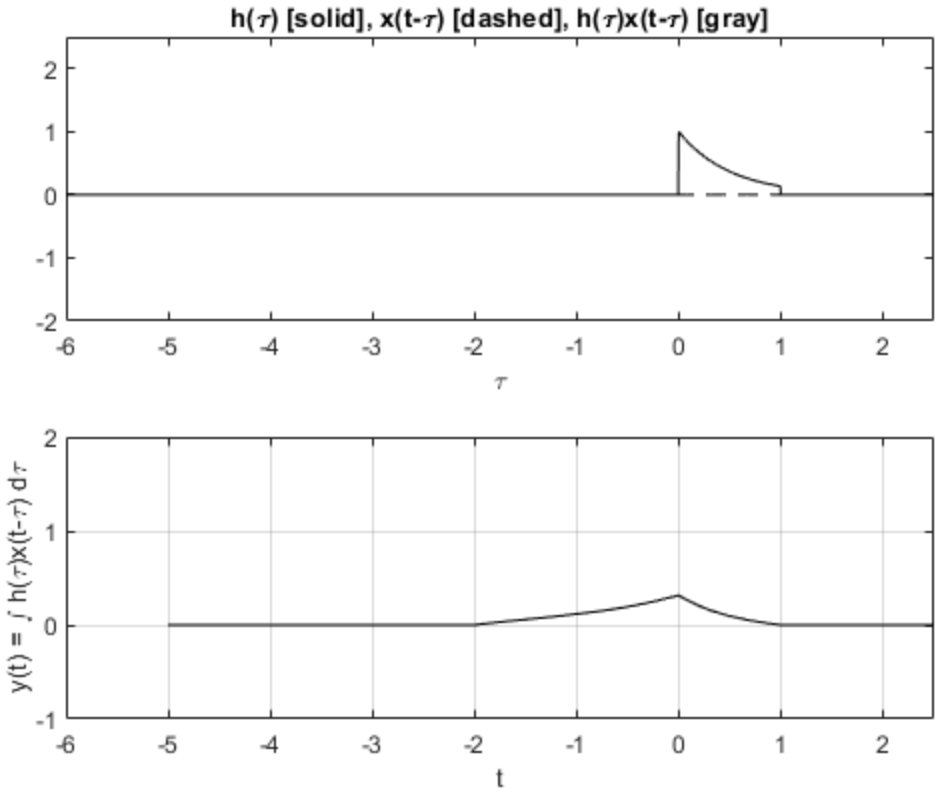
```
figure(1) % Create figure window and make visible on screen
A = 1.0;
B = 2.0;
u = @(t) 1.0 * (t >= 0);
x = @(t) A * (u(t - 3) - u(t - 5));
h = @(t) B * (u(t + 5) - u(t + 3));
dtau = 0.005; tau = -6:dtau:2.5;
ti = 0; tvec = -5:0.1:5;
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for t = tvec,
    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;
    pause;
end
```



B.3.3

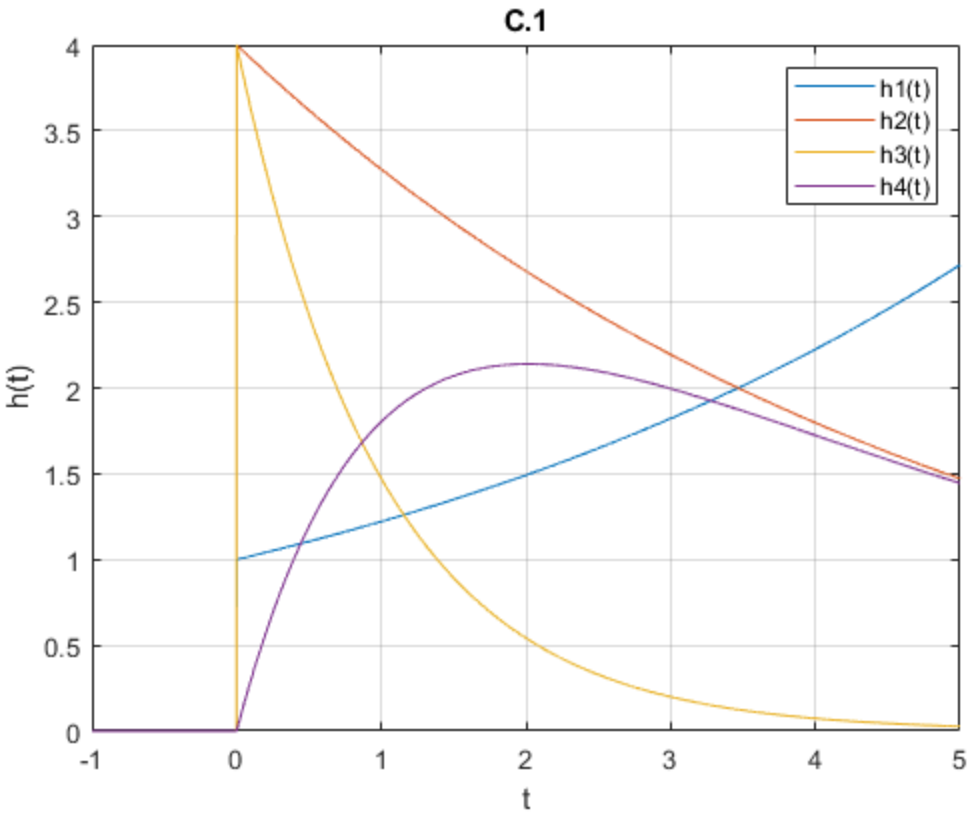
```
figure(1) % Create figure window and make visible on screen
A = 1.0;
B = 2.0;
u = @(t) 1.0 * (t >= 0);
x = @(t) exp(t) .* (u(t + 2) - u(t));
h = @(t) exp(-2 * t) .* (u(t) - u(t - 1));
dtau = 0.005; tau = -6:dtau:2.5;
ti = 0; tvec = -5:0.1:5;
y = NaN*zeros(1,length(tvec)); % Pre-allocate memory
for 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;
    pause;
end
figure;
```



C.1

```
%Period
t = -1:0.001:5;
% 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);

%Plot
plot(t,h1(t));
hold on;
plot(t,h2(t));
plot(t,h3(t));
plot(t,h4(t));
xlabel("t");
ylabel("h(t)");
title("C.1");
grid;
legend("h1(t)", "h2(t)", "h3(t)", "h4(t)");
hold off;
```



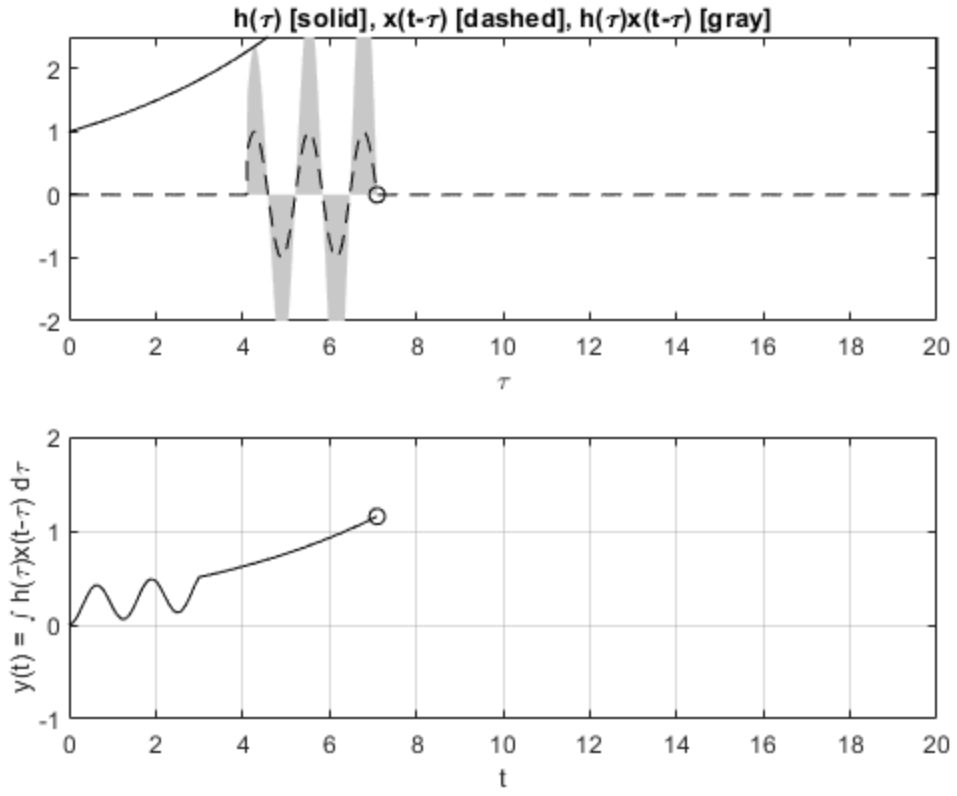
C.2

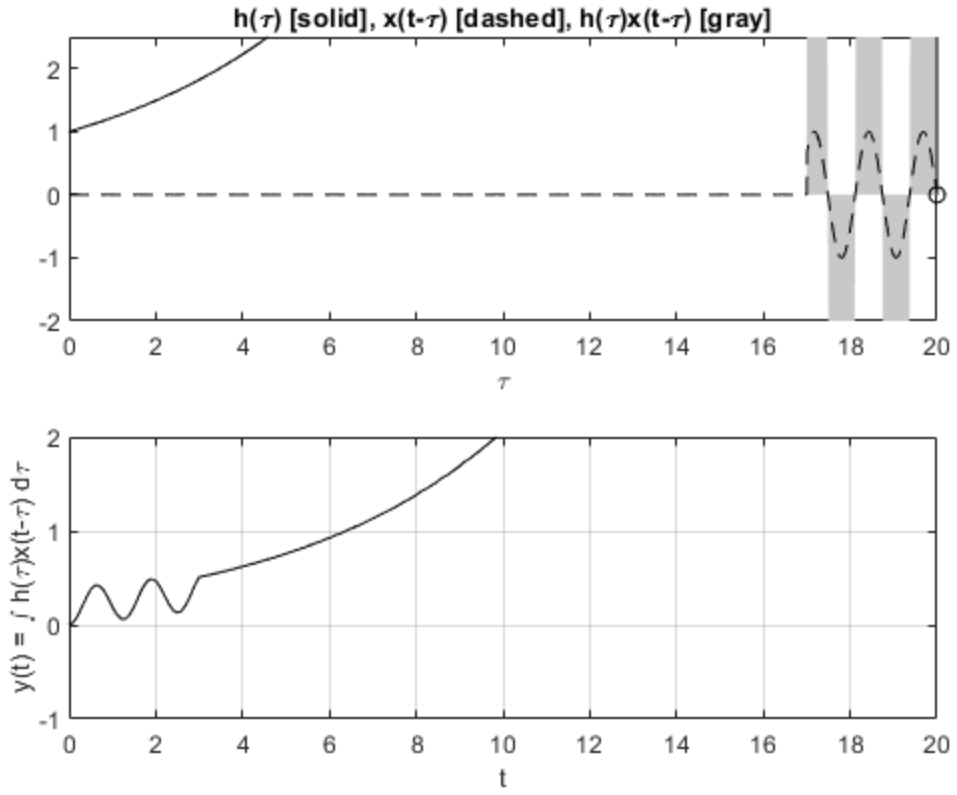
```
% 1/5
% -1/5
% -1
% -1/5, -1
% These values can be found just by looking at the exponents of e in each
% of the functions.
```

C.3 (h1)

```
u = @(t) 1.0 .* (t >= 0);
x = @(t) sin(5 .* t) .* (u(t) - u(t - 3));
h = @(t) exp(t / 5) .* (u(t) - u(t - 20));

dtau = 0.005;
tau = 0:dtau:20;
ti = 0;
tvec = 0:0.1:20;
y = NaN*zeros(1,length(tvec));

for t = tvec,
    ti = ti+1;
    xh = x(t-tau).*h(tau);
    lxh = length(xh);
```

```
    y(ti) = sum(xh.*dtau);
    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
```

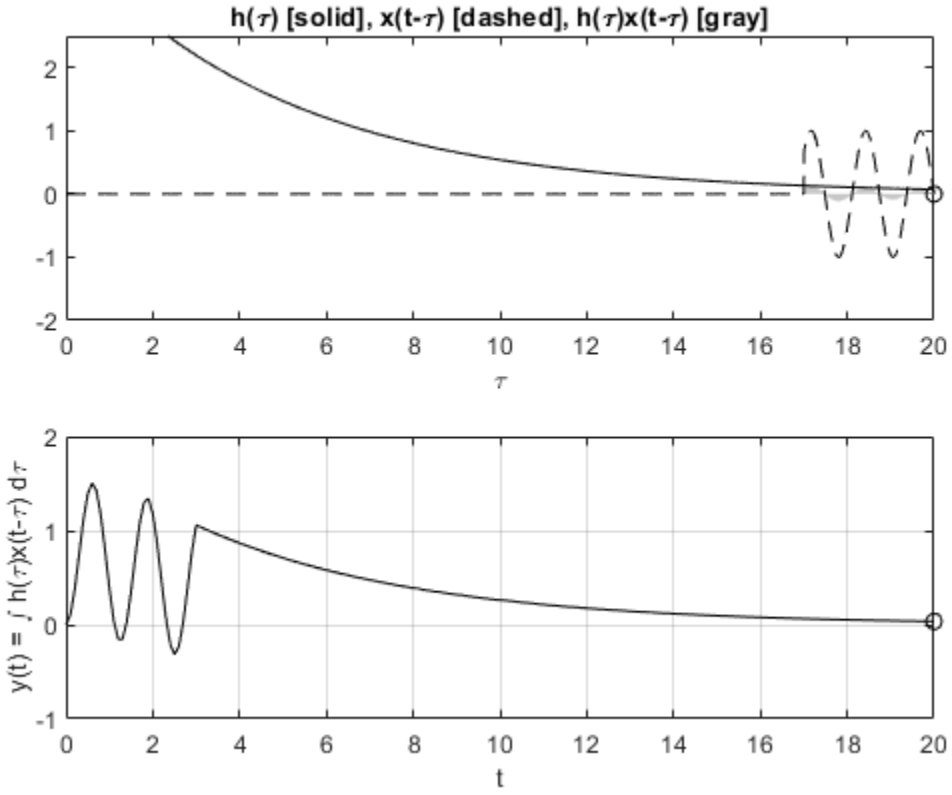




C.3 (h2)

```
u = @(t) 1.0 .*(t >= 0);
x = @(t) sin(5 .* t) .* (u(t) - u(t - 3));
h = @(t) 4 .* (exp(-t / 5)) .* (u(t) - u(t - 20));

dtau = 0.005;
tau = 0:dtau:20;
ti = 0;
tvec = 0:0.1:20;
y = NaN*zeros(1,length(tvec));

for t = tvec,
    ti = ti+1;
    xh = x(t-tau).*h(tau);
    lxh = length(xh);
    y(ti) = sum(xh.*dtau);
    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)]);
```

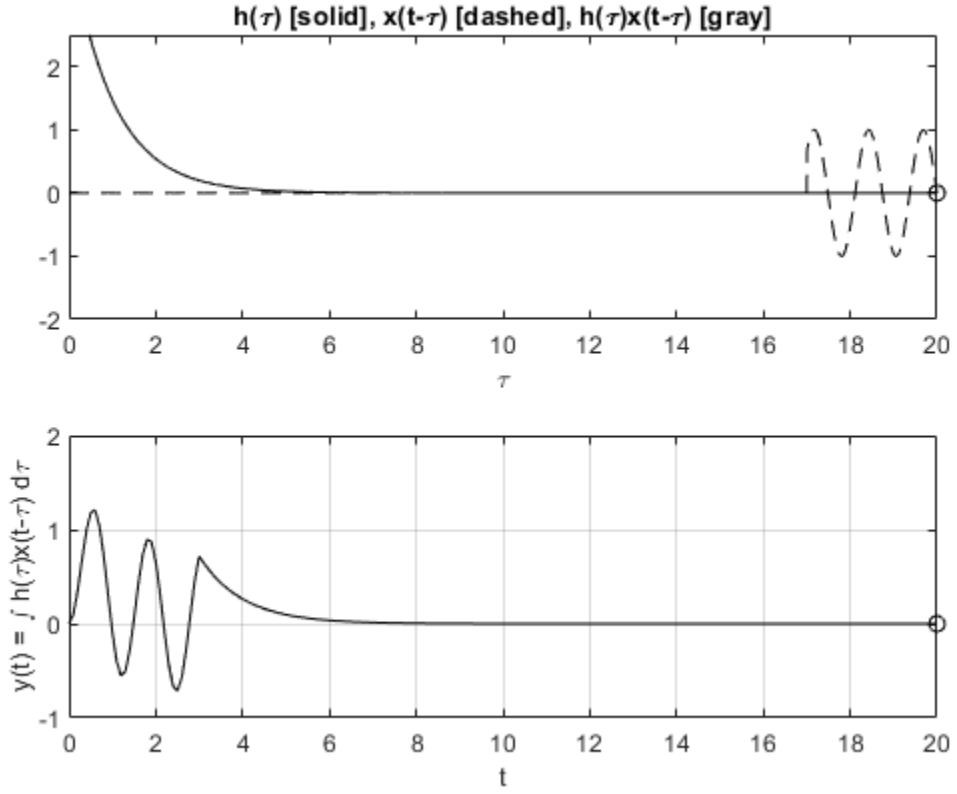
```
    subplot(2,1,2),plot(tvec,y,"k",tvec(ti),y(ti),"ok");
    xlabel("t"); ylabel("y(t) = \int h(\tau)x(t-\tau) d\tau");
    axis([tau(1) tau(end) -1.0 2.0]); grid;
    drawnow;
end
```



C.3 (h3)

```
u = @(t) 1.0 .* (t >= 0);
x = @(t) sin(5 .* t) .* (u(t) - u(t - 3));
h = @(t) 4 .* (exp(-t)) .* (u(t) - u(t - 20));

dtau = 0.005;
tau = 0:dtau:20;
ti = 0;
tvec = 0:0.1:20;
y = NaN*zeros(1,length(tvec));

for t = tvec,
    ti = ti+1;
    xh = x(t-tau).*h(tau);
    lxh = length(xh);
    y(ti) = sum(xh.*dtau);
    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)],...
```

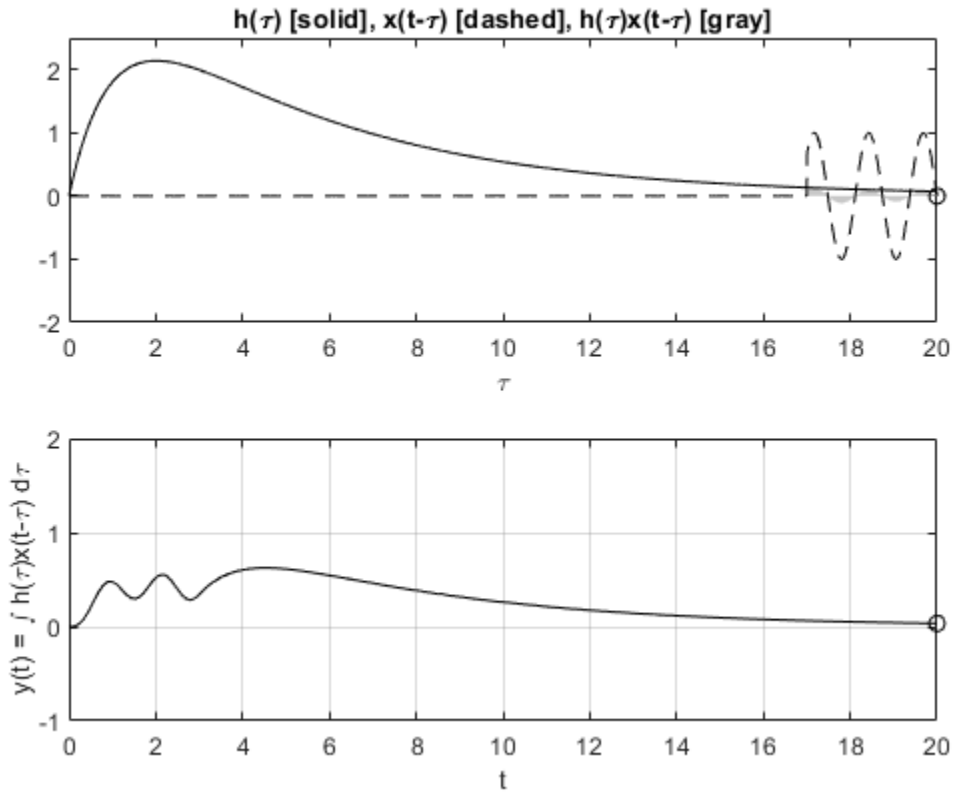
```
        [zeros(1,lxh-1);xh(1:end-1);xh(2:end);zeros(1,lxh-1)],...
        [.8 .8 .8],"edgecolor","none");
    xlabel("\tau"); title("h(\tau) [solid], x(t-\tau) [dashed], h(\tau)x(t-
\tau) [gray]");
    c = get(gca,'children'); set(gca,'children',[c(2);c(3);c(4);c(1)]);
    subplot(2,1,2),plot(tvec,y,"k",tvec(ti),y(ti),"ok");
    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
```



C.3 (h4)

```
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) - u(t - 20));

dtau = 0.005;
tau = 0:dtau:20;
ti = 0;
tvec = 0:0.1:20;
y = NaN*zeros(1,length(tvec));

for t = tvec,
    ti = ti+1;
    xh = x(t-tau).*h(tau);
```

```
    lxh = length(xh);
    y(ti) = sum(xh.*dtau);
    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
```



Published with MATLAB® R2023a