

```
% Name: Lamin Jammeh
% Class: EE480 Online
% Semester: Fall 2023
% HW_6

% Basic Problems
%% ***** 3.23a *****
clear;
clc;
% Step 1 define a syms function with t and s
syms t s;
% define the impulse response of the LTI system
h_t = exp(-2*t) * heaviside(t);
% define the input of the system;
x_t = heaviside(t) - heaviside(t-3);
% Step 2 change both the H(t) and x(t) to laplace transform
H_s = laplace(h_t);
X_s = laplace(x_t);
%define Y(s)
Y1_s = H_s * X_s;
Y_simplified = simplifyFraction(Y1_s);
% Step 3 change Y(s) to y(t) to get the output in time domain
y_t = ilaplace(Y1_s);

%% ***** 3.22a *****
clear;
clc;
%define syms in terms of t and s;
syms t s;
% define x(t)
x_t = heaviside(t) - 2*heaviside(t-1) + heaviside(t-2);
% define X(s) as laplace transform of x(t)
X_s = laplace(x_t);
%define Y(s)
Y1_s = ((s+2)*(1-exp(-s))^2) / (s*(s+1)^2);
%define H(s) as impulse response in laplace domain
H_s = Y1_s/X_s;
H_simplified = simplifyFraction(H_s);

%% ***** MATLAB Problems 3.32a *****
%Find the inverse laplace transform of X(s)
% define the syms as t and s
clear;
clc;
syms t s;
%define X(s)
X_s = (s^2 + 2*s + 1) / (s*(s+1)*(s^2 + 10*s + 50));
x_t = ilaplace(X_s);
x_simplified = simplifyFraction(x_t);
```

```
%% ***** MATLAB Problems 3.32b *****
%Find the inverse laplace transform of X(s)
clear;
clc;
% define the syms as t and s
syms t s;
X_s = (1-s*exp(-s)) / (s*(s+2)); % define X(s)
x_t = ilaplace(X_s); % define x(t)
x_simplified = simplifyFraction(x_t);
subplot(2,1,1)
fplot(x_simplified,[0,10], 'r', "LineWidth",2); % Plot x(t) as a function plot
xlabel('Time (sec)');
ylabel("x(t)");
title("x(t) vs Time Q3.32b");
ylim([-1,1]) % assign limits to y axis
grid on;
subplot(2,1,2)
fplot(X_s,[0,10], 'b', "LineWidth",2); %Plot X(s) as a function plot
xlabel('Time (sec)');
ylabel("X(s)");
title("X(s) vs Time Q3.32b");
ylim([0,20]) % assign limits to y axis
grid on;

%% ***** MATLAB Problems 3.35 *****
clear;
clc;
% determine h(t) from Y(s)
%Step 1 determine the numerator & denominator of the H(s)
num = [1,2]; %defines the numerator
denum = [1,2,1]; %define the denominator
%Step 2 define the transform function of the impulse response H(s)
H_s = tf(num, denum); % use the tf function for the H(s)
%Step 3 use the pole function to determine the poles
poles = pole(H_s);
%condition for BIBO stable
if pole(H_s) < 0
    disp('The system is BIBO stable')
else
    disp('The system is not BIBO stable')
end

%% ***** MATLAB Problems 3.37a *****
clear;
clc;
b1 = 5;
b0 = 6;
%Define syms function;
```

```
syms t s;
%Define the input of the system
x_t = heaviside(t);
X_s = laplace(x_t);
%Define H(s)
H_s = 1 / (s^2 + b1*s + b0);
%Define Y(s) = X(s)*H(s)
Y1_s = H_s * X_s;
y_t = ilaplace(Y1_s);
%plot y(t) using a fplot
fplot(y_t,[0,10],'r','LineWidth',2); % Plot y(t) as a function plot
xlabel('Time (sec)');
ylabel('y(t)');
title('y(t) vs Time Q3.37a');
grid on;
```

```
%% ***** MATLAB Problems 3.39b *****
```

```
clear;
clc;
%define syms as t and s
syms t s;
%define input of the system x(t)
x_t = heaviside(t);
X_s = laplace(x_t);

Y1_s = (s+1) / (s*(s^2 + 2*s + 4));
y1_t = ilaplace(Y1_s);
%define the transfer function of the system
H1_s = simplifyFraction(Y1_s / X_s)
% plot the poles and zeros
num1 = [1,1];
denum1 = [1,2,4];
H1_s = tf(num1,denum1);
figure
pzmap(H1_s)
title("pzmap of H1(s)");
grid on;
% define x and y axis limits
xlim([-2,0]) % assign limits to y axis
ylim([-2,2]) % assign limits to y axis
```

```
Y2_s = 1 / (s+2)^2;
y2_t = ilaplace(Y2_s);
H2_s = simplifyFraction(Y2_s / X_s)
% plot the poles and zeros
num1 = [1];
denum1 = [1,2,4];
H2_s = tf(num1,denum1);
```

```
figure
pzmap(H2_s)
title("pzmap of H2(s)");
grid on;
% define x and y axis limits
xlim([-2,2]) % assign limits to x axis
ylim([-2,2]) % assign limits to y axis

Y3_s = (s-1) / (s^2*((s+1)^2 + 9));
y3_t = ilaplace(Y3_s);
H3_s = simplifyFraction(Y3_s / X_s)
% plot the poles and zeros
num1 = [1,-1];
denum1 = [1,2,10,0];
H3_s = tf(num1,denum1);
figure
pzmap(H3_s)
title("pzmap of H3(s)");
grid on;
% define x and y axis limits
xlim([-2,2]) % assign limits to x axis
ylim([-2,2]) % assign limits to y axis
```