```
%% Homework 1
% MECH 6314.001
% Tanner Kogel
clear all; clc

%% Plot settings
set(0,'defaultLineLineWidth', 2)
set(0,'defaultAxesFontName' , 'Times')
set(0,'defaultTextFontName' , 'Times')
set(0,'defaultAxesFontSize' , 18)
set(0,'defaultTextFontSize' , 18)
set(0,'defaulttextinterpreter','latex')
set(0,'defaultlegendinterpreter','latex')
set(0,'defaultAxesGridLineStyle','-.')

%% problem 4a

figure(1)
clf
hold on
grid on
[x1,x2]=meshgrid(-3:0.125:3,-3:0.125:3);
x1_dot = x2;
x2_dot = x1.^4 + x1.^3 + 2*x1.^2 + 2*x1 - x2;
L = sqrt(x1_dot.^2 + x2_dot.^2);
q=quiver(x1,x2,x1_dot./L,x2_dot./L,0.4);
q.Color='black';
q.AutoScale='on';
q.MaxHeadSize=0.2;
axis([-2 2 -2 2]);
title('System 4a')
xlabel('$x_1$','interpreter','Latex'); ylabel('$x_2$','interpreter','Latex')
xic = [-1  0 0.5 -1 0.5 1 -1.5 1.5 0.25 0];
yic = [-2 -2 -2  -3 -1 -2  2.5 -2   -1  2];
for ic = [xic;yic]
[~,x] = ode45(@system4a,[0,12],ic);
plot(x(:,1),x(:,2),'k');
end


%% problem 4b

figure(2)
clf
hold on
grid on
[x1,x2]=meshgrid(-3:0.125:3,-3:0.1125:3);
x1_dot = 2*x1 - x1.*x2;
x2_dot = -x2/2 + x1.*x2;
L = sqrt(x1_dot.^2 + x2_dot.^2);    %normalize the length of the arrows
q=quiver(x1,x2,x1_dot./L,x2_dot./L,0.4);
```

```
q.Color='black';
q.AutoScale='on';
q.MaxHeadSize=0.2;
axis([-2 2 -1 3]);
title('System 4b')
xlabel('$x_1$','interpreter','Latex'); ylabel('$x_2$','interpreter','Latex')
xic = [0.5  1  0  0 0.5 -0.25 0.25];
yic = [1.5 1.5 3 -3 0.5    3  -0.5];
for ic = [xic;yic]
[~,x] = ode45(@system4b,[0,9],ic);
plot(x(:,1),x(:,2),'k');
end


%% problem 4c

figure(3)
clf
hold on
grid on
[x1,x2]=meshgrid(-3:0.125:3,-3:0.125:3);
x1_dot = 2*x1 - 2*x1.^2 - x1.*x2;
x2_dot = -x1/2 + x1.*x2;
L = sqrt(x1_dot.^2 + x2_dot.^2);    %normalize the length of the arrows
q=quiver(x1,x2,x1_dot./L,x2_dot./L,0.4);
q.Color='black';
q.AutoScale='on';
q.MaxHeadSize=0.2;
axis([-2 2 -2 2]);
title('System 4c')
xlabel('$x_1$','interpreter','Latex'); ylabel('$x_2$','interpreter','Latex')
xic = [2  2 -0.125 0.125   2     2   0.125 0.125 0.125 2 -0.125 -0.125 -0.125 -0.125↙
-0.125];
yic = [1 0.5  0.5  0.375 0.375 0.625 0.625   1     0   0    2     -2     -1      1↙
0];
for ic = [xic;yic]
[~,x] = ode45(@system4c,[0,9],ic);
plot(x(:,1),x(:,2),'k');
end


%% problem 8b

z0 = [1; 0; -1; 0]; % define initial condition
u = @(t)0*t;        % define input functino as 0

[t,z] = ode45(@(t,z)system8(u,t,z),[0,20],z0); % simulate response of the system

% plot position figures
figure(4)
clf
```

```
hold on
grid on
plot(t,z(:,1));
plot(t,z(:,3));
title('Position of System 8b')
xlabel('Time'); ylabel('Position')
legend('$x_1$','$x_2$','interpreter','Latex');

% plot velocity figures
figure(5)
clf
hold on
grid on
plot(t,z(:,2),'LineWidth',2);
plot(t,z(:,4),'Linewidth',2);
title('Velocity of System 8b')
xlabel('Time'); ylabel('Velocity')
legend('$v_1$','$v_2$','interpreter','Latex');


%% problem 8c

% define constants and initial conditions
A = 5;
omega = pi;
z0 = [0;0;0;0];

u = @(t)A*sin(omega*t); % define input function

[t,z] = ode45(@(t,z)system8(u,t,z),[0,20],z0); % simulate response of the system

% plot position figures
figure(6)
clf
hold on
grid on
plot(t,z(:,1));
plot(t,z(:,3));
title('Position of System 8c')
xlabel('Time'); ylabel('Position')
legend('$x_1$','$x_2$','interpreter','Latex');

% plot velocity figures
figure(7)
clf
hold on
grid on
plot(t,z(:,2),'LineWidth',2);
plot(t,z(:,4),'Linewidth',2);
title('Velocity of System 8c')
xlabel('Time'); ylabel('Velocity')
```

```
legend('$v_1$','$v_2$','interpreter','Latex');


%% Functions

%ode45 function for system 4a
function dxdt = system4a(~,x)
x1d=x(1);
x2d=x(2);
dxdt = [x2d; (x1d^4 + x1d^3 + 2*x1d^2 + 2*x1d - x2d) ];
end

%ode45 function for system 4b
function dxdt = system4b(~,x)
x1d=x(1);
x2d=x(2);
dxdt = [ (2*x1d - x1d*x2d) ; (x1d*x2d - x2d/2) ];
end

%ode45 function for system 4b
function dxdt = system4c(~,x)
x1d=x(1);
x2d=x(2);
dxdt = [ (2*x1d - 2*x1d^2 - x1d*x2d) ; (x1d*x2d - x1d/2) ];
end

%ode45 function for system 8
function dxdt = system8(u,t,z)
z1=z(1);
z2=z(2);
z3=z(3);
z4=z(4);
% define constants
b1 = 1;
b2 = 1;
k1 = 1;
k2 = 1;
M1 = 1;
M2 = 1;
dxdt = [ z2 ; ( -z1*(k1+k2)/M1 - z2*(b1+b2)/M1 + z3*k2/M1 + z4*b2/M1 ) ; z4 ; ( z1*k2/M2↙
+ z2*b2/M2 - z3*k2/M2 - z4*b2/M2 + u(t)/M2 ) ];
end
```