

---

## Table of Contents

Part a) .....	1
Part b) .....	1
Part c) .....	2

### Part a)

```
clear; close all; clc;
load('pendulum.mat');

% B matrix when the only input is torque
B_torque = [0; 0; -0.1; 1.1];

% controllability matrix
CM = ctrb(A,B_torque);
if rank(CM)~=4
    display('not controllable');
end

% % controllable decomposition
% C = eye(4);
% [Abar, Bbar, Cbar, T] = ctrbf(A,B_torque,C);

% Eigen vector test for stabilizability
% computing eigenvalues and their corresponding eigen vectors
[V,D] = eig(A');

% first and fourth column of V are the eigenvector corresponding to
% the eigenvalue >= 0
B_torque'*V(:,1);
B_torque'*V(:,4);

% the fourth column eigenvector is in the kernel of B', thus not
% stabilizable !

not controllable
```

### Part b)

```
clear; close all; clc;
load('pendulum.mat');

% observability matrix for position
C_position = [1 0 0 0];
OM = obsv(A,C_position);
if rank(OM)~=4
    display('not observable');
else
    display('observable when position is measured');
```

---

```

end

% observability matrix for angle
C_angle = [0 1 0 0];
OM = obsv(A,C_angle);
if rank(OM)~=4
    display('not observable');
else
    display('observable when angle is measured');
end

% % observability decomposition
% [Abar, Bbar, Cbar, T] = obsvf(A,B,C_angle);

[V,D] = eig(A);
% first and second columns of V are the eigenvectors corresponding
% eigenvalues >=0
% check if they are in the kernel of C matrix
C_angle*V(:,1)
C_angle*V(:,2)
% the first column of V is in the kernel of C matrix, thus not
% detectable.

observable when position is measured
not observable

ans =

    0

ans =

   -0.2928

```

## Part c)

```

clear; close all; clc;
load('pendulum.mat');

C = [0 0 1 0; 0 1 0 0];
OM = obsv(A,C);
if rank(OM)~=4
    display('not observable');
else
    display('observable when p_dot and theta are measured');
end

D = zeros(2,2);
sys = ss(A,B,C,D);
% invariant zero
inv_zero = tzero(sys); % invariant zero = 0

```

---

```
% transmission zero
trans_zero = tzero(minreal(sys));
% there is no transmission zero?

% Rosenbrock matrix
syms s
P = [s*eye(4)-A B;
     -C D];
det(P)

s = inv_zero;
P = [s*eye(4)-A B;
     -C D];
null(P)

% tf = C*inv(s*eye(4)-A)*B + D;

not observable
1 state removed.

ans =

s/10

ans =

     1
     0
     0
     0
     0
     0
```

*Published with MATLAB® R2016a*