Table of Contents

Question 1

```
A = [[1 1 0];[0 1 0];[0 0 1]];
B = [0;1;1];
C = [[1 1 0];[0 1 0]];
D = 0;

Omega_r = [B A*B A*A*B];
rank(Omega_r);

Omega_o = [C;C*A;C*A*A];
rank(Omega_o);
null(Omega_o)


ans =

     0
     0
     1
```

Question 2.b

```
syms omega
A = [[0 1 0 0];[3*omega^2 0 0 2*omega];[0 0 0 1];[0 -2*omega 0 0]];
B = [[0 0];[1 0];[0 0];[0 1]];

Omega_r = [B A*B A*A*B A*A*A*B]
if rank(Omega_r) ~= min(size(A))
    fprintf('the system is not fully reachable')
else
    fprintf('the system is fully reachable')
end


Omega_r =

[0, 0,        1,        0,         0,     2*omega,   -omega^2,          0]
```

```
[1, 0,        0, 2*omega, -omega^2,         0,         0, -2*omega^3]
[0, 0,        0,       1, -2*omega,         0,         0, -4*omega^2]
[0, 1, -2*omega,       0,        0, -4*omega^2, 2*omega^3,          0]
```

the system is fully reachable

Question 2.c - no axial thruster

```
syms omega
A = [[0 1 0 0];[3*omega^2 0 0 2*omega];[0 0 0 1];[0 -2*omega 0 0]];
B = [[0 0];[0 0];[0 0];[0 1]];

Omega_r = [B A*B A*A*B A*A*A*B]
if rank(Omega_r) ~= min(size(A))
    fprintf('the system with no axial thruster is not fully reachable')
else
    fprintf('the system is fully reachable, even without an axial thruster')
end
```

```
Omega_r =

[0, 0, 0,       0, 0,    2*omega, 0,          0]
[0, 0, 0, 2*omega, 0,          0, 0, -2*omega^3]
[0, 0, 0,       1, 0,          0, 0, -4*omega^2]
[0, 1, 0,       0, 0, -4*omega^2, 0,          0]
```

the system is fully reachable, even without an axial thruster

Question 2.d - no tangential thruster

```
syms omega
A = [[0 1 0 0];[3*omega^2 0 0 2*omega];[0 0 0 1];[0 -2*omega 0 0]];
B = [[0 0];[1 0];[0 0];[0 0]];

Omega_r = [B A*B A*A*B A*A*A*B]
if rank(Omega_r) ~= min(size(A))
    fprintf('the system with no tangential thruster is not fully reachable')
else
    fprintf('the system is fully reachable, even without a tangential
 thruster')
end
```

```
Omega_r =

[0, 0,        1, 0,        0, 0,  -omega^2, 0]
[1, 0,        0, 0, -omega^2, 0,         0, 0]
[0, 0,        0, 0, -2*omega, 0,         0, 0]
[0, 0, -2*omega, 0,        0, 0, 2*omega^3, 0]
```

the system with no tangential thruster is not fully reachable

Question 2.e - transfer function computation

```
syms omega s
A = [[0 1 0 0];[3*omega^2 0 0 2*omega];[0 0 0 1];[0 -2*omega 0 0]];

adjoint((s*eye(4) - A))
```

```
ans =

[4*omega^2*s + s^3,          s^2,               0,         2*omega*s]
[    3*omega^2*s^2,          s^3,               0,       2*omega*s^2]
[       -6*omega^3,   -2*omega*s, omega^2*s + s^3,   s^2 - 3*omega^2]
[     -6*omega^3*s, -2*omega*s^2,               0, s^3 - 3*omega^2*s]
```

Question 3 - circuit analysis

```
A = [[-2 0];[0 2]];
B = [0; 1];
C = [1 0];

Omega_r = [B A*B]
Omega_o = [C;C*A]
```

```
Omega_r =

     0     0
     1     2


Omega_o =

     1     0
    -2     0
```

Question 4 & 5 - checking chariot eigendecomposition

```
a = 0.2; % ratio of d to M
b = 2; % ratio of gravity to length l
A = [[0 1 0 0]; [0 -a 0 0]; [0 0 0 1]; [-b 0 b 0]];

[V,D,W] = eig(A)

syms l g d M
A_p = [[0 1 0 0]; [0 -d/M 0 0]; [0 0 0 1]; [-g/l 0 g/l 0]]
C_p = [[0 0 1/l 0];[1 0 1 0]]
Omega_o = [C_p;C_p*A_p;C_p*A_p*A_p;C_p*A_p*A_p*A_p]
rank(Omega_o)
```

```
V =

         0         0    0.7071   -0.6863
         0         0         0    0.1373
    0.5774   -0.5774    0.7071   -0.7003
    0.8165    0.8165   -0.0000    0.1401


D =

    1.4142         0         0         0
         0   -1.4142         0         0
         0         0         0         0
         0         0         0   -0.2000


W =

   -0.5889    0.5609    0.1961         0
   -0.3648   -0.4620    0.9806    1.0000
    0.5889   -0.5609         0         0
    0.4164    0.3966         0         0


A_p =

[   0,    1,   0, 0]
[   0, -d/M,   0, 0]
[   0,    0,   0, 1]
[-g/l,    0, g/l, 0]


C_p =

[0, 0, 1/l, 0]
[1, 0,   1, 0]


Omega_o =

[     0,              0,   1/l,     0]
[     1,              0,     1,     0]
[     0,              0,     0,   1/l]
[     0,              1,     0,     1]
[-g/l^2,              0, g/l^2,     0]
[  -g/l,           -d/M,   g/l,     0]
[     0,         -g/l^2,     0, g/l^2]
[     0, d^2/M^2 - g/l,      0,   g/l]


ans =

     4
```

Published with MATLAB® R2023a