Lab Session 01
Introduction to MATLAB

Exercise 1

Consider the two polynomials $p(s) = s^2 + 2s + 1$ and $q(s) = s + 1$. Use MATLAB to compute

a) $p(s) \times q(s)$

b) Roots of $p(s)$ and $q(s)$

c) $p(-1) and\ q(6)$

Exercise 1, Code 01

```
%%  FCS Lab 01 - Exercise 01
%   Saad Mashkoor Siddiqui, EE-16163, Section D, TE-EE 16-17 Spring '19
%%  Defining polynomials
p_s = [1, 2, 1];            % p(s) = s^2 + 2s + 1
q_s = [1, 1];               % q(s) = s + 1

%% PART A - Compute Product
fprintf('The coefficients of the product p(s)*q(s) are')
prod_pq = conv(p_s, q_s)

%% PART B - Compute roots of p(s) and q(s)
fprintf('The roots of polynomial p(s) are');
roots_p = roots(p_s)

fprintf('The roots of polynomial q(s) are')
roots_q = roots(q_s)

%% PART C - Evaluate p(-1) and q(6)
fprintf('The value of p(-1) is' )
polyval(p_s, -1)

fprintf('The value of q(6) is')
polyval(q_s, 6)
```

Exercise 01, Code 01 - Output

```
fcs_01_exercise_01
The coefficients of the product p(s)*q(s) are
```

```
prod_pq = 1     3     3     1

The roots of polynomial p(s) are
roots_p = -1   -1

The roots of polynomial q(s) are
roots_q = -1

The value of p(-1) is
ans = 0

The value of q(6) is
ans = 7
```

Based on the program's results

a) $p(s) \times q(s) = s^3 + 3s^2 + 3s + 1$

b) $p(s) = (s+1)^2 \Rightarrow p(-1) = 0$ and $q(s) = s + 1 \Rightarrow q(-1) = 0$

c) $p(-1) = 0$ and $q(6) = 7$

Exercise 02

Use MATLAB commands to find the partial fractions of the following

a) $\frac{2s^3+5s^2+3s+6}{s^3+6s^2+11s+6}$

b) $\frac{s^2+2s+3}{(s+1)(s+1)}$

Exercise 02, Code 01

```
%% %%  FCS Lab 01 - Exercise 02
%   Saad Mashkoor Siddiqui, EE-16163, Section D, TE-EE 16-17 Spring '19

%% Defining expressions for partial fraction decomposition
numer_A = [2, 5, 3, 6];             % 2s^3 + 5s^2 + 3s + 6
denom_A = [1, 6, 11, 6];            % s^3 + 6s^2 + 11s + 6

numer_B = [1, 2, 3];                % s^2 + 2s + 3
denom_B = conv([1, 1], [1, 1]);     % (s + 1)^2

%% PART A
```

```
[res_A, poles_A, const_A] = residue(numer_A, denom_A);

%% PART B
[res_B, poles_B, const_B] = residue(numer_B, denom_B);
```

Exercise 02, Code 01 - Output

```
fcs_01_exercise_02

res_A = -6.0000 -4.0000 3.0000

poles_A = -3.0000 -2.0000 -1.0000

const_A = 2

res_B = 0 2

poles_B = -1 -1

const_B = 1
```

Based on the program's results

a) $p(s) = 2 + \frac{-6}{s+3} + \frac{-4}{s+2} + \frac{3}{s+1}$

b) $q(s) = 1 + \frac{2}{(s+1)^2}$

Exercise 03

Use MATLAB commands to obtain the following

a) Extract the fourth row of the matrix generated by `magic(6)`

b) Given `x = [0 : 0.1 : 1.1]` and `y = [10 : 21]`, show the results of '`x`' multiplied by '`y`' and of '`y`' divided by '`x`'.

c) Generate a random matrix '`r`' of size 4 by 5 with numbers varying between -8 and 0.

Exercise 03, Code 01

```
%% %%  FCS Lab 01 - Exercise 03
%   Saad Mashkoor Siddiqui, EE-16163, Section D, TE-EE 16-17 Spring '19

%% Part A - Extracting fourth row of magic(6)
```

```
mag_6 = magic(6) fourth_row = mag_6(4,:)

%% PART B - Show the results of x * y and y / x
x = [0 : 0.1 : 1.1 ]; y = [ 10 : 21 ];
element_wise_prod = x .* y
matrix_prod = x * y'    % transposing so matrix dimensions agree

element_wise_quotient = y ./ x
matrix_quotient = y / x

%% PART C - Generating a 4 x 5 random matrix with numbers betweenn [-8, 0]
lower = -8; upper = 0;
random_mat = (upper - lower ) * rand(4, 5) + lower
```

Exercise 02, Code 01 - Output

```
fcs_01_exercise_03
mag_6 =
    35     1     6    26    19    24
     3    32     7    21    23    25
    31     9     2    22    27    20
     8    28    33    17    10    15
    30     5    34    12    14    16
     4    36    29    13    18    11
fourth_row = 8    28    33    17    10    15

element_wise_prod = Columns 1 through 7
         0    1.1000    2.4000    3.9000    5.6000    7.5000    9.6000
  Columns 8 through  12
        11.9000   14.4000   17.1000   20.0000   23.1000

matrix_prod = 116.6000

element_wise_quotient = Columns 1 through 7
       Inf  110.0000   60.0000   43.3333   35.0000   30.0000   26.6667
  Columns 8 through 12
   24.2857   22.5000   21.1111   20.0000   19.0909

matrix_quotient = 23.0435
random_mat =
   -0.1387   -7.1378   -5.9142   -5.4982   -7.2462
   -2.1580   -0.7495   -3.2451   -6.7081   -3.2118
   -5.2490   -0.9628   -7.8199   -6.5699   -4.2326
   -3.3274   -1.4579   -4.5979   -4.6169   -2.4324
```