Construct a random matrix K of order 3×4, reverse the order of the rows of K, reverse the order of the columns of K and then perform both operations simultaneously. Find the matrix L of order 4×3 whose columns are obtained by taking the elements of K sequentially by columns.

question1.2

SOLUTION1.2

```
>> K = rand(3,4)

K =

0.5269    0.4160    0.7622    0.7361
0.0920    0.7012    0.2625    0.3282
0.6539    0.9103    0.0475    0.6326

>> K(3:-1:1,:)

ans =

0.6539    0.9103    0.0475    0.6326
0.0920    0.7012    0.2625    0.3282
0.5269    0.4160    0.7622    0.7361
```

SOLUTION1.2

```
>> K(:,4:-1:1)

ans =

0.7361    0.7622    0.4160    0.5269
0.3282    0.2625    0.7012    0.0920
0.6326    0.0475    0.9103    0.6539

>> K(3:-1:1,4:-1:1)

ans =

0.6326    0.0475    0.9103    0.6539
0.3282    0.2625    0.7012    0.0920
0.7361    0.7622    0.4160    0.5269
```

SOLUTION1.2

```
>> L = reshape(K,4,3)

L =

0.5269 0.7012 0.0475
0.0920 0.9103 0.7361
0.6539 0.7622 0.3282
0.4160 0.2625 0.6326
```

ARITHMETIC OPERATORS

Operator	Role played
+	*Sum of scalars, vectors, or matrices*
-	*Subtraction of scalars, vectors, or matrices*
*	*Product of scalars or arrays*
.*	*Product of scalars or vectors*
\	$A \backslash B = inv\,(A) * B$, *where A and B are matrices*
.\	$A. \backslash B = [B(i,j)\, /A\,(i, j)]$, *where A and B are vectors* $[dim\,(A) = dim\,(B)]$
/	*Quotient, or* $B/A = B * inv\,(A)$, *where A and B are matrices*
./	$A\, /\, B = [A(i,j)/b\,(i, j)]$, *where A and B are vectors* $[dim\,(A) = dim\,(B)]$
^	*Power of a scalar or matrix* (M_{p})
.^	*Power of vectors* $(A.\,\hat{}\,B = [A(i,j)^{B\,(i,\,j)}]$, *for vectors A and B*$)$

TASK#1

- Generate 3 by 3 square matrix named “A” and “B” with random values of integers using “randi” function.
- Make all the arithmetic operations shown on the table
- A*B = B*A or not ?
- A.*B = A*B or not ? What is the different ?
- Determinant of A = ?
- Inverse of B = ? Which matrix is invertible ?

PRIZE QUESTION ?

- Solve the equations shown below using MATLAB :

$$3x_1 + 2x_2 - x_3 = 10$$

$$-x_1 + 3x_2 + 2x_3 = 5$$

$$x_1 - x_2 - x_3 = -1$$

SOLVING LINEAR EQUATIONS

We can use basic matrix operations to solve a linear systems in a few steps:

Example: A system of 3 linear equations with 3 unknowns (x_1, x_2, x_3):

$$3x_1 + 2x_2 - x_3 = 10$$

$$-x_1 + 3x_2 + 2x_3 = 5$$

$$x_1 - x_2 - x_3 = -1$$

We can write these equation systems in $Ax=b$ form.

$$A = \begin{bmatrix} 3 & 2 & 1 \\ -1 & 3 & 2 \\ 1 & -1 & -1 \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \qquad b = \begin{bmatrix} 10 \\ 5 \\ -1 \end{bmatrix}$$

SOLVING LINEAR EQUATIONS

As you remember from linear algebra courses, the solution of the system is:

$$Ax = b$$

$$A^{-1}Ax = A^{-1}b$$

$$x = A^{-1}b$$

```
>> A = [3 2 -1; -1 3 2; 1 -1 -1];

>> b = [10; 5; -1];

>> x = inv(A)*b

x =

-2.0000

5.0000

-6.0000
```

Create a system of 5 linear equations with 5 unknowns and solve the system using MATLAB.

homework2.1

FIGURES

The objective of this exercise is to understand figure generation in MATLAB.

1. There are many useful figure properties in MATLAB. Common figure commands are; `plot, stem, bar plot3, bar3 area, scatter`...

If you use the `plot` command, MATLAB will open new Figure window automatically. However it is absolutely recommended to run `figure;` command just before a plot command. Example:

```
x=0:0.1:50; % We generated the vector with 1x501 size
y=sin(x); % We created a sinus signal from the variable
figure; % A figure window opened
plot(y); % The graph of the y is plotted.
```

Plot command, draws the values of the vector against its size. If you use `plot(y)` MATLAB plots values of the vector y to the Y-axis and it automatically fills the X-axis with size of the y. Hence `plot(y)` equals to `plot(y,x)`

```
figure; % A new figure window opened
plot(x,y); % The graph of the y is plotted.
```

Remember that, the size of the x and y must be equal! Otherwise you will get error message.

FIGURES

2. We can change the color, size, shape of the line with additional options. Example:

```
figure;
plot(x,y,'LineStyle','--'); % Line style changed
figure;
plot(x,y,'Color','Red'); % Line color changed
```

We can combine options together:

```
figure;
plot(x,y,'LineStyle','--','Color','Red');
```

The values on the figure are discrete but they look like continuous because MATLAB automatically interpolates between values. If you want to emphasize real values you can use markers.

```
figure;
plot(x,y,'Marker','X');
```

FIGURES

3. You can add a title to your figure with `title` command just after the `plot` command.

```
figure;
plot(x,y);
title('This is a figure.')
```

4. You can combine your plots into one figure with `subplot(m,n,p)` command. This command generates m-by-n sub windows. p value is the number of the graphic. Example:

```
figure;
subplot(3,1,1);
plot(x,y,'Color','Red'); %First plot
title('First figure')
subplot(3,1,2);
plot(x,y,'LineStyle','--'); %Second plot
title('Second figure')
subplot(3,1,3);
plot(x,y,'Marker','X'); %Third plot
title('Third figure')
```

I. Open the document of plot function. At the “Input Arguments” part, look the “LineSpec — Line style, marker symbol, and color” section.

II. Generate three different plot with different Color, marker and Line style properties. Save them as PNG file for your report.

III. Merge your plots into a sub plot with size of 2x2. Give a title to each plot. Save the figure

homework2.2

I. Generate a 1-by-n size vector with step size of 0.1.

II. Determine the size randomly with `100+randi(50)` command. This command will generate an integer between 50 and 150.

III. Generate a vector from `sin` function with the variable of the vector that you generated in step I

IV. Plot the vector that you generated in step III with “magenta” color, “dotted line” and “cross marker”

V. Plot the vector again in a new figure but only plot the values between 15 and 80

homework2.3

FIGURES

5. In MATLAB it is possible to plot multiple values on the same figure with `hold on` command. It is very useful if you want to compare two signals. Example:

```
x=0:0.1:50;
y1=sin(x);
y2=cos(x);
figure;
plot(x,y1,'Color','Black');
hold on % Keep the same figure
plot(x,y2,'Color','Red');
```

FIGURES

Three dimensional figures are also supported by MATLAB

```
x=randi(20,1,50); % Generate 1x50 random vector maximum value of 20
y=randi(20,1,50); % Generate 1x50 random vector maximum value of 20
z=randi(20,1,50); % Generate 1x50 random vector maximum value of 20
figure;
plot3(x,y,z); %3D plot of x,y,z vectors
```

POLYNOMIALS

$$p(x) = x^3 - 2x - 5$$

To enter this polynomial into MATLAB, use

```
>>p = [1 0 -2 -5];
```

The roots function calculates the roots of a polynomial:

```
>>r = roots(p)

r =

   2.0946 + 0.0000i

  -1.0473 + 1.1359i

  -1.0473 - 1.1359i
```

By convention, MATLAB stores roots in column vectors. The function `poly` returns to the polynomial coefficients:

```
>> p2 = poly(r)
```

POLYNOMIALS

Convolution and Deconvolution

Polynomial multiplication and division correspond to the operations convolution and deconvolution. The functions `conv` and `deconv` implement these operations. Consider the polynomials below:

$$a(s) = s^2 + 2s + 3$$

$$b(s) = 4s^2 + 5s + 6$$

To compute their product,

```
>>a = [1 2 3];

>>b = [4 5 6];

>>c = conv(a,b)

c =

4 13 28 27 18
```

POLYNOMIALS

Use deconvolution to divide back out of the product:

```
>>[q,r] = deconv(c,a)

q =

4 5 6

r =

0 0 0 0 0
```

POLYNOMIALS

Partial Fraction Expansion

`residue` function finds the partial fraction expansion of the ratio of two polynomials. This is particularly useful for applications that represent systems in transfer function form. For polynomials b and a;

$$\frac{b(s)}{a(s)} = \frac{r_1}{s - p_1} + \frac{r_2}{s - p_2} + \cdots + \frac{r_n}{s - p_n} + k_s$$

For example consider the transfer function b(s)/a(s);

$$b(s) = -4s + 8$$

$$a(s) = s^2 + 6s + 8$$

POLYNOMIALS

```
b=[-4 8];
a=[1 6 8];
[r,p,k] = residue(b,a)
r =

   -12

     8

p =

    -4

    -2

k =

     []
```

POLYNOMIALS

Which means that;

$$\frac{b(s)}{a(s)} = \frac{-4s+8}{s^2+6s+8} = \frac{\overbrace{-12}^{r(1)}}{s-(\underbrace{-4}_{p(1)})} + \frac{\overbrace{8}^{r(2)}}{s-(\underbrace{-2}_{p(2)})} + \overbrace{0}^{k}$$

The reverse of the residue function is also available. Given three input arguments (r, p, and k), residue converts back to polynomial form:

```
>>[b2,a2] = residue(r,p,k)

b2 = -4 8

a2 = 1 6 8
```

Use MATLAB command to find the partial fraction of the following;

a. $$\frac{B(s)}{A(s)} = \frac{2s^3+5s^2+3s+6}{s^3+6s^2+11s+6}$$

b. $$\frac{B(s)}{A(s)} = \frac{s^2+2s+3}{(s+1)^3}$$

homework2.4