Chapter Five
Built−in Functions

Matlab contains a large library of mathematical and logical functions that can be classified as follows:

a) General arithmetic functions.
b) Logarithmic and power functions.
c) Trigonometric functions.
d) Hyperbolic functions.
e) Round−off functions.
f) Sorting functions.

5.1 General arithmetic functions

This category involve functions to do simple, yet, repetitive and tedious tasks. They include evaluating average, sum and product values of groups of numbers (matrices).

Average value: it is accomplished using the function (mean):

```
x= [ 3  4  1  2  5 ];          x = [ 2  5   8 ; 1  3  4 ] ;
y = mean( x ) ↵                y = mean( x ) ↵
y = 3                          y = 1.5    4    6
```

Summation of numbers: the sum of a group of numbers is evaluated using (sum):

```
x= [ 3  4  1  2  5 ];          x = [ 2  5   8 ; 1  3  4 ] ;
y = sum( x ) ↵                 y = sum( x ) ↵
y = 15                         y = 3    8    12
```

Cumulative summation: this process generates a matrix having a cumulative sum of a group of numbers using the function (cumsum):

```
x = [ 2  5  8  1  3  4 ] ;              x = [ 2  5  8 ; 1  3  4 ] ;
y = cumsum( x ) ↵                       y = cumsum( x ) ↵
y = 2  7  15  16  19  23                 y = 2   5   8
                                             3   8   12
```

Product of numbers: to evaluate the result of multiplying the elements of a group of numbers, the function (prod) is used:

```
x = [ 2  5  8  1  3  4 ] ;
y = prod( x ) ↵
y = 960
```

```
x = [ 2  5  8 ; 1  3  4 ] ;
y = prod( x ) ↵
y = 2   15   32
```

Cumulative product: this process generates a matrix of a cumulative multiplying values using the function (cumprod):

```
x = [ 2  5  8  1  3  4 ] ;
y = cumprod( x ) ↵
y = 2  10  80  80  240  960
```

```
x = [ 2  5  8 ; 1  3  4 ] ;
y = cumprod( x ) ↵
y = 2    5    8
    2   15   32
```

Ex. (5.1): Write Matlab program to display a table of three columns where the first column is the months of the year (numbers from 1 to 12), the second column is the sales of a car company (number of cars sold each month) and the third column is the profits of the company each month. Calculate also the average number of cars sold each month and the total profit of the year.

```
clear,clc
Month=1:12;
No_cars=[3 5 10 6 2 8 6 4 5 3 8 10];
Profits=[12 15 45 30 22 44 33 22 25 40 30 20];
Table=[Month' No_cars' Profits'];
disp(Table)
disp(' ')
avg=mean(No_cars);
tot_Prof=sum(Profits);
disp([avg,tot_Prof])
```

5.2 Logarithmic and power functions

This group includes the functions used to raise to a power or evaluating natural and base−ten logarithms.

Square root: to find the square root of positive numbers the function (sqrt) is used:

```
y = sqrt( 4 )        y = sqrt( [ 4 , 9 , 16 ] )        y = sqrt( [ 9 , 4 , 100 ; 81 , 49 , 36 ] )
y = 2                y =  2   3   4                    y =  3    2    10
                                                            9    7    6
```

General root: to evaluate the nth root of a number the function (nthroot) is used:

```
y = nthroot( 8 , 3 )   y = nthroot( [ 8 , 27 , 64 ] , 3 )   y = nthroot( [ 243 , 125 ] , [ 5 , 3 ] )
y = 2                  y =    2      3      4                y =    3      5
```

Exponential: it is the number that is the exponent to (e = 2.7183). It is obtained by the function (exp):

$y = e^{2}$ ⟶ y = exp(2) ⟶ y = 7.3891

Natural logarithm: which is the logarithm to the base (e) and is evaluated by the function (log):

y = ln(2) ⟶ y = log(2) ⟶ y = 0.6931

Base−10 logarithm: which is the logarithm to the base (10) and is evaluated by the function (log10):

$y = \log_{10}(100)$ ⟶ y = log10(100) ⟶ y = 2

Note: to find (y) that is the logarithm of the number (x) to any base (z) the following formula can be used: y = ln(x) / ln (z)

e.g.: $y = \log_{3}(4)$ ⟶ y = log(4) / log(3) ⟶ y = 1.2619

Ex. 5.2 Write Matlab code to evaluate the following function for 5 values of x equally spaced between 4.5 and 8.5:

$$y = \frac{(ln(x) + 2)^{1.5}}{\sqrt{log_{10}(x) + 4}} + log_2(x^{2.5} + 1)$$

```
4.5000    8.4988
5.5000    9.4439
6.5000   10.2371
7.5000   10.9201
8.5000   11.5196
```

Sol.

```
clear , clc
x = linspace( 4.5 , 8.5 , 5 ) ;
A = ( log(x) + 2 ) . ^ 1.5 ; B = sqrt( log10( x ) + 4) ;
C = log( x . ^ 2.5 + 1 ) . / log( 2 ) ; y = A . / B + C ; disp( [ x ; y ] ' )
```