MATLAB

ARRAY

Session 4

MS.c Haneen ALhariri

MS.c Ola Ali

ARRAYS AND MATRICES

An array is MATLAB's basic data structure

- Can have any number of dimensions. Most common are:

*vector: one dimension (a single row or column)

*matrix: two or more dimensions

*Scalar: matrices with only one row and one column.

-Arrays can have numbers or letters

Creating Matrices

In MATLAB, a vector is created by assigning the elements of the vector to a variable.

Row vector: In a row vector, the elements are entered with a space or a coma between the elements inside the square brackets X=[1 2 3]

x=[1, 2, 3] or x=[1 2 3]

Column vector: In a column vector, the elements are entered with a semicolon between the elements inside the square brackets.

X=[1;2;3]

$$x = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

A matrix can be created in MATLAB by typing the elements (numbers) inside square brackets[ ]

>> matrix [1 2 3 ;4 5 6 ;7 8 9]

Examples

```
>> A = [2 -3  5; -1  4  5]
A =
     2  -3  5
    -1  4   5
>> x = [1  4  7]
X=
    1  4  7
>> x = [1; 4; 7]
X=
1
4
7
```

Built-in Functions to Generate Matrices

zeros (r, c): makes matrix of r rows and c columns, all with zeros

ones (r, c): makes matrix of r rows and ccolumns, all with ones

rand (r, c): makes matrix of r rows and c columns, with random numbers

eye (n): makes square matrix of n rows and columns. Main diagonal (upper left to lower right) has ones, all other elements are zero magic (n) - makes a special square matrix of n rows and c columns, called Durer's matrix

Examples

```
>> a=zeros (3,4)
a = 0 0 0 0
    0 0 0 0
    0 0 0 0
```

```
>> B=ones (4,3)
B=
     1 1 1
     1 1 1
     1 1 1
     1 1 1
>> c = rand (2,3)
C=

0.8147   0.1270  0.6324
d-eye (4)

d = 1 0 0 0
      0 1 0 0
      0 0 1 0
      0 0 0 1
e=magic(4)
e=16 2 3 13
    5 11 10 8
    9  7 6 12
    4 14 15 1
```

*To make a matrix filled with a particular number, multiply ones (m, n) by tha

```
>> z=100*ones (3, 4)
  Z=
      100   100   100   100
      100   100   100   100
      100   100   100   100
      100   100   100   100
```

Using a Colon : in Addressing Arrays

The colon:lets you address a range of elements

- Vector (row or column)
  - X(:) – all elements
  - X(m:n)- elements m through n
- Matrix
  - A(:,n) - all rows of column n
  - A(m,:) – all columns of row m
  - A(:,m:n) – all rows of columns m through n
  - A(m:n,:) – all columns of rows m through n
  - A(m:n,p:q) - columns p through q of rows m through n

```
A=[2,4,10,13;16,3,7,18;8,4,9,25;3,12,15,17]
A=
     2    4    10   13
     16   3    7    18
     8    4    9    25
     3    12   15   17
```

A(:,3)=

ANS =

10

7

9

15

A(3,:)

ANS=

8 4 9 25

A(:,2:3)

ANS=

4 10

3 7

4 9

12 15

A(2:3,1:3)

ANS=

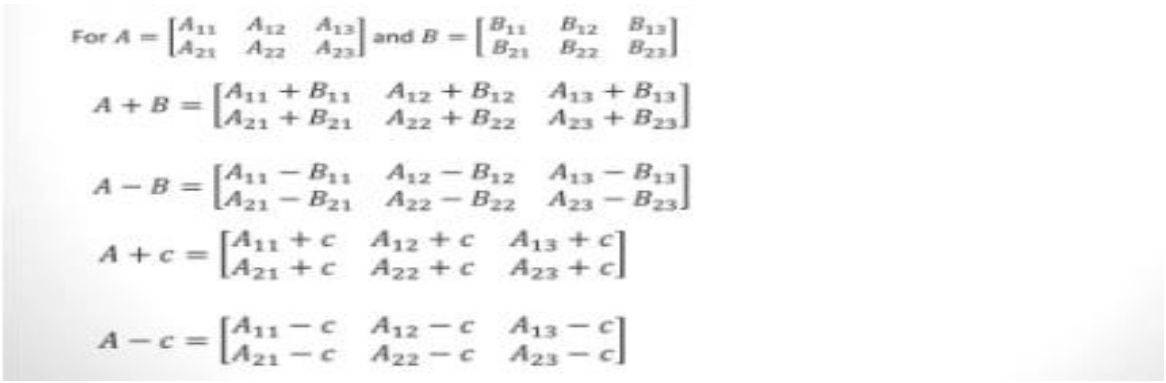
16 3 7

8 4 9

Addition and Subtraction

When adding/ subtracting two arrays A and B, MATLAB adds/subtracts the corresponding elements (element wise addition/subtraction)

When add/subtract a scalar to an array, MATLAB adds/ subtracts the scalar to every element of the array EXAMPLE



Matrix Multiplication

There are two ways of multiplying matrices - matrix multiplication and elementwise multiplication

• MATLAB denotes this with asterisk (*)

• Number of columns in left matrix must be same as number of rows in right matrix

A=

3 3 1

3 2 2

1 1 3

B=

3 3 1

1 2 2

3 3 3

>>C=A*B

C= 15 18 12

17 19 13

13 14 12

>>D=B*A

D= 19 16 12

11 9 11

21 18 18

$$A = \begin{matrix} 2 & 0 & 3 \\ 4 & 5 & 6 \\ 7 & 1 & 9 \end{matrix} \qquad B = \begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix}$$

```
A=[2,0,3;4,5,6;7,1,9]
B=[1,2,3;4,5,6;7,8,9]
Mat=A.*B
Mat1=A.^2
```

A=

2	0	3
4	5	6
7	1	9

B=

1	2	3
4	5	6
7	8	9

Mat=

2	0	9
16	25	36
49	8	81

Mat1=

4	0	9
16	25	36
49	1	81