



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

INTRODUCTION TO SCIENTIFIC & ENGINEERING COMPUTING BIL 108E, CRN24023

Dr. S. Gökhan Karaman

Technical University of Istanbul

March 15, 2010



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

Tentative Course Schedule, CRN 24023

Week	Date	Topics
1	Feb. 08	Introduction to Scientific and Engineering Computing
2	Feb. 15	Introduction to Program Computing Environment
3	Feb. 22	Variables, Operations and Simple Plot
4	Mar. 01	Algorithms and Logic Operators
5	Mar. 08	Flow Control, Errors and Source of Errors
6	Mar. 15	Functions
6	Mar. 20	Exam 1
7	Mar. 22	Arrays
8	Mar. 29	Solving of Simple Equations
9	Apr. 05	Polynomials Examples
10	Apr. 12	Applications of Curve Fitting
11	Apr. 19	Applications of Interpolation
11	Apr. 24	Exam 2
12	Apr. 26	Applications of Numerical Integration
13	May 03	Symbolic Mathematics
14	May 10	Ordinary Differential Equation (ODE) Solutions with Built-in Functions



LECTURE # 6

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

LECTURE # 6 LINEAR ALGEBRA

- 1 INNER PRODUCT OF VECTORS
- 2 OUTER PRODUCT OF VECTORS
- 3 MATRIX DEFINITION
 - MATRIX OPERATIONS
 - SQUARE MATRIX
 - TRANSPOSE OF A MATRIX
 - SYMMETRIC MATRIX
 - IDENTITY MATRIX
 - INVERSE MATRIX
 - EXAMPLES
- 4 LINEAR EQUATIONS
- 5 SOLUTIONS
- 6 EXAMPLES



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

VECTORS

VECTORS

- A vector is an ordered list of numbers (one-dimensional). In MATLAB they can be represented as a row-vector or a column-vector ($1 \times n$) or $(n \times 1)$.



VECTOR INNER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

VECTOR INNER PRODUCT

- In physics, analytical geometry, and engineering, the dot product has a geometric interpretation.

$$a \cdot b = a_1 b_1 + a_2 b_2 + \dots + a_n b_n$$

$$c = \text{dot}(a, b)$$
- Returns the scalar product of the vectors a and b.
- a and b must be vectors of the same length.
- When a and b are both column vectors, $\text{dot}(a,b)$ is the same as $a * b'$.



VECTOR INNER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_01.m

```
a = [1 2 3];
b = [5 8 13];
c = dot(a,b)
a * b'
a' * b
```

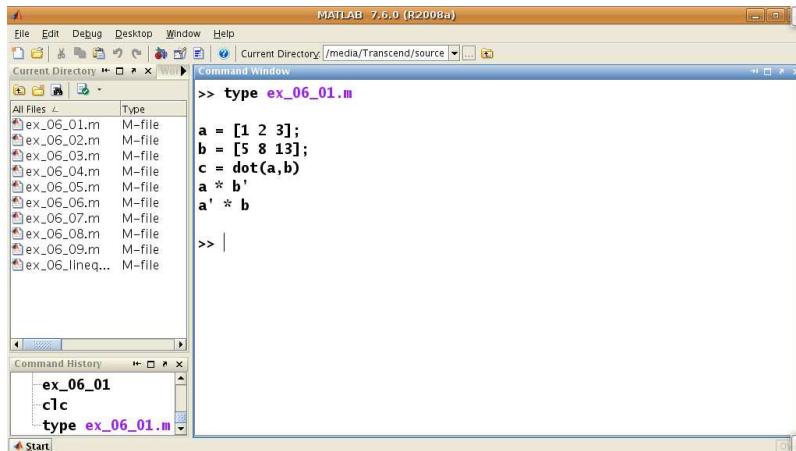


VECTOR INNER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

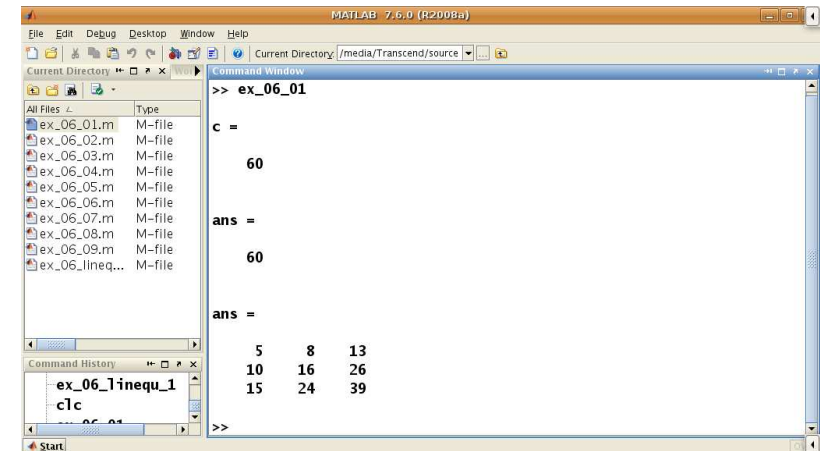


VECTOR INNER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:





VECTOR INNER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

VECTOR INNER PRODUCT

- The rules of linear algebra impose compatibility requirements on the inner product.
- The inner product of x and y requires that x be a row vector y be a column vector.

$$a = [x_1 \quad x_2 \quad \dots \quad x_n] \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{bmatrix} =$$

$$= x_1y_1 + x_2y_2 + \dots x_ny_n$$



VECTOR INNER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

VECTOR INNER PRODUCT

- The $*$ operator performs the inner product if two vectors are compatible.
- The inner product result is a scalar.



VECTOR OUTER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

VECTOR OUTER PRODUCT

- The outer product creates a matrix.
- $A = uv^T$
- $a(i,j) = a_{ij} = u(i)v(j)$

$$A = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix} [v_1 \quad v_2 \quad \dots \quad v_n]$$

$$= \begin{bmatrix} u_1v_1 & u_1v_2 & \dots & u_1v_n \\ u_2v_1 & u_2v_2 & \dots & u_2v_n \\ \vdots & \vdots & \vdots & \vdots \\ u_mv_1 & u_mv_2 & \dots & u_mv_n \end{bmatrix}$$



VECTOR OUTER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_02.m

```
u = (0:3); % u and v are
v = (3: -1:0); % column vectors
s = u' * v
t = v' * u
```



VECTOR OUTER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_06_02.m
u = (0:3); % u and v are
v = (3: -1:0); % column vectors
s = u' * v
t = v' * u
>> ex_06_02
s =
    0    0    0    0
    3    2    1    0
    6    4    2    0
    9    6    3    0
t =
    0    3    6    9
    0    2    4    6
    0    1    2    3
    0    0    0    0
Command History
c1c
type ex_06_02.m
ex_06_02
Start

```



VECTOR OUTER PRODUCT

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> ex_06_02
s =
    0    0    0    0
    3    2    1    0
    6    4    2    0
    9    6    3    0
t =
    0    3    6    9
    0    2    4    6
    0    1    2    3
    0    0    0    0
Command History
c1c
type ex_06_02.m
ex_06_02
>>
Start

```



MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRICES

- Columns and Rows of a Matrix are Vectors
- Addition and Subtraction
- Multiplication by a scalar
- Transpose
- Linear Combinations of Vectors
- Matrix – Vector Product
- Matrix – Matrix Product



MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX OPERATIONS

- Addition and subtraction

$$C = A + B$$

$$c(i,j) = a(i,j) + b(i,j)$$

$$i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n$$
- Multiplication by a scalar

$$B = \lambda A$$

$$b(i,j) = \lambda a(i,j) \quad i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n$$



MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_03.m

```
% ex_06_03.m
A= magic(3)
B= [1 2 3; 5 8 13; 21 34 55]
C = A + B
C = B + A
C = A - B
B = [1 2 3; 5 8 13]
C = A + B
```

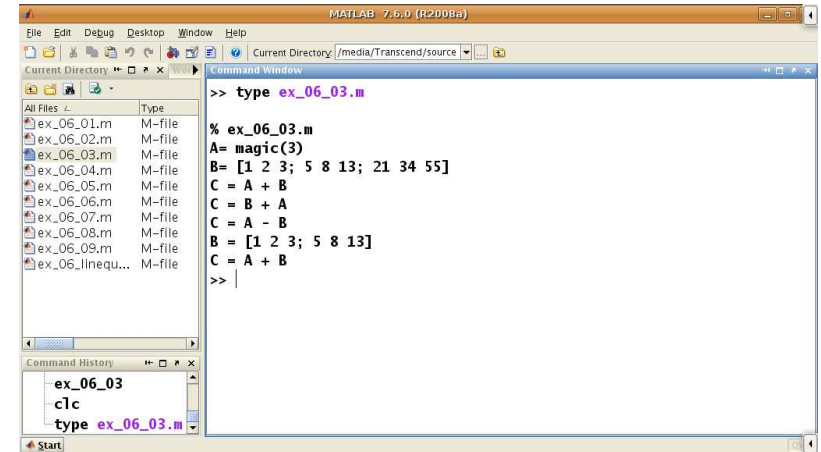


MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

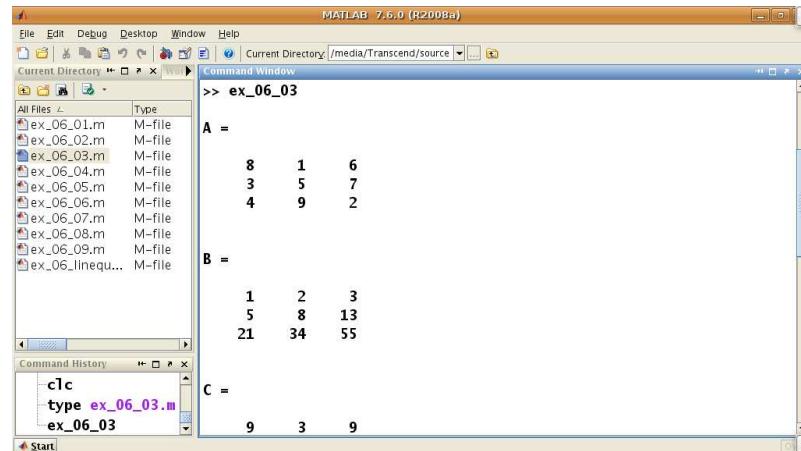


MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

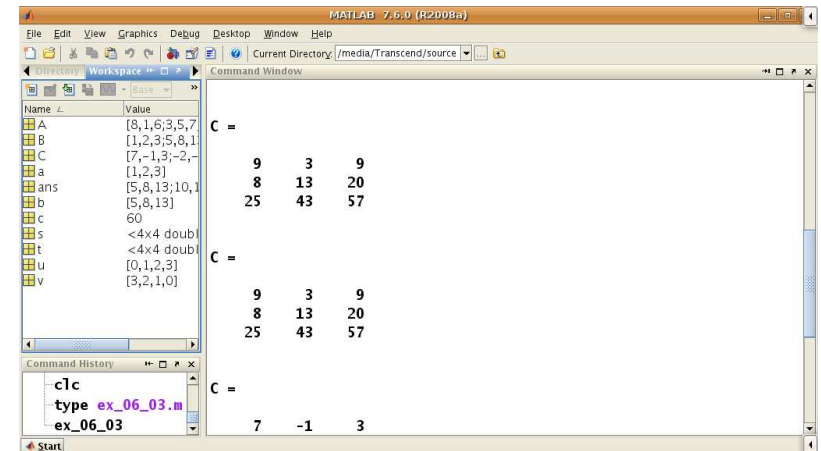


MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:





MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit View Graphics Debug Desktop Window Help
Current Directory: /media/Transcend/source
Workspace: /media/Transcend/source
Name Value
A [8,1,6;3,5,7]
B [1,2,3;5,8,1]
C [7,-1,3;-2,-17,-25]
a [1,2,3]
ans [5,8,13;10,1]
b [5,8,13]
c 60
s <4x4 double>
t <4x4 double>
u [0,1,2,3]
v [3,2,1,0]
Command History
clc
type ex_06_03.m
ex_06_03
C =
     9     3     9
     8    13    20
    25    43    57
  
```



MATRIX OPERATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit View Graphics Debug Desktop Window Help
Current Directory: /media/Transcend/source
Workspace: /media/Transcend/source
Name Value
A [8,1,6;3,5,7]
B [1,2,3;5,8,1]
C [7,-1,3;-2,-3,-6]
a [1,2,3]
ans [5,8,13;10,1]
b [5,8,13]
c 60
s <4x4 double>
t <4x4 double>
u [0,1,2,3]
v [3,2,1,0]
Command History
clc
type ex_06_03.m
ex_06_03
C =
     7     -1     3
    -2     -3     -6
   -17   -25   -53
B =
     1     2     3
     5     8    13
??? Error using ==> plus
Matrix dimensions must agree.
Error in ==> ex_06_03 at 8
C = A + B
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SQUARE MATRIX

- If the number of rows are equal to number of columns, than the matrix is a square matrix.

EXAMPLE:

filename: ex_06_04.m

```

% SQUARE MATRIX EXAMPLE
A = magic(4)
dimA = size(A)
dimA_row = dimA(1)
dimA_column = dimA(2)
A'
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Workspace: /media/Transcend/source
All Files
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_lineq... M-file
Command Window
>> type ex_06_04.m
% SQUARE MATRIX EXAMPLE
A = magic(4)
dimA = size(A)
dimA_row = dimA(1)
dimA_column = dimA(2)
A'
>> |
Command History
type ex_06_01.m
clc
type ex_06_04.m
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> ex_06_04
A =
    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1

dimA =
     4     4

dimA_row =
     4
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
dimA_row =
     4

dimA_column =
     4

ans =
    16     5     9     4
     2    11     7    14
     3    10     6    15
    13     8    12     1

c1c
type ex_06_04.m
ex_06_04
>>
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
     4     4

dimA_row =
     4

dimA_column =
     4

>> A(1, :)
ans =
    16     2     3    13

>>
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
dimA_column =
     4

>> A(1, :)
ans =
    16     2     3    13

>> sum(A(1, :))
ans =
    34

ex_06_04
A(1, :)
sum(A(1, :))
>>
  
```



SQUARE MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z:
Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> A(1, :)
ans =
    16     2     3    13

>> sum(A(1, :))
ans =
    34

>> sum(A(:, 2))
ans =
    34

Command History
A(1, :)
sum(A(1, :))
sum(A(:, 2))
>>
  
```



MATRIX TRANSPOSE

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX TRANSPOSE

$$B = A^T$$

$$b(i, j) = a(j, i)$$

$$i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n$$



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX MULTIPLICATION

$$c_{ij} = \sum_{k=1}^p a_{ik} b_{kj}$$

$$i = 1, 2, \dots, m, j = 1, 2, \dots, n$$



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_05.m

% MATRIX MULTIPLICATION EXAMPLE

A = [1 2 1; 0 1 2; 0 0 2]

B = [1 2 0; -1 1 2; 0 1 2]

C = A * B

C = B * A

B = [1 2 0; -1 1 2]

C = A * B

A = [1 2 1]

B = [1 2 0; -1 1 2; 0 1 2]

C = A * B

A = [1 2 1; 0 1 2; 0 0 2]

B = [1 2 0]

C = A * B



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory /media/Transcend/source
Command Window
All Files z Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

% MATRIX MULTIPLICATION EXAMPLE
A = [1 2 1; 0 1 2; 0 0 2]
B = [1 2 0; -1 1 2; 0 1 2]
C = A * B
C = B * A
B = [1 2 0; -1 1 2]
C = A * B
A = [1 2 1]
B = [1 2 0; -1 1 2; 0 1 2]
C = A * B
A = [1 2 1; 0 1 2; 0 0 2]
B = [1 2 0]
C = A * B
A = [1 2; 0 1; 0 2]
B = [1 2 0; -1 1 2]
C = A * B
C = B * A
>>

Command History
sum(A(:, 2))
clc
type ex_06_05.m
Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory /media/Transcend/source
Command Window
All Files z Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> A = [1 2 1; 0 1 2; 0 0 2]
A =
     1     2     1
     0     1     2
     0     0     2

>> B = [1 2 0; -1 1 2; 0 1 2]
B =
     1     2     0
    -1     1     2
     0     1     2

>> |
Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory /media/Transcend/source
Command Window
All Files z Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

0 0 2
>> B = [1 2 0; -1 1 2; 0 1 2]
B =
     1     2     0
    -1     1     2
     0     1     2

>> C = A * B
C =
    -1     5     6
    -1     3     6
     0     2     4

>>
Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory /media/Transcend/source
Command Window
All Files z Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> C = A * B
C =
    -1     5     6
    -1     3     6
     0     2     4

>> C = B * A
C =
     1     4     5
    -1    -1     5
     0     1     6

>>
Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
-1 3 6
0 2 4
>> C = B * A
C =
1 4 5
-1 -1 5
0 1 6
>> B = [1 2 0; -1 1 2]
B =
1 2 0
-1 1 2
Command History
C = A * B
C = B * A
B = [1 2 0; -1 1 2]
>>

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
C =
1 4 5
-1 -1 5
0 1 6
>> B = [1 2 0; -1 1 2]
B =
1 2 0
-1 1 2
>> C = A * B
??? Error using ==> mtimes
Inner matrix dimensions must agree.
Command History
B = [1 2 0; -1 1 2]
C = A * B
>>

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
-1 1 2
>> C = A * B
??? Error using ==> mtimes
Inner matrix dimensions must agree.
>> size(A)
ans =
3 3
>> size(B)
ans =
2 3
Command History
C = A * B
size(A)
>>

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

Command Window
2 3
>> A = [1 2 1]
A =
1 2 1
>> B = [1 2 0; -1 1 2; 0 1 2]
B =
1 2 0
-1 1 2
0 1 2
>>

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

Command Window

    1    2    1

>> B = [1 2 0; -1 1 2; 0 1 2]

B =

    1    2    0
   -1    1    2
    0    1    2

>> C = A * B

C =

   -1    5    6

>>

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)

File Edit Debug Desktop Window Help
Current Directory [media/Transcend/source]
Command Window
B =
    1    2    0
   -1    1    2
    0    1    2

>> C = A * B

C =

   -1    5    6

>> A = [1 2 1; 0 1 2; 0 0 2];
>> B = [1 2 0];
>> C = A * B
??? Error using ==> mtimes
Inner matrix dimensions must agree.

Command History
A = [1 2 1; 0 1 2; 0 0 2];
B = [1 2 0];
C = A * B

Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)

File Edit Debug Desktop Window Help
Current Directory [media/Transcend/source]
Command Window
>> C = A * B
??? Error using ==> mtimes
Inner matrix dimensions must agree.

>> A = [1 2; 0 1; 0 2];
>> B = [1 2 0; -1 1 2];
>> size(A)

ans =

    3    2

>> size(B)

ans =

    2    3

Command History
A = [1 2; 0 1; 0 2];
B = [1 2 0; -1 1 2];
size(A)
size(B)

Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)

File Edit Debug Desktop Window Help
Current Directory [media/Transcend/source]
Command Window
ans =

    3    2

>> size(B)

ans =

    2    3

>> C = A * B

C =

   -1    4    4
   -1    1    2
   -2    2    4

Command History
size(A)
size(B)

Start

```



MATRIX MULTIPLICATION

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z:
Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> C = A * B
C =
     2     3
    -1     4     4
    -1     1     2
    -2     2     4

>> C = B * A
C =
     1     4
    -1     3

Command History
size(B)
C = A * B
C = B * A
>>
  
```



DIAGONAL MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

DIAGONAL MATRICES

Diagonal matrices have non-zero elements only on the main diagonal.

$$C = \text{diag}(c_1, c_2, \dots, c_n)$$

$$C = \begin{bmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & c_n \end{bmatrix}$$



DIAGONAL MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_06.m

```

% DIAGONAL MATRICES
x = [1 2 3 5 8];
C = diag(x)
  
```



DIAGONAL MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z:
Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> type ex_06_06.m
% DIAGONAL MATRICES
x = [1 2 3 5 8];
C = diag(x)
>>

Command History
C = B * A
clc
type ex_06_06.m
>>
  
```



DIAGONAL MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> ex_06_06
C =
    1    0    0    0    0
    0    2    0    0    0
    0    0    3    0    0
    0    0    0    5    0
    0    0    0    0    8
>> |
Command History
ex_06_04
clc
ex_06_06

```



IDENTITY MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

IDENTITY MATRICES

An identity matrix is a square matrix with ones on the main diagonal.

An identity matrix is special because

$$AI = IA = A$$

for any compatible matrix A. This is like multiplying by one in scalar arithmetic.

$$I = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}$$



IDENTITY MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_07.m

```

% IDENTITY MATRIX
I = eye (4);
A = magic(4);
C = A * I
C = I * A
I = eye (4,3)

```



IDENTITY MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_06_07.m
% IDENTITY MATRIX
I = eye (4);
A = magic(4);
C = A * I
C = I * A
I = eye (4,3)
>> |
Command History
ex_06_06
clc
type ex_06_07.m

```



IDENTITY MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_06_07.m
% IDENTITY MATRIX
I = eye (4);
A = magic(4);
C = A * I
C = I * A
I = eye (4,3)
>> ex_06_07

C =

    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1

Command History
>> clc
>> type ex_06_07.m
>> ex_06_07

```



IDENTITY MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
C = I * A
I = eye (4,3)
>> ex_06_07

C =

    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1

Command History
>> clc
>> type ex_06_07.m
>> ex_06_07

```



IDENTITY MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
    4    14    15     1

C =

    16     2     3    13
     5    11    10     8
     9     7     6    12
     4    14    15     1

I =

     1     0     0
     0     1     0
     0     0     1
     0     0     0

Command History
>> clc
>> type ex_06_07.m
>> ex_06_07

```



SYMMETRIC MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SYMMETRIC MATRICES

$$a_{ij} = a_{ji}$$

EXAMPLE:

filename:ex_06_08.m

```
% SYMMETRIC MATRIX
```

```
A = [1 2 4; 2 1 8; 4 8 1];
```

```
dimA = size(A);
```

```
m = dimA(1);
```

```
n = dimA(2);
```

```
if (isequal(A, A') == 1)
```

```
    disp('A IS A SYMMETRIC MATRIX.')
```

```
else
```

```
    disp('A IS NOT A SYMMETRIC MATRIX.')
```

```
end
```



SYMMETRIC MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> type ex_06_08.m

% SYMMETRIC MATRIX
A = [1 2 4; 2 1 8; 4 8 1];
dimA = size(A);
m = dimA(1);
n = dimA(2);

if (isequal(A, A') == 1)
    disp('A IS A SYMMETRIC MATRIX.')
else
    disp('A IS NOT A SYMMETRIC MATRIX.')
end
end
>>

Command History
ex_06_07
clc
type ex_06_08.m
Start

```



SYMMETRIC MATRICES

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_linequ... M-file

>> type ex_06_08.m

% SYMMETRIC MATRIX
A = [1 2 4; 2 1 8; 4 8 1];
dimA = size(A);
m = dimA(1);
n = dimA(2);

if (isequal(A, A') == 1)
    disp('A IS A SYMMETRIC MATRIX.')
else
    disp('A IS NOT A SYMMETRIC MATRIX.')
end
end
>> ex_06_08
A IS A SYMMETRIC MATRIX.

Command History
ex_06_08
clc
type ex_06_08.m
ex_06_08
Start

```



INVERSE OF A MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

INVERSE OF A MATRIX

Inverse of matrix A is A^{-1} . $AA^{-1} = A^{-1}A = I$

$$Ax = b$$

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

So, the solution of $Ax = b$ is

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



INVERSE OF A MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

filename: ex_06_09.m

$$A = [2 \ -1 \ 0; \ 1 \ 2 \ 1; \ 0 \ -1 \ 2]$$

$$B = \text{inv}(A)$$

$$A * B$$

$$B * A$$



INVERSE OF A MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_06_09.m
A = [2 -1 0; 1 2 1; 0 -1 2]
B = inv(A)
A * B
B * A
>> |
Command History
ex_06_08
clc
type ex_06_09.m
Start

```



INVERSE OF A MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> ex_06_09
A =
    2   -1    0
    1    2    1
    0   -1    2
B =
    0.4167    0.1667   -0.0833
   -0.1667    0.3333   -0.1667
   -0.0833    0.1667    0.4167
ans =
    clc
    type ex_06_09.m
    ex_06_09
    1.0000    0    0
Start

```



INVERSE OF A MATRIX

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
B =
    0.4167    0.1667   -0.0833
   -0.1667    0.3333   -0.1667
   -0.0833    0.1667    0.4167
ans =
    1.0000    0    0
   -0.0000    1.0000   -0.0000
    0    0    1.0000
ans =
    1.0000    0.0000    0
    0    1.0000    0
    0    0.0000    1.0000
Command History
clc
type ex_06_09.m
ex_06_09
Start

```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

- zeros: creates a matrix that all elements are equal to zero.
- ones: creates a matrix that all elements are equal to one.
- size: returns the dimension of the matrix.
- eye: creates an identity matrix.
- diag: creates a diagonal matrix
- inv: creates the inverse of a given matrix.
- trace: returns the sum of the diagonal terms of a matrix.
- det: returns the determinant of a matrix.
- \: left division
- /: right division



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> A = magic(3)
A =
     8     1     6
     3     5     7
     4     9     2
>> |
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> A = magic(3)
A =
     8     1     6
     3     5     7
     4     9     2
>> A(2, 3)
ans =
     7
>> |
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
     8     1     6
     3     5     7
     4     9     2
>> A(2, 3)
ans =
     7
>> A(:, 3)
ans =
     6
     7
     2
>> |
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
ans =
     7
>> A(:, 3)
ans =
     6
     7
     2
>> A(2, :)
ans =
     3     5     7
>> |
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z: Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_lineq... M-file
Command History
A(:, 3)
A(2, :)
A = eye(3)
>>
6
7
2
>> A(2, :)
ans =
3 5 7
>> A = eye(3)
A =
1 0 0
0 1 0
0 0 1
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z: Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_lineq... M-file
Command History
A(2, :)
A = eye(3)
size(A)
>>
ans =
3 5 7
>> A = eye(3)
A =
1 0 0
0 1 0
0 0 1
>> size(A)
ans =
3 3
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z: Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_lineq... M-file
Command History
A = eye(3)
size(A)
trace(A)
>>
A =
1 0 0
0 1 0
0 0 1
>> size(A)
ans =
3 3
>> trace(A)
ans =
3
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
All Files z: Type
ex_06_01.m M-file
ex_06_02.m M-file
ex_06_03.m M-file
ex_06_04.m M-file
ex_06_05.m M-file
ex_06_06.m M-file
ex_06_07.m M-file
ex_06_08.m M-file
ex_06_09.m M-file
ex_06_lineq... M-file
Command History
size(A)
trace(A)
A = ones(3,3)
>>
ans =
3 3
>> trace(A)
ans =
3
>> A = ones(3,3)
A =
1 1 1
1 1 1
1 1 1
>>
  
```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
3
>> A = ones(3,3)
A =
    1    1    1
    1    1    1
    1    1    1
>> A = zeros(3,3)
A =
    0    0    0
    0    0    0
    0    0    0
Command History
trace(A)
A = ones(3,3)
A = zeros(3,3)
>>

```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
1 1 1
1 1 1
>> A = zeros(3,3)
A =
    0    0    0
    0    0    0
    0    0    0
>> A = magic(3);
>> det(A)
ans =
   -360
Command History
A = zeros(3,3)
A = magic(3);
det(A)
>>

```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> A
A =
    8    1    6
    3    5    7
    4    9    2
>> inv(A)
ans =
    0.1472   -0.1444    0.0639
   -0.0611    0.0222    0.1056
   -0.0194    0.1889   -0.1028
Command History
clc
A
inv(A)
>>

```



SOME MATRIX FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLES:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
4 9 2
>> inv(A)
ans =
    0.1472   -0.1444    0.0639
   -0.0611    0.0222    0.1056
   -0.0194    0.1889   -0.1028
>> diag(A(:,1))
ans =
    8    0    0
    0    3    0
    0    0    4
Command History
A
inv(A)
diag(A(:,1))
>>

```



LINEAR EQUATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

LINEAR EQUATIONS

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2$$

...

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n$$

$$Ax = b$$

Unknowns could be calculated with matrix operations.

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



SOLUTION OF LINEAR EQUATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

- Use matrix operations to solve the following systems of linear equations.

$$4x - 2y + 6z = 8$$

$$2x + 8y + 2z = 4$$

$$6x + 10y + 3z = 0$$



SOLUTION OF LINEAR EQUATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

$$A = [4 \ -2 \ 6; \dots$$

$$2 \ 8 \ 2; \dots$$

$$6 \ 10 \ 3];$$

$$b = [8; 4; 0];$$

```
% ***
```

```
% SOLVING BY USING LEFT DIVISION
```

```
% ***
```

$$X_left = A \setminus b$$

```
% ***
```

```
% SOLVING BY USING INVERSE
```

```
% ***
```

$$X_inv = inv(A) * b$$



SOLUTION OF LINEAR EQUATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```

MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
>> type ex_06_linequ_1.m
A = [4 -2 6; ...
     2 8 2; ...
     6 10 3];
b = [8; 4; 0];
% ***
% SOLVING BY USING LEFT DIVISION
% ***
X_left = A \ b
% ***
% SOLVING BY USING INVERSE
% ***
X_inv = inv(A) * b
>>
Command History
ex_06_linequ_1
c1c
type ex_06_linequ_1
Start

```



SOLUTION OF LINEAR EQUATIONS

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

EXAMPLE:

```
MATLAB 7.6.0 (R2008a)
File Edit Debug Desktop Window Help
Current Directory: /media/Transcend/source
Command Window
% ***
X_inv = inv(A) * b
>> ex_06_linequ_1
X_left =
    -1.8049
     0.2927
     2.6341
X_inv =
    -1.8049
     0.2927
     2.6341
Command History
>> type ex_06_line
ex_06_linequ_1
>>
```



References

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

References for Week 6

- 1 Alfio Quarteroni, Fausto Saleri, Wissenschaftliches Rechnen mit Matlab, Springer, 2006.
- 2 Klemens Burg, Herbert Haf, Friedrich Wille, Höhere Mathematik für Ingenieure, Band II, Lineare Algebra, Teubner, 2008.