INTRODUCTION TO SCIENTIFIC & ENGINEERING COMPUTING BIL 108E, CRN24023

Dr. S. Gökhan Karaman

Technical University of Istanbul

March 15, 2010

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

LECTURE # 6 LINEAR ALGEBRA

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 x n) or (n x 1).

VECTOR INNER PRODUCT

VECTOR INNER PRODUCT

- In physics, analytical geometry, and engineering, the dot product has a geometric interpretation.
  $a \cdot b = a_1\, b_1 + a_2\, b_2 + \ldots + a_n\, b_n$

  `c = 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, `dot(a,b)` is the same as `a * b'`.

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR INNER PRODUCT

EXAMPLE:

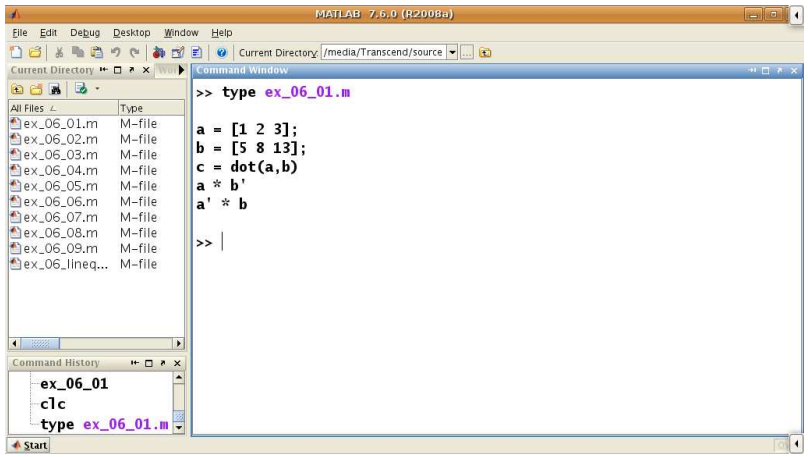
filename: ex_ 06_ 01.m

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

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR INNER PRODUCT

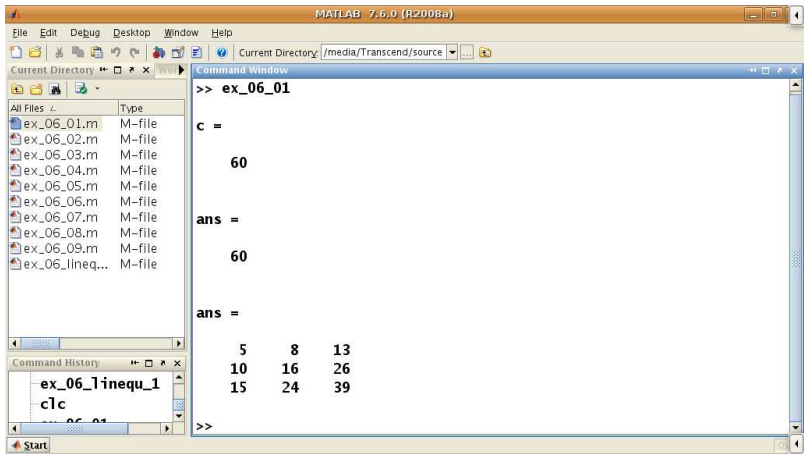
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR INNER PRODUCT

EXAMPLE:



VECTOR INNER PRODUCT

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.

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

$$= x_1 y_1 + x_2 y_2 + \dots x_n y_n$$

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR INNER PRODUCT

VECTOR INNER PRODUCT

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

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR OUTER PRODUCT

VECTOR OUTER PRODUCT

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

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

$$= \begin{bmatrix} u_1 v_1 & u_1 v_2 & \dots & u_1 v_n \\ u_2 v_1 & u_2 v_2 & \dots & u_2 v_n \\ \vdots & \vdots & \vdots & \vdots \\ u_m v_1 & u_m v_2 & \dots & u_m v_n \end{bmatrix}$$

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR OUTER PRODUCT

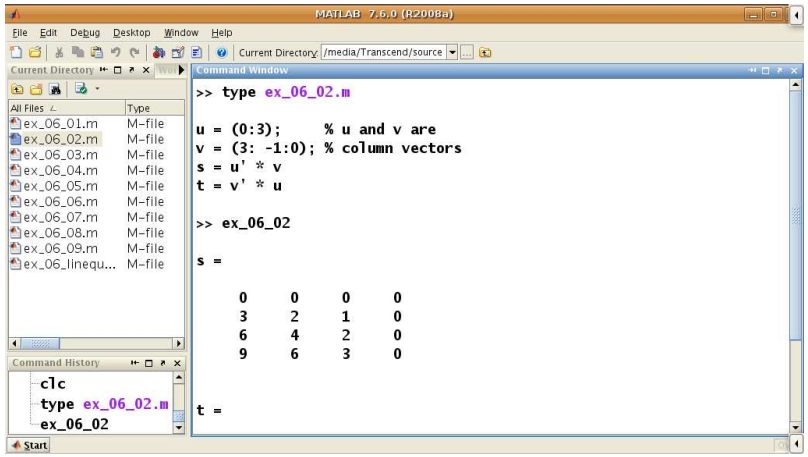
EXAMPLE:

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

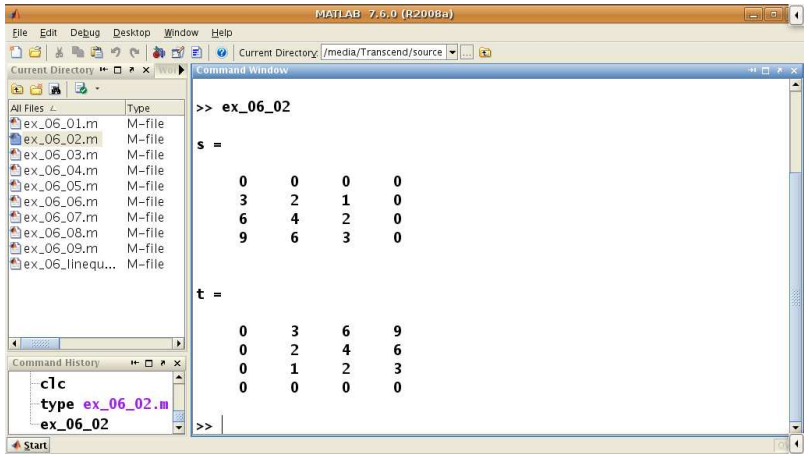
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

VECTOR OUTER PRODUCT

EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

MATRICES

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

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

MATRIX OPERATIONS

MATRIX OPERATIONS

- Addition and subtraction
  $C = A + B$
  $c(i,j) = a(i,j) + b(i,j)$
  $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$
- Multiplication by a scalar
  $B = \lambda A$
  $b(i,j) = \lambda a(i,j)$ $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$

MATRIX OPERATIONS

EXAMPLE:

filename: ex_ 06_ 03.m

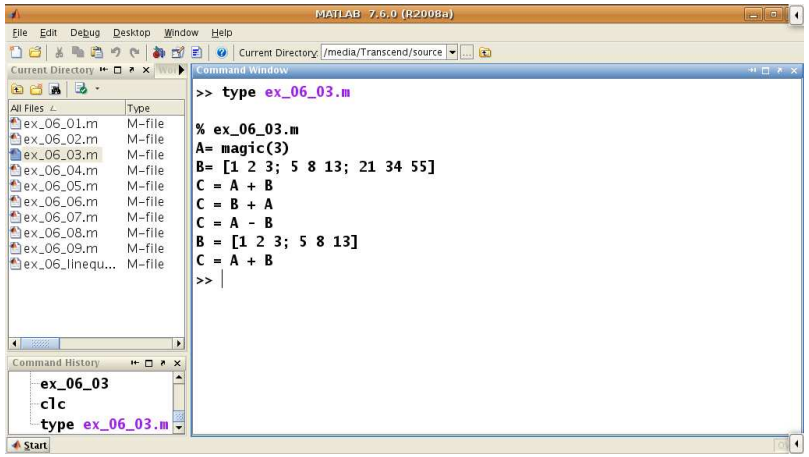
```
% ex_06_03.m
A= magic(3)
B= [1 2 3; 5 8 13; 21 34 55]
C = A + B
C = B + A
C = A - B
B = [1 2 3; 5 8 13]
C = A + B
```

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX OPERATIONS

EXAMPLE:

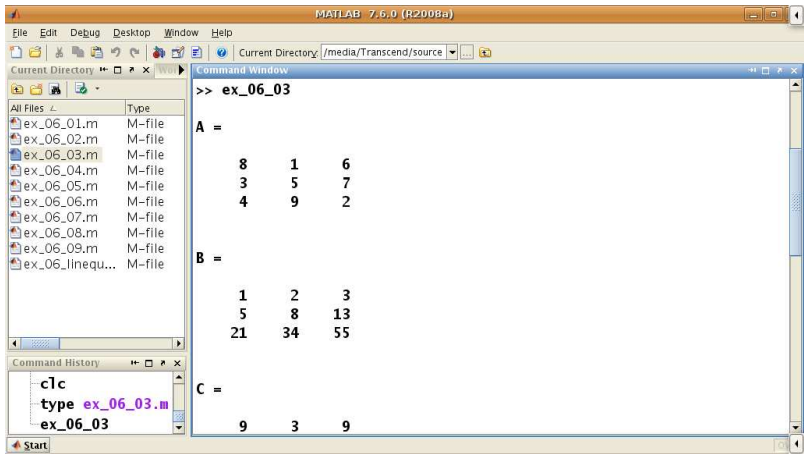


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX OPERATIONS

EXAMPLE:

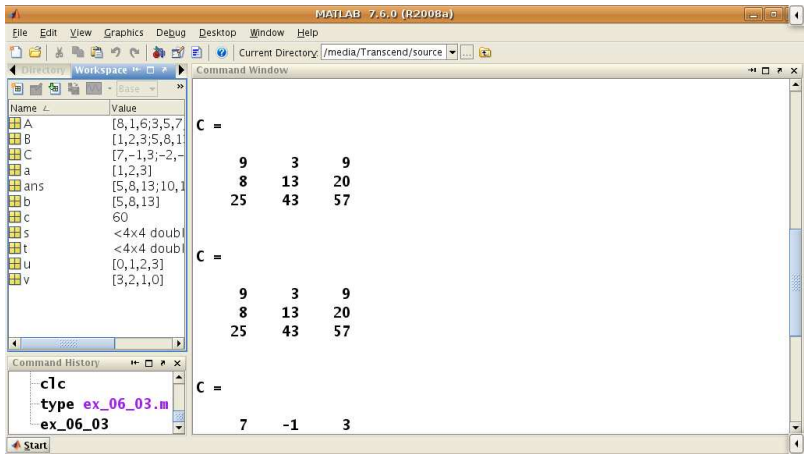


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

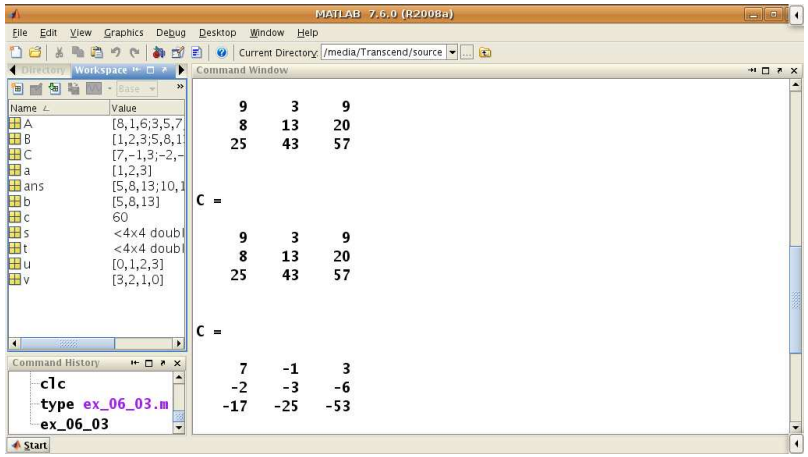
MATRIX OPERATIONS

EXAMPLE:



MATRIX OPERATIONS

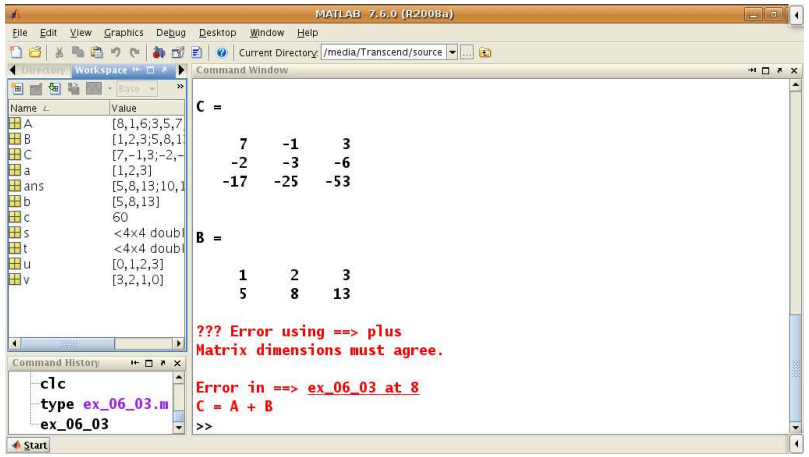
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

MATRIX OPERATIONS

EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

SQUARE MATRIX

SQUARE MATRIX

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

EXAMPLE:

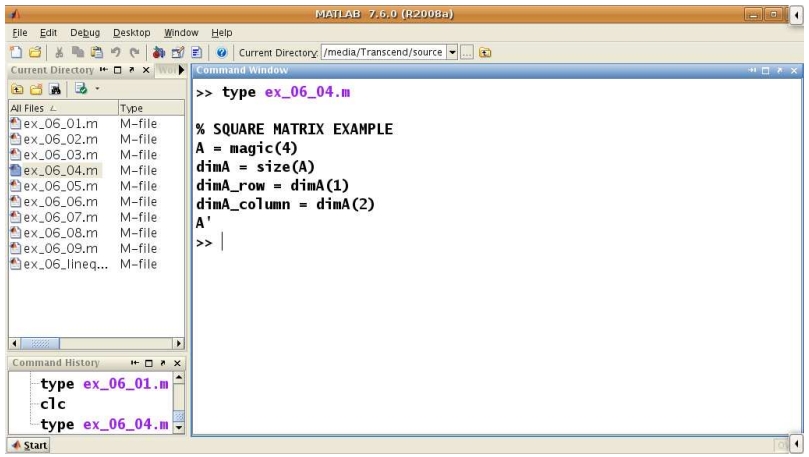
filename: ex_06_04.m

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

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

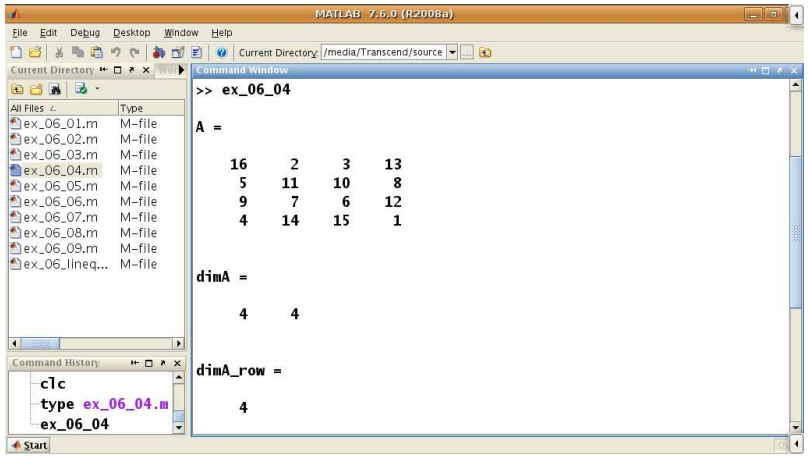
SQUARE MATRIX

EXAMPLE:



SQUARE MATRIX

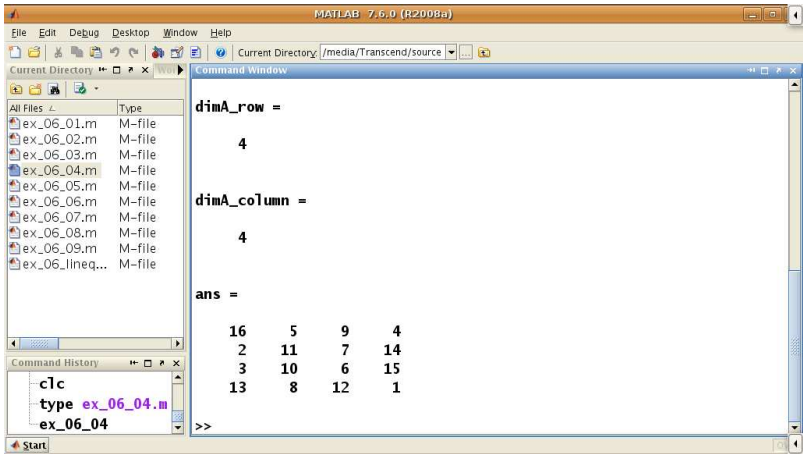
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

SQUARE MATRIX

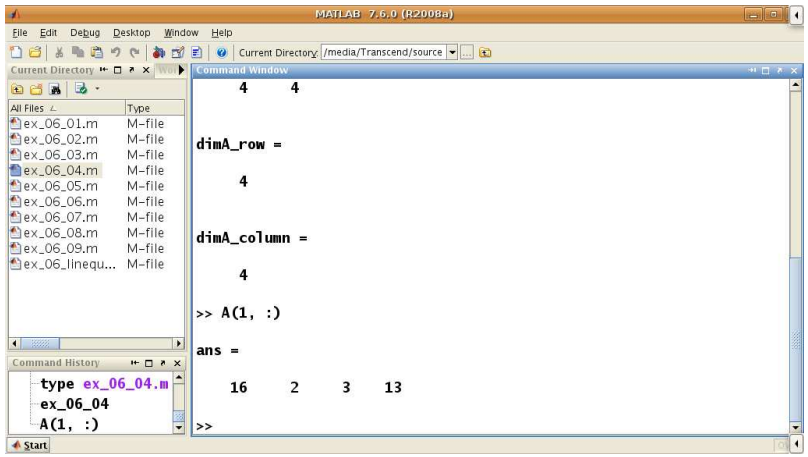
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

SQUARE MATRIX

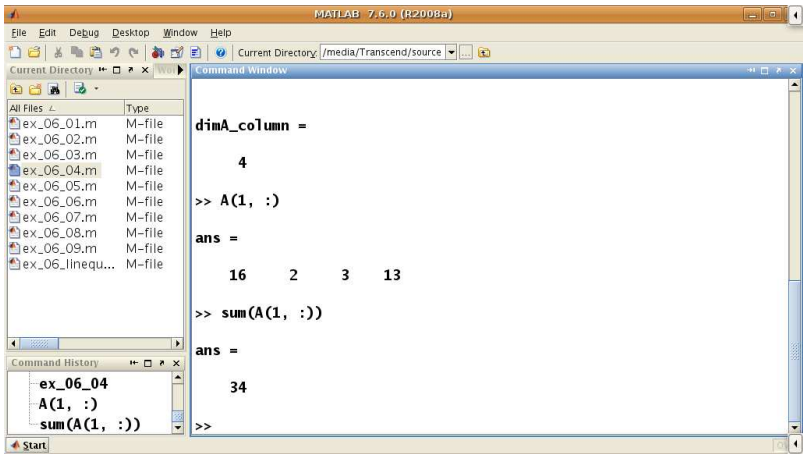
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

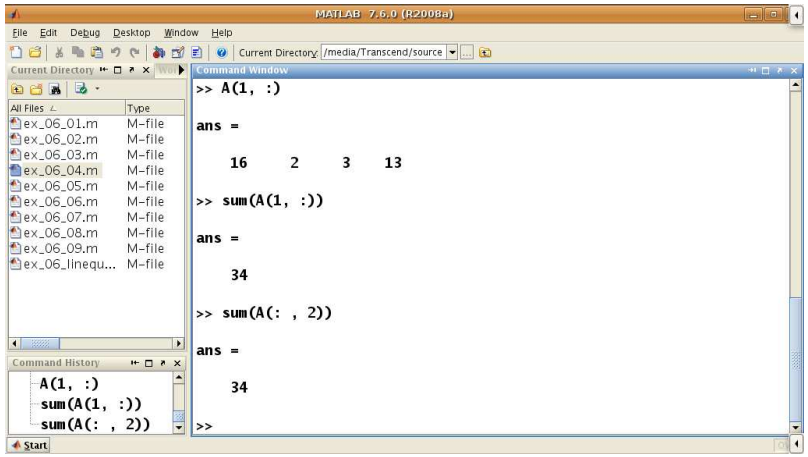
SQUARE MATRIX

EXAMPLE:



SQUARE MATRIX

EXAMPLE:



MATRIX TRANSPOSE

MATRIX TRANSPOSE

$B = A^T$

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

$i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$

MATRIX MULTIPLICATION

MATRIX MULTIPLICATION

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

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

MATRIX MULTIPLICATION

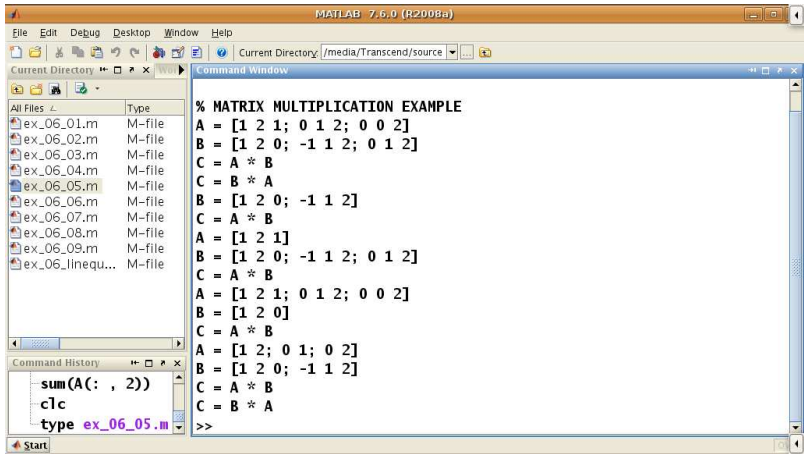
EXAMPLE:

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

EXAMPLE:

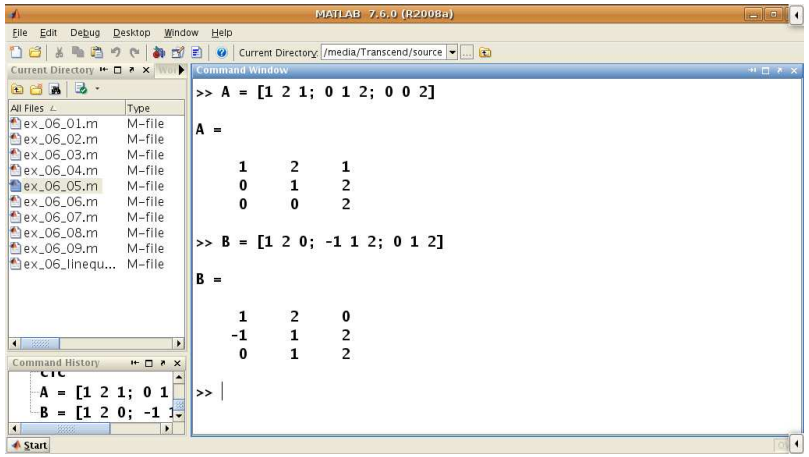


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX MULTIPLICATION

EXAMPLE:

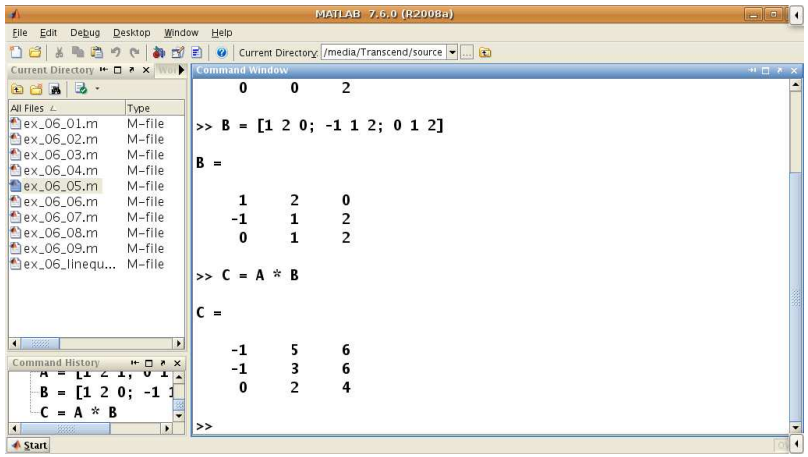


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX MULTIPLICATION

EXAMPLE:

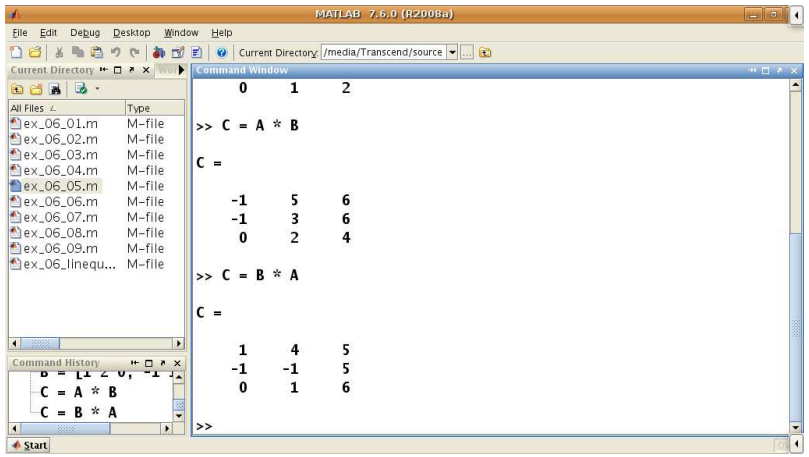


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

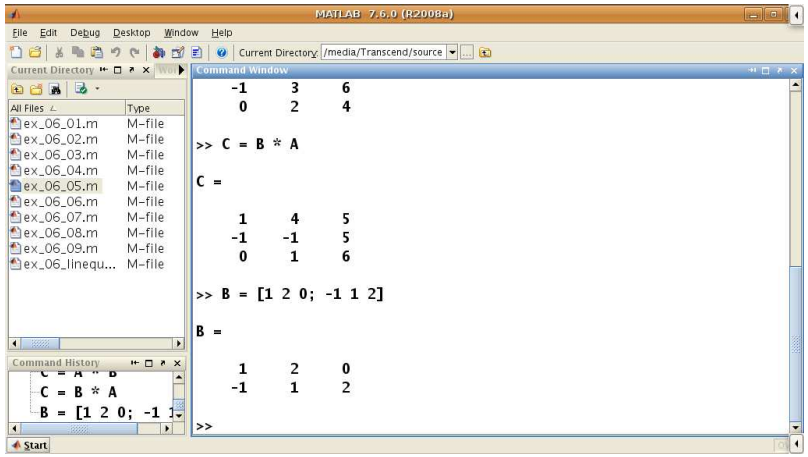
MATRIX MULTIPLICATION

EXAMPLE:



MATRIX MULTIPLICATION

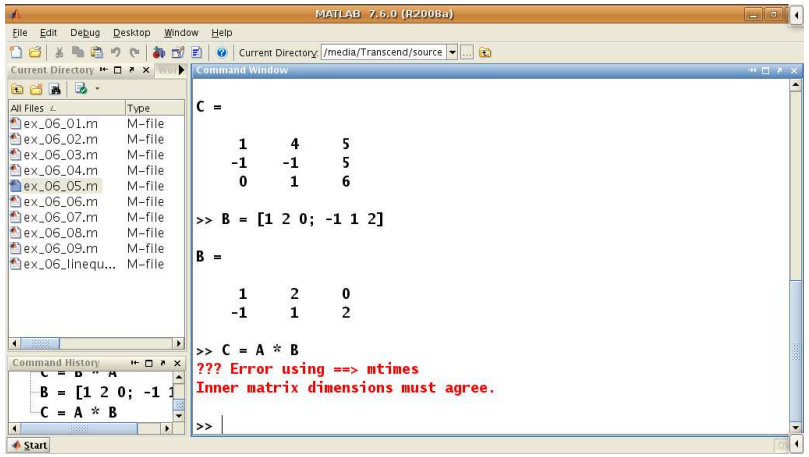
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

MATRIX MULTIPLICATION

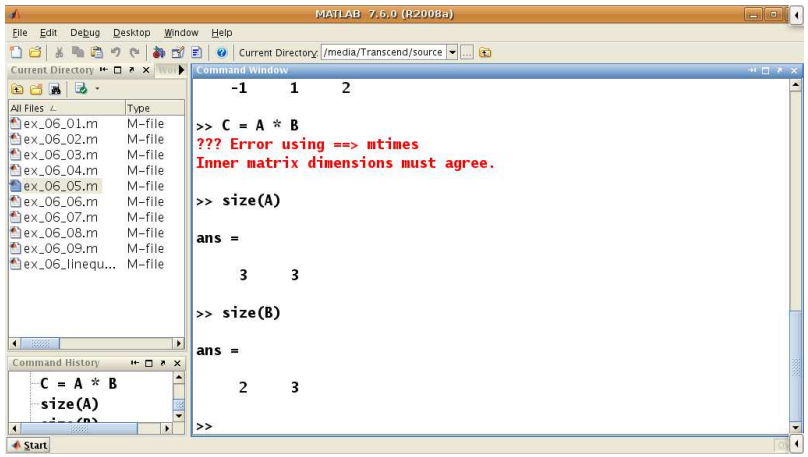
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

MATRIX MULTIPLICATION

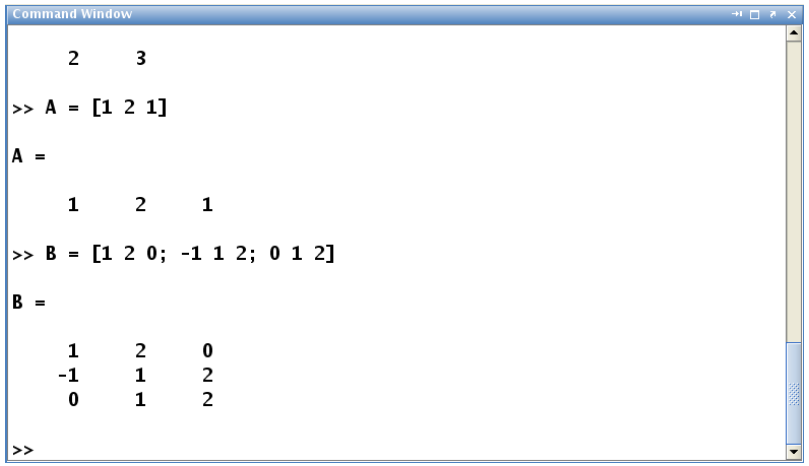
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

MATRIX MULTIPLICATION

EXAMPLE:



MATRIX MULTIPLICATION

EXAMPLE:

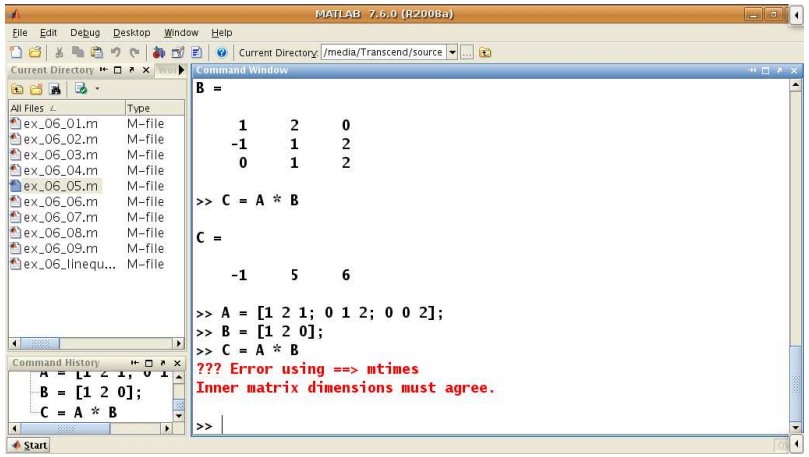
```
Command Window

     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

>>
```

Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX MULTIPLICATION

EXAMPLE:

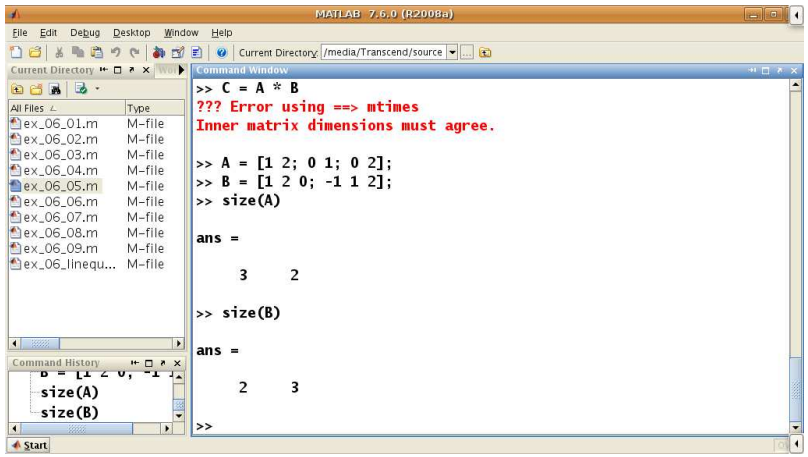


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

MATRIX MULTIPLICATION

EXAMPLE:

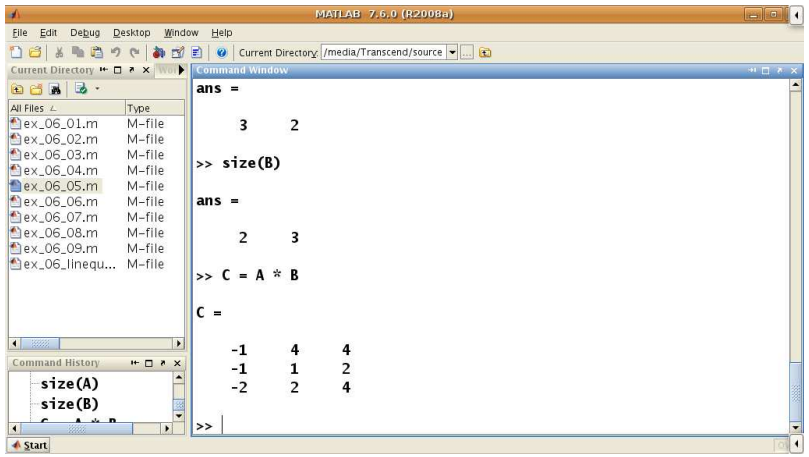


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

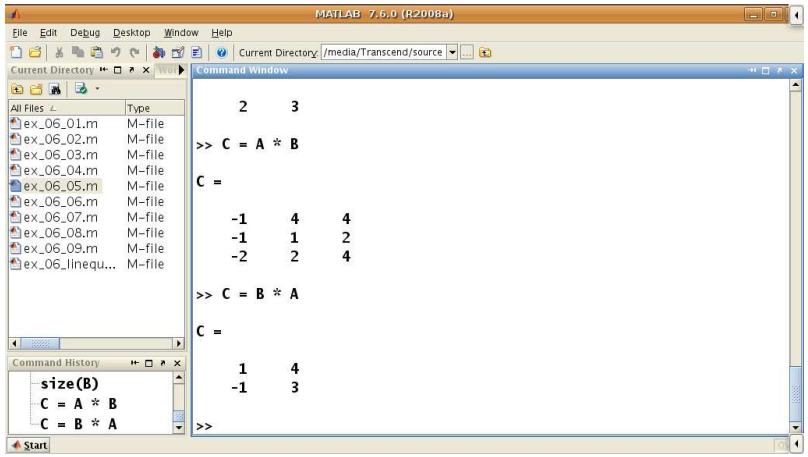
MATRIX MULTIPLICATION

EXAMPLE:



MATRIX MULTIPLICATION

EXAMPLE:



DIAGONAL MATRICES

DIAGONAL MATRICES

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

$C = diag(c_1, c_2, \ldots, c_n)$

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

DIAGONAL MATRICES

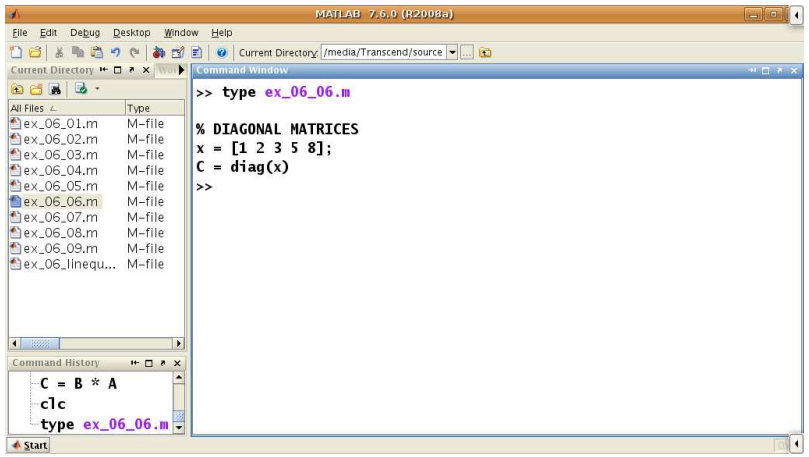
EXAMPLE:

filename: ex_06_06.m

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

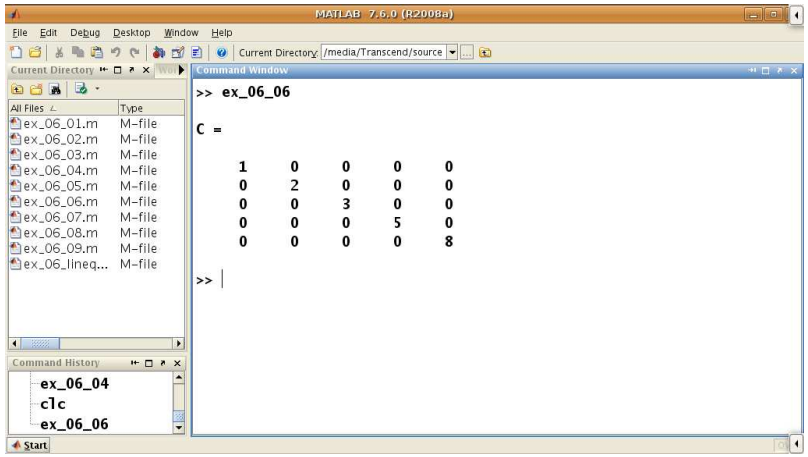

DIAGONAL MATRICES

EXAMPLE:



DIAGONAL MATRICES

EXAMPLE:



IDENTITY MATRICES

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.

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

IDENTITY MATRIX

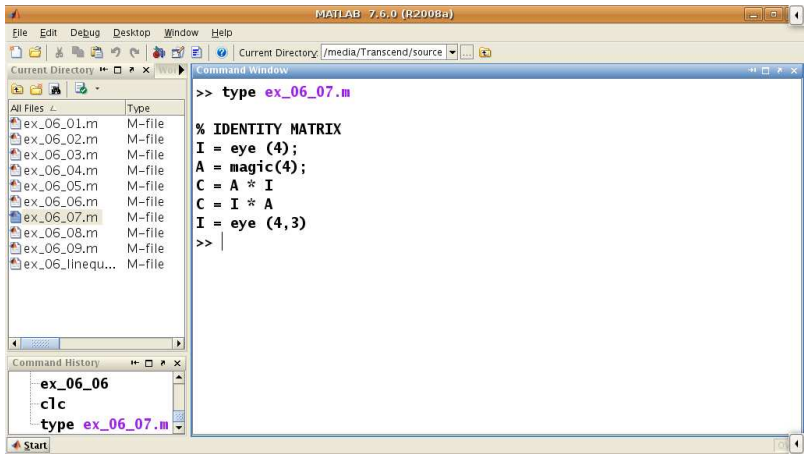
EXAMPLE:

filename: ex_ 06_ 07.m

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

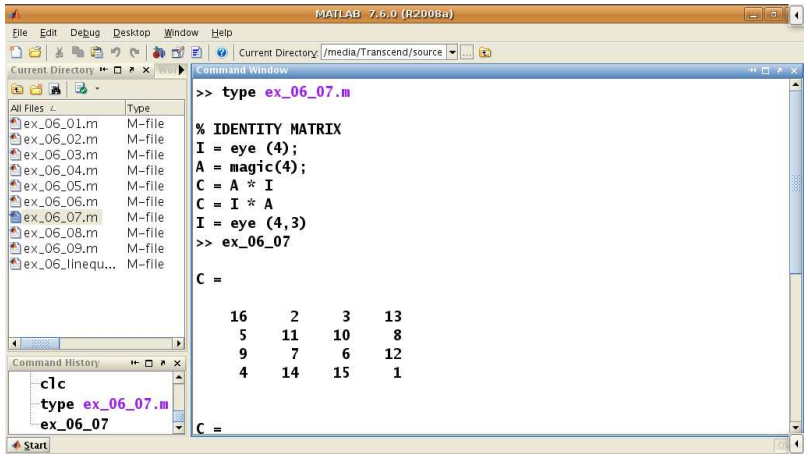

IDENTITY MATRIX

EXAMPLE:



IDENTITY MATRIX

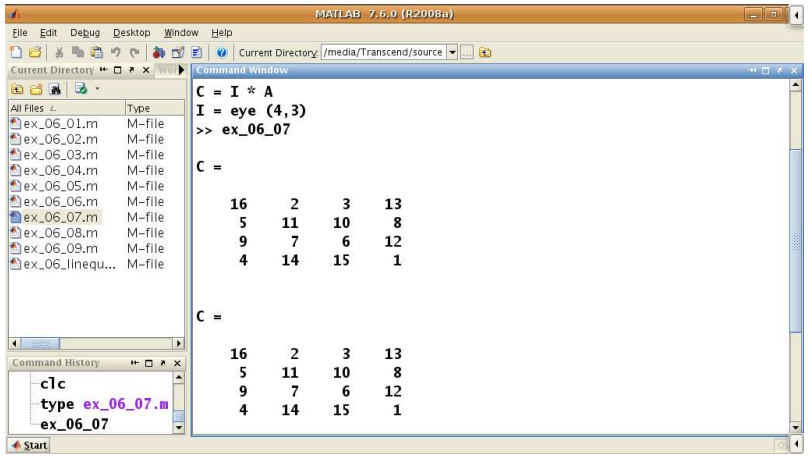
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

IDENTITY MATRIX

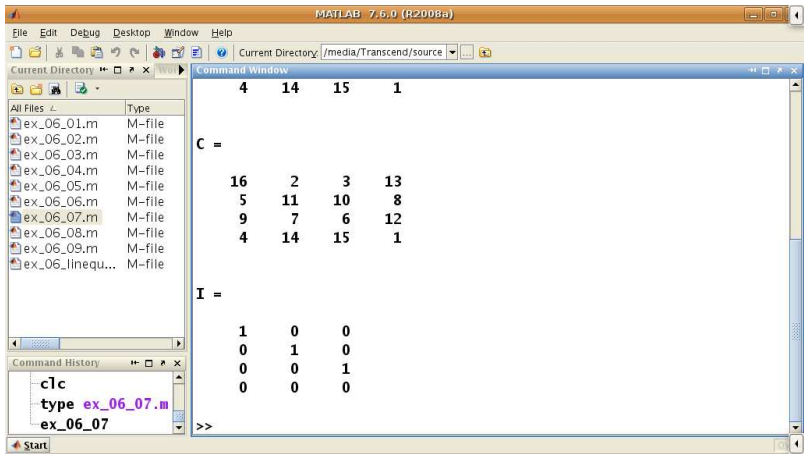
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

IDENTITY MATRIX

EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

SYMMETRIC MATRICES

SYMMETRIC MATRICES

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

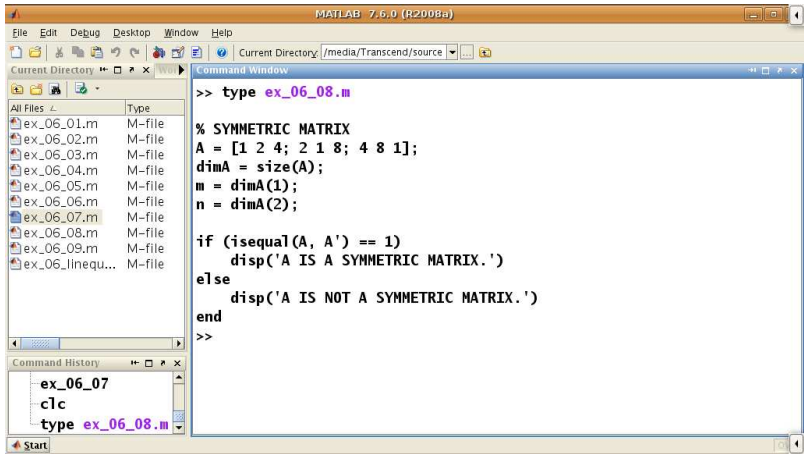
EXAMPLE:

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

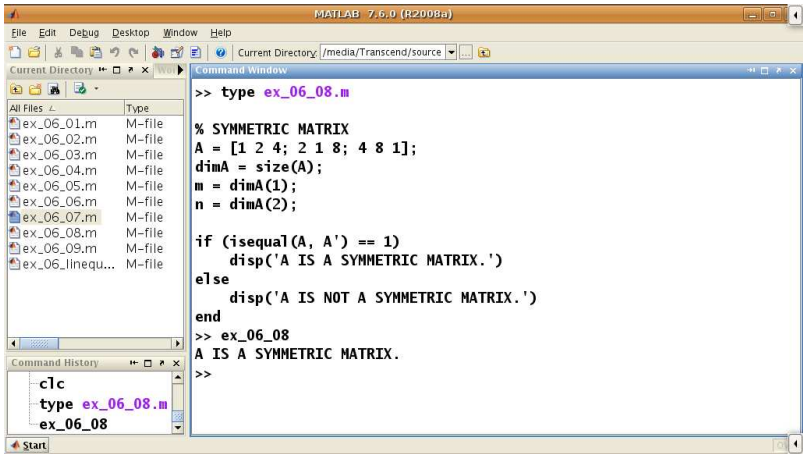
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

SYMMETRIC MATRICES

EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

INVERSE OF A MATRIX

INVERSE OF A MATRIX

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

$A\,x = b$

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

So, the solution of $A\,x = b$ is

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

Introduction to Scientific and Engineering Computing, BIL108E
Karaman

INVERSE OF A MATRIX

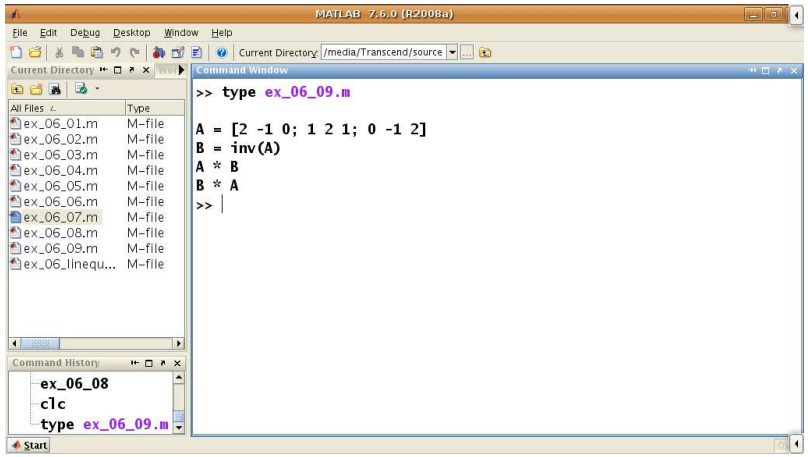
EXAMPLE:

filename: ex_ 06_ 09.m

```
A = [2 -1 0; 1 2 1; 0 -1 2]
B = inv(A)
A * B
B * A
```

INVERSE OF A MATRIX

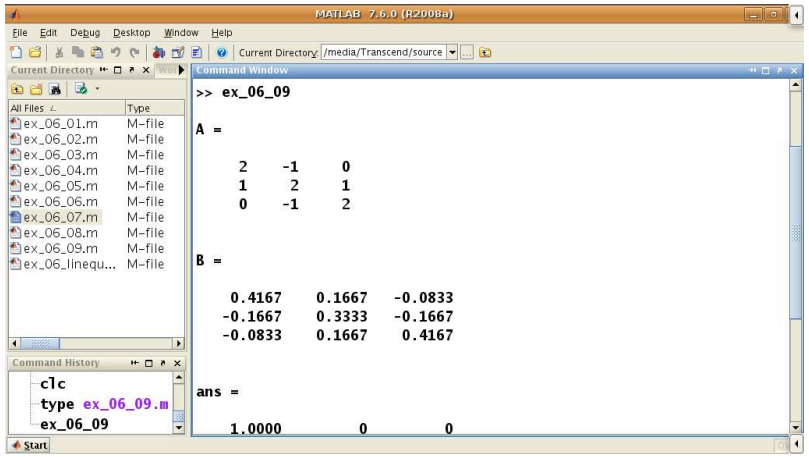
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

INVERSE OF A MATRIX

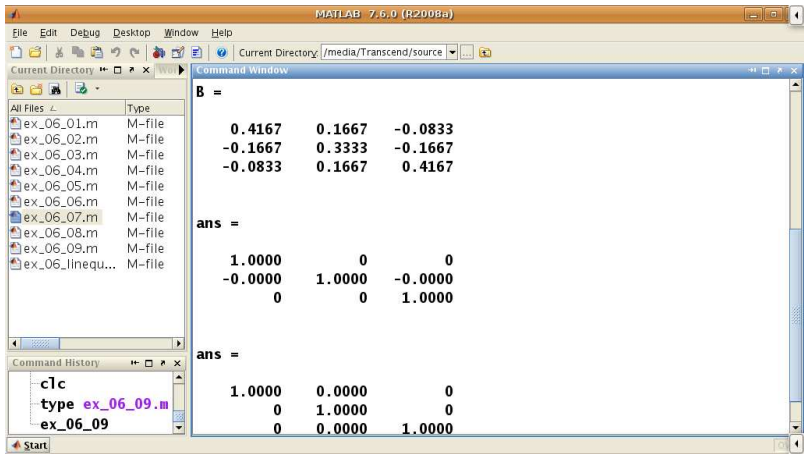
EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

INVERSE OF A MATRIX

EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E
Karaman

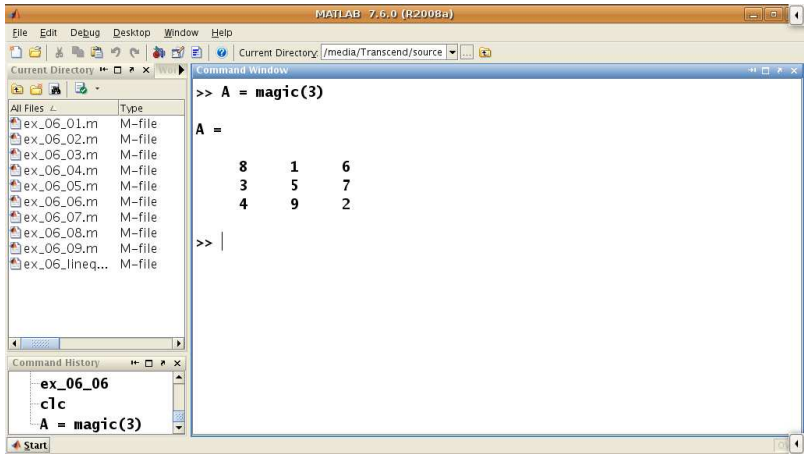
SOME MATRIX FUNCTIONS

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

EXAMPLES:

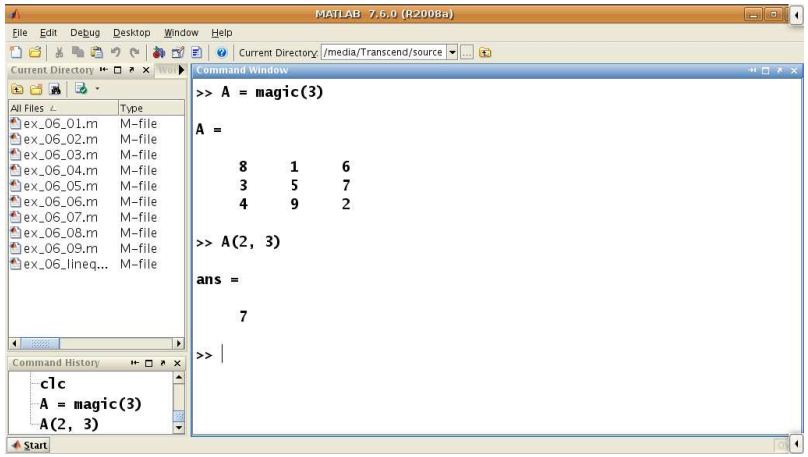


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:

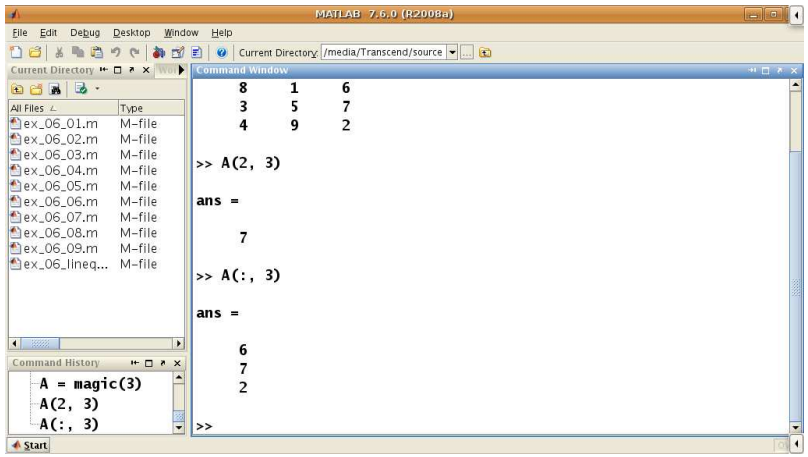


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:

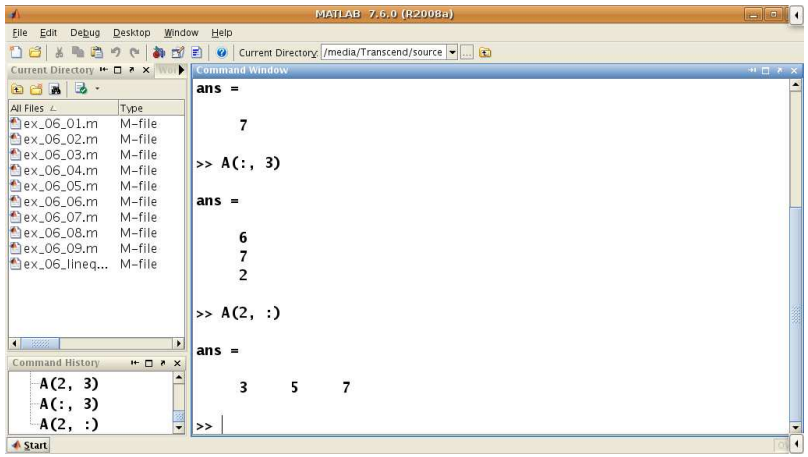


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

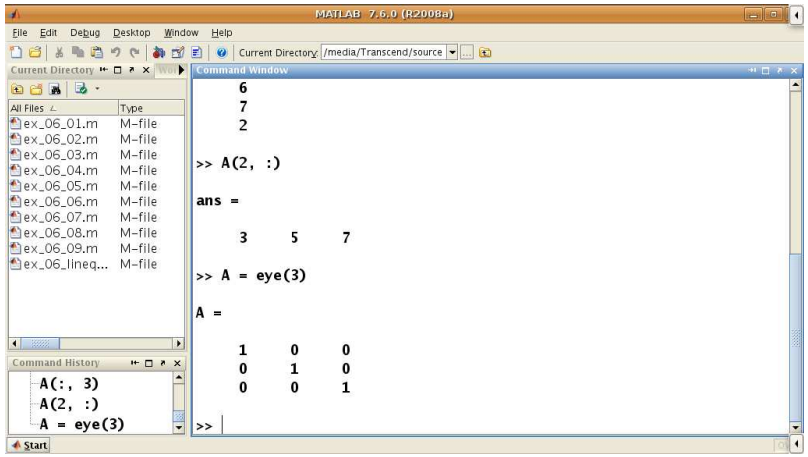
SOME MATRIX FUNCTIONS

EXAMPLES:



SOME MATRIX FUNCTIONS

EXAMPLES:

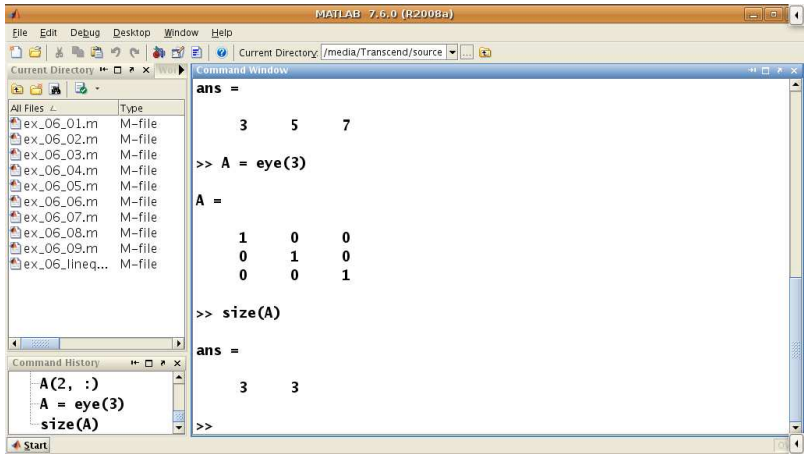


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:

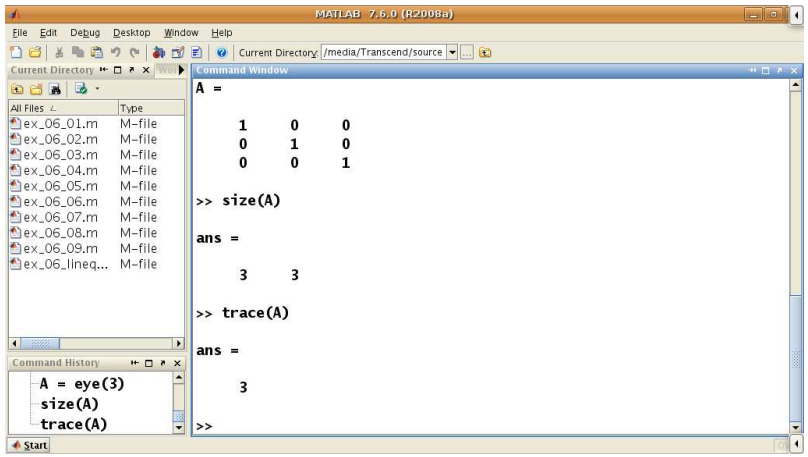


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:

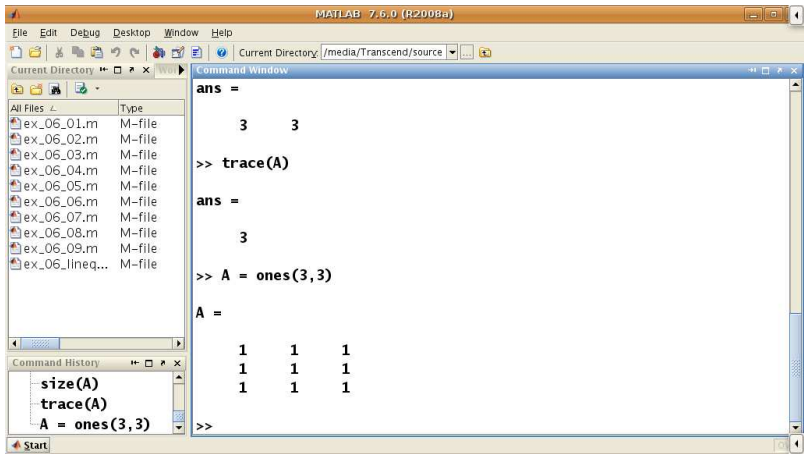


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

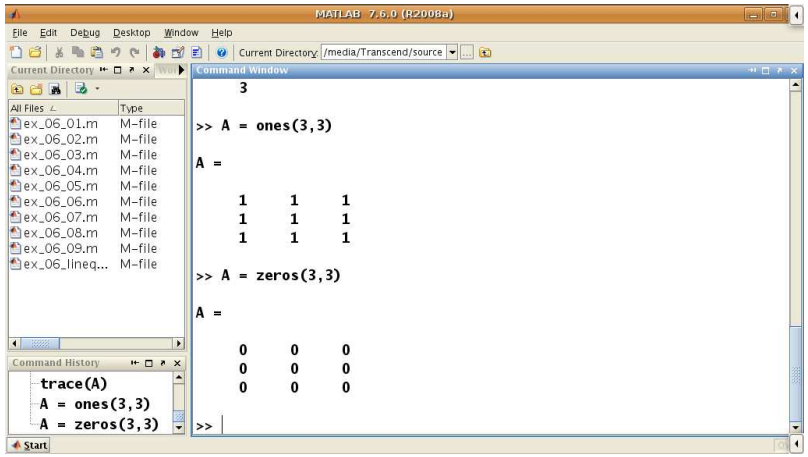
SOME MATRIX FUNCTIONS

EXAMPLES:



SOME MATRIX FUNCTIONS

EXAMPLES:

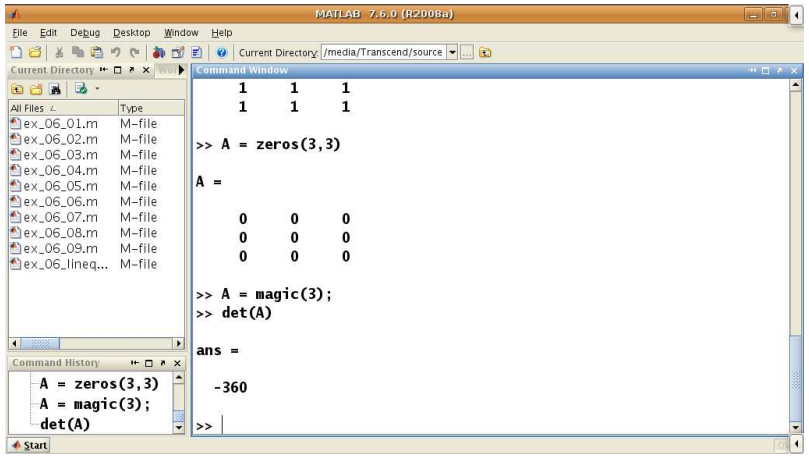


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:

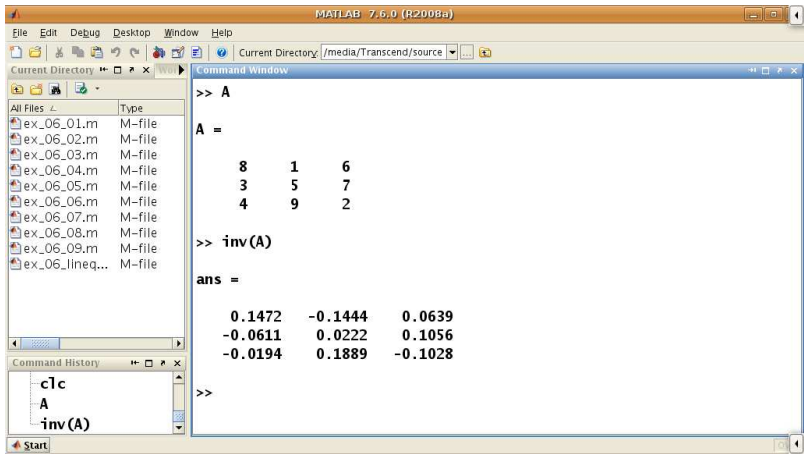


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:

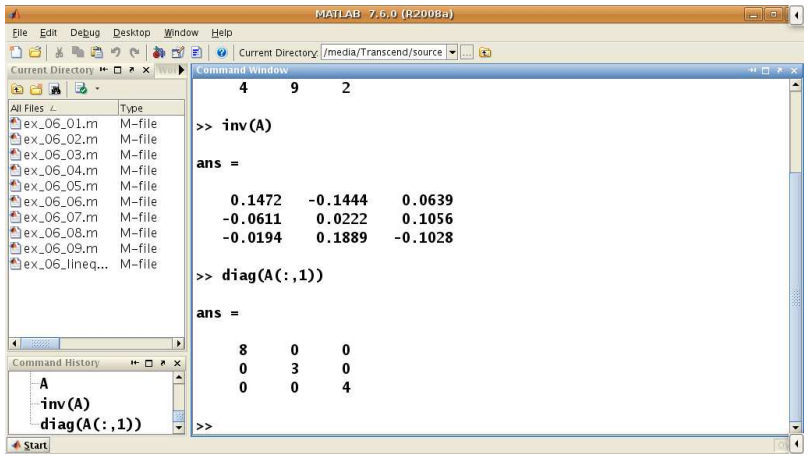


Introduction to Scientific and Engineering Computing, BIL108E

Karaman

SOME MATRIX FUNCTIONS

EXAMPLES:



LINEAR EQUATIONS

LINEAR EQUATIONS

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1$$
$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2$$
$$\ldots$$
$$a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{n\,n}x_n = b_n$$

$$A\,x = b$$

Unknowns could be calculated with matrix operations.
$x = A^{-1} \times b$

SOLUTION OF LINEAR EQUATIONS

EXAMPLE:

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

$$4x - 2y + 6z = 8$$
$$2x + 8y + 2z = 4$$
$$6x + 10y + 3z = 0$$

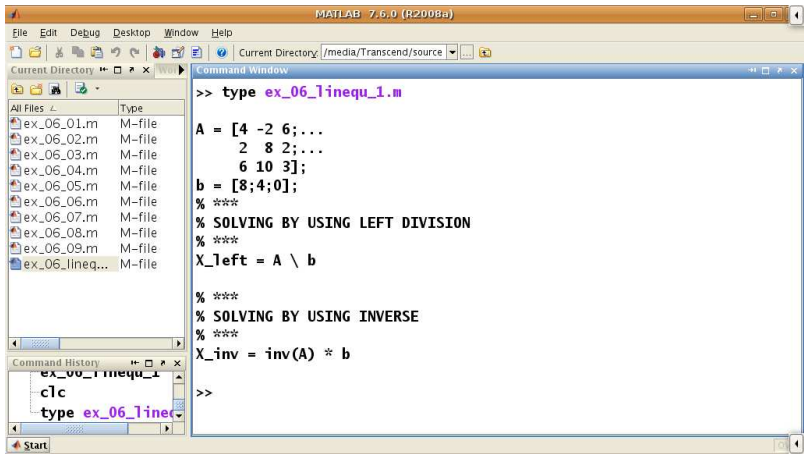
SOLUTION OF LINEAR EQUATIONS

EXAMPLE:

```
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
```

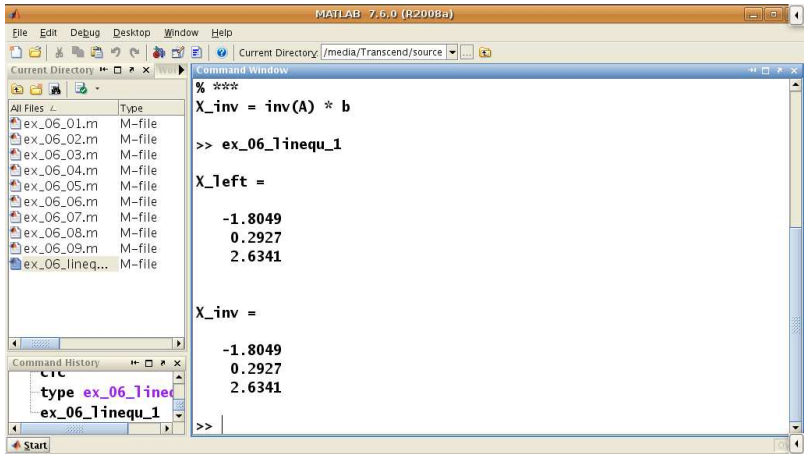
SOLUTION OF LINEAR EQUATIONS

EXAMPLE:



SOLUTION OF LINEAR EQUATIONS

EXAMPLE:



Introduction to Scientific and Engineering Computing, BIL108E

Karaman

References

References for Week 6

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