

Introduction to MATLAB

Selim Aksoy
Bilkent University
Department of Computer Engineering
saksoy@cs.bilkent.edu.tr

MATrix LABoratory

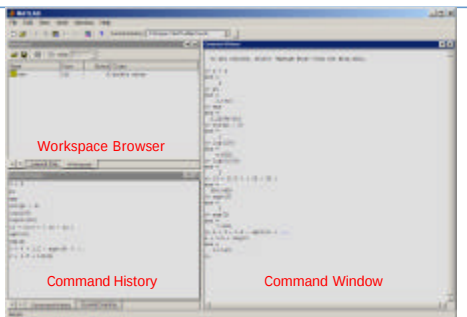
- www.mathworks.com
- Advantages of MATLAB
 - Ease of use
 - Platform independence
 - Predefined functions
 - Plotting
- Disadvantages of MATLAB
 - Can be slow
 - Expensive

Summer 2004

CS 111

2

MATLAB Desktop



Summer 2004

CS 111

3

MATLAB Basics

- A program can be input
 - command by command using the command line (lines starting with "»" on the MATLAB desktop)
 - as a series of commands using a file (a special file called **M-file**)
- If a command is followed by a semicolon (;), result of the computation is not shown on the command window

Summer 2004

CS 111

4

MATLAB Basics: Getting Help

- **help**
 - help *toolbox* → e.g., help elfun
 - help *command* → e.g., help sin
- **helpdesk**, **helpwin**, "?" button
- **lookfor**
 - lookfor *keyword* → e.g., lookfor cotangent
- **which**
 - which *name* → e.g., which log
- **demo**

Summer 2004

CS 111

5

MATLAB Basics: Scratchpad

2 * 2

$\cot(3)\sqrt{(\log(3))^3} + \cos(3)*\sin(\log(3))$

$\cot(2.7)\sqrt{(\log(2.7))^3} + \cos(2.7)*\sin(\log(2.7))$

$\log(\sin(0.5)+\cos(0.5)^2) + \sqrt[4]{\sin(0.5)+\cos(0.5)^2} - (\sin(0.5)+\cos(0.5)^2)^2$

Summer 2004

CS 111

6

MATLAB Basics: Variables

- **Variable** is a name given to a reserved location in memory
 - `class_code = 111;`
 - `number_of_students = 65;`
 - `name = 'Bilkent University';`
 - `radius = 5;`
 - `area = pi * radius^2;`

Summer 2004

CS 111

7

MATLAB Basics: Variables

- Use meaningful names for variables
- MATLAB variable names
 - must begin with a letter
 - can contain any combination of letters, numbers and underscore (`_`)
 - must be unique in the first 31 characters
- MATLAB is case sensitive: "name", "Name" and "NAME" are considered different variables
- Never use a variable with the same name as a MATLAB command
- Naming convention: use lowercase letters

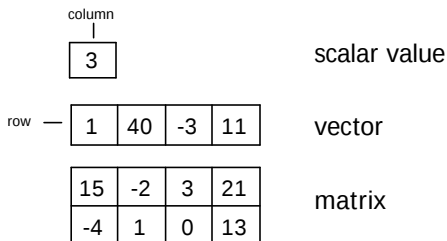
Summer 2004

CS 111

8

MATLAB Basics: Arrays

- The fundamental unit of data is **array**



Summer 2004

CS 111

9

MATLAB Basics: Variables

- Initialization using assignment statements

```
x = 5
x =
    5
y = x + 1
y =
    6
vector = [ 1 2 3 4 ]
vector =
    1    2    3    4
```

Summer 2004

CS 111

10

MATLAB Basics: Variables

- `matrix = [ 1 2 3; 4 5 6 ]`
`matrix =`
1 2 3
4 5 6
- `matrix = [ 1 2 3; 4 5 ]`
??? Error
- `a = [ 5 (2+4) ]`
`a =`
5 6

Summer 2004

CS 111

11

MATLAB Basics: Variables

- Initialization using shortcut statements

```
colon operator → first:increment:last
x = 1:2:10
x =
    1    3    5    7    9
y = 0:0.1:0.5
y =
    0    0.1    0.2    0.3    0.4    0.5
```

Summer 2004

CS 111

12

MATLAB Basics: Variables

- transpose operator → '
 - `u = [ 1:3 ]'`
`u =`
1
2
3
 - `v = [ u u ]`
`v =`
1 1
2 2
3 3
 - `v = [ u'; u' ]`
`v =`
1 2 3
1 2 3

Summer 2004

CS 111

13

MATLAB Basics: Variables

- Initialization using built-in functions
 - `zeros()`
 - `x = zeros(2)`
`x =`
0 0
0 0
 - `z = zeros(2,3)`
`z =`
0 0 0
0 0 0
 - `ones(), size(), length()`
 - `y = zeros(1,4)`
`y =`
0 0 0 0
 - `t = zeros( size(z) )`
`t =`
0 0 0
0 0 0

Summer 2004

CS 111

14

MATLAB Basics: Variables

- Initialization using keyboard input
 - `input()`
 - `value = input( 'Enter an input value: ' )`
Enter an input value: 1.25
`value =`
1.2500
 - `name = input( 'What is your name: ', 's' )`
What is your name: Selim
`name =`
Selim

Summer 2004

CS 111

15

MATLAB Basics: Subarrays

- Array indices start from 1
- `x = [ -2 0 9 1 4 ];`
 - `x(2)`
`ans =`
0
 - `x(4)`
`ans =`
1
 - `x(8)`
??? Error
 - `x(-1)`
??? Error

Summer 2004

CS 111

16

MATLAB Basics: Subarrays

- `y = [ 1 2 3; 4 5 6 ];`
 - `y(1,2)`
`ans =`
2
 - `y(2,1)`
`ans =`
4
 - `y(2)`
`ans =`
4 (column major order)

Summer 2004

CS 111

17

MATLAB Basics: Subarrays

- `y = [ 1 2 3; 4 5 6 ];`
 - `y(1,:)`
`ans =`
1 2 3
 - `y(:,2)`
`ans =`
2
5
 - `y(2,1:2)`
`ans =`
4 5
 - `y(1,2:end)`
`ans =`
2 3
 - `y(:,2:end)`
`ans =`
2 3
5 6

Summer 2004

CS 111

18

MATLAB Basics: Subarrays

- `x = [ -2 0 9 1 4 ];`
 - `x(2) = 5`
`x =`
-2 5 9 1 4
 - `x(4) = x(1)`
`x =`
-2 5 9 -2 4
 - `x(8) = -1`
`x =`
-2 5 9 -2 4 0 0 -1

Summer 2004

CS 111

19

MATLAB Basics: Subarrays

- `y = [ 1 2 3; 4 5 6 ];`
 - `y(1,2) = -5`
`y =`
1 -5 3
4 5 6
 - `y(2,1) = 0`
`y =`
1 -5 3
0 5 6
 - `y(1,2:end) = [ -1 9 ]`
`y =`
1 -1 9
0 5 6

Summer 2004

CS 111

20

MATLAB Basics: Subarrays

- `y = [ 1 2 3; 4 5 6; 7 8 9 ];`
 - `y(2:end,2:end) = 0`
`y =`
1 2 3
4 0 0
7 0 0
 - `y(2:end,2:end) = [ -1 5 ]`
??? Error
 - `y(2,[1 3]) = -2`
`y =`
1 2 3
-2 0 -2
7 0 0

Summer 2004

CS 111

21

MATLAB Basics: Special Values

- **pi**: π value up to 15 significant digits
- **i, j**: $\sqrt{-1}$
- **Inf**: infinity (such as division by 0)
- **NaN**: Not-a-Number (such as division of zero by zero)
- **clock**: current date and time as a vector
- **date**: current date as a string (e.g. 16-Feb-2004)
- **eps**: epsilon
- **ans**: default variable for answers

Summer 2004

CS 111

22

MATLAB Basics: Displaying Data

- Changing the data format
 - `value = 12.345678901234567`
format short $\rightarrow$ 12.3457
long $\rightarrow$ 12.34567890123457
short e $\rightarrow$ 1.2346e+001
long e $\rightarrow$ 1.234567890123457e+001
rat $\rightarrow$ 1000/81
compact
loose

Summer 2004

CS 111

23

MATLAB Basics: Displaying Data

- The **disp(array)** function
 - `disp( 'Hello' );`
Hello
 - `disp(5);`
5
 - `disp( [ 'Bilkent ' 'University' ] );`
Bilkent University
 - `name = 'Selim'; disp( [ 'Hello ' name ] );`
Hello Selim

Summer 2004

CS 111

24

MATLAB Basics: Displaying Data

- The **num2str()** and **int2str()** functions
 - `d = [ num2str(16) '-Feb-' num2str(2004) ];`
 - `disp(d);`
16-Feb-2004
 - `x = 23.11;`
 - `disp( [ 'answer = ' num2str(x) ] );`
answer = 23.11
 - `disp( [ 'answer = ' int2str(x) ] );`
answer = 23

Summer 2004

CS 111

25

MATLAB Basics: Displaying Data

- The **fprintf(*format*, *data*)** function
 - `%d` integer
 - `%f` floating point format
 - `%e` exponential format
 - `\n` new line character
 - `\t` tab character

Summer 2004

CS 111

26

MATLAB Basics: Displaying Data

- `fprintf( 'Result is %d', 3 );`
Result is 3
- `fprintf( 'Area of a circle with radius %d is %f', 3, pi*3^2 );`
Area of a circle with radius 3 is 28.274334
- `x = 5;`
- `fprintf( 'x = %3d', x );`
x = 5
- `x = pi;`
- `fprintf( 'x = %0.2f', x );`
x = 3.14
- `fprintf( 'x = %6.2f', x );`
x = 3.14
- `fprintf( 'x = %d\ny = %d\n', 3, 13 );`
x = 3
y = 13

Summer 2004

CS 111

27

MATLAB Basics: Data Files

- **save filename var1 var2 ...**
 - `save homework.mat x y` → binary
 - `save x.dat x -ascii` → ascii
- **load filename**
 - `load filename.mat` → binary
 - `load x.dat -ascii` → ascii

Summer 2004

CS 111

28

MATLAB Basics: Scalar Operations

- ***variable_name* = *expression*;**
 - addition `a + b` → `a + b`
 - subtraction `a - b` → `a - b`
 - multiplication `a x b` → `a * b`
 - division `a / b` → `a / b`
 - exponent `ab` → `a ^ b`

Summer 2004

CS 111

29

MATLAB Basics: Scalar Operations

- `x = 3 * 2 + 6 / 2`
 - `x = ?`
- Processing order of operations is important
 - parenthesis (starting from the innermost)
 - exponentials (left to right)
 - multiplications and divisions (left to right)
 - additions and subtractions (left to right)
- `x = 3 * 2 + 6 / 2`
 - `x = 9`

Summer 2004

CS 111

30

MATLAB Basics: Built-in Functions

- `result = function_name( input );`
 - `abs`, `sign`
 - `log`, `log10`, `log2`
 - `exp`
 - `sqrt`
 - `sin`, `cos`, `tan`
 - `asin`, `acos`, `atan`
 - `max`, `min`
 - `round`, `floor`, `ceil`, `fix`
 - `mod`, `rem`
- `help elfun`

Summer 2004

CS 111

31

MATLAB Basics: Debugging

- Syntax errors
 - Check spelling and punctuation
- Run-time errors
 - Check input data
 - Can remove “;” or add “disp” statements
- Logical errors
 - Use shorter statements
 - Check typos
 - Check units
 - Ask your friends, TAs, instructor, parents, ...

Summer 2004

CS 111

32

MATLAB Basics: Useful Commands

- `help command` → Online help
- `lookfor keyword` → Lists related commands
- `which` → Version and location info
- `clear` → Clears the workspace
- `clc` → Clears the command window
- `diary filename` → Sends output to file
- `diary on/off` → Turns diary on/off
- `who`, `whos` → Lists content of the workspace
- `more on/off` → Enables/disables paged output
- `Ctrl+c` → Aborts operation
- `...` → Continuation
- `%` → Comments

Summer 2004

CS 111

33