

#### Introduction to Scientific and Engineering Computing, BIL108E

## INTRODUCTION TO SCIENTIFIC & ENGINEERING COMPUTING BIL 108E, CRN24023

Dr. S. Gökhan Karaman

Technical University of Istanbul

March 08, 2010

### Tentative Course Schedule, CRN 24023

troduction Scientific and ngineering omputing,			
BIL108E	Week	Date	Topics
Karaman	1	Feb. 08	Introduction to Scientific and Engineering Computing
Naraman	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 # 5

Introduction to Scientific and Engineering Computing, BIL108E

### LECTURE # 5

- **1** INLINE FUNCTIONS
- 2 M-FILE
  - SCRIPT M-FILES
  - FUNCTION M-FILES
- **3** EXAMPLES
- **4** RECURSIVE FUNCTIONS
- **5** STRUCTURES
- 6 EXAMPLES



## INLINE FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

#### INLINE FUNCTIONS

- Short mathematical functions may be written as one=line inline objects.
- Usage:



## INLINE FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

A	MATLAB 7.6.0 (R2008a)		1
<u>File Edit Debug D</u> esktop <u>Win</u>	dow Help		
🗋 🙆 🕺 🖷 🛍 🤊 🛯 🚵 🖻	🛿 📄 🛛 Current Directory: /media 🛛 👻 📖 😰		
Current Directory 🏎 🗖 🔻 ₩	Command Window	- * 🗆 * - >	×
🖻 🖆 🖪 🖥 🔹 🔹	New to MATLA8? Watch this Video, see Demos, or read Getting Started.	3	×
All Files 🗠 Type	>> help inline		•
🗅 cdrom Folder	INLINE Construct INLINE object.		
cdrom0 Folder	TNI TNE(EXPR) constructs an inline function object from the	1	
□.hal-mtab HAL-MT	MATLAR expression contained in the string EVEP. The input		
	and the expression contained in the string tark. The input		
	arguments are automatically determined by searching EXPR		
	for variable names (see SYMVAR). If no variable exists, 'x'		
	is used.		
	INLINE(EXPR, ARG1, ARG2,) constructs an inline		
	function whose input arguments are specified by the		
	strings ARG1, ARG2, Multicharacter symbol names may		
	he used		_
	be used.		
Command Illiston	TNUTNE(EVPR N) where N is a scalar constructs an		
Command History II- I X	INLINE (EAFK, N), where N is a scalar, constructs an		
ex_04_0	initine function whose input arguments are 'x', 'P1',		
B % 3/2/10 10:00	P2',, 'PN'.		
help inline 루			
	E CONTRACTOR DE LA CONTRA		



Introc to Sc

Engir Com BIL

## INLINE FUNCTIONS

20 ·		MATLAB 7.6.0 (R2008a)
<u>File E</u> dit De <u>b</u> ug	Desktop <u>W</u> indo	w Help
🗋 🖆 👗 🐚 🛍 '	2 6 3 3	🖹 🥹 Current Directory /media 💌 🗔 🕤
Current Directory 🗰	□ ? × ₩▶	Command Window
🖻 🖆 🖪   🗟 🔹		New to MATLAB? Watch this <u>Video</u> , see <u>Demos</u> , or read <u>Getting Started</u> .
All Files 🔟	Type	be used.
⊇ cdrom0 ] .hal-mtab	Folder HAL-MT	<pre>INLINE(EXPR, N), where N is a scalar, constructs an inline function whose input arguments are 'x', 'P1', 'P2',, 'PN'. Examples: g = inline('t^2') g = inline('sin(2*pi*f + theta)') g = inline('sin(2*pi*f + theta)', 'f', 'theta') g = inline('x^P1', 1)</pre>
<ul> <li>Command History</li> <li>ex_04_6</li> <li>% 3/2/10</li> <li>help inl</li> </ul>	> +	See also <u>symvar</u> . Overloaded methods: <u>eml.inline</u> >>



## INLINE FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

EXAMPLE:

1 Convert degrees to radians

deg2rad=inline(' deg / 180 \* pi ', 'deg')

- 2 Calculate hypothenus
  - hyp = inline('sqrt(a<sup>2</sup> + b<sup>2</sup>)', 'a', 'b')



Intr

to S

and

## INLINE FUNCTIONS

auction	
rientific	
and the first state of the stat	

#### EXAMPLES;





### **INLINE FUNCTIONS**

Introduction to Scientific and Engineering Computing, BIL108E

### EXAMPLES;





## INLINE FUNCTIONS

Introduction to Scientific and **EXAMPLES:** Engineering Computing, BIL108E File Edit Debug Desktop Window Help 🞦 ظ 🐇 ங 🛅 🤊 🥲 🐌 🗊 🖹 🥹 Current Directory. /media - ... F Current Directory III a \* X W Com New to MATLAB? Watch this Video, see Demos, or read Getting Started. 🖻 🖆 🖪 🗟 🔸 All Files L Type >> hyp = inline('sqrt(a^2 + b^2)' ,'a', 'b') cdrom Folder ⊇cdrom0 Folder HAL-MT hyp = 🗋 .hal-mtab Inline function:  $hyp(a,b) = sqrt(a^2 + b^2)$ >> \* 🗆 \* hyp = inline('s^ clc hyp = inline('s

# And the second s

# INLINE FUNCTIONS

Introduction to Scientific and Engineering Computing, BIL108E

### EXAMPLES;



## SCRIPT M-FILES

#### Introduction to Scientific and

BIL108E

Engineering Computing, S

#### SCRIPT M-FILES

- Programs are contained in m-files. m-files are plain text files. Not binary files produced by word processors.
- File must have ".m" extension
- m-file must be in the path Matlab maintains its own internal path
- The path is the list of directories that Matlab will search when looking for an m-file to execute.
- Manually modify the path with the path, addpath, and rmpath built-in functions.



Introduction to Scientific and Engineering Computing, BIL108E

#### CURRENT DIRECTORY





## SCRIPT M-FILES

Introduction to Scientific and Engineering Computing, BIL108E



## SCRIPT M-FILES

Introduction to Scientific and Engineering Computing, BIL108E

#### **CREATE M-FILE**

MATLAB 7.610 (R2008a)	_ = []
Editor - unifiled	
Eile Edit Text Go Cell Tools Debug Desktop Window Help	5 K
1) 😂 🖩 👌 🍡 🟥 ヴ (*) 🍪 🗇 · 🛤 🏟 🏘 🏚 🏚 🖉 · 📾 🏶 👘 🏦 🛍 Stack: Base -	8088
** 🖫 - 1.0 +   ÷ 1.1 ×   🕸 % 🔍 (Step)	
script	Ln 1 Col 1 OV



## SCRIPT M-FILES

#### Introduction to Scientific and

#### path





Introduction to Scientific and Engineering Computing, BIL108E



# SCRIPT M-FILES

Introd

to Sci

Engin

Comp

BIL1

	MATLAB 7.6.0 (R200Ba)
<u>File Edit Debug</u>	Desktop Window Help
1 🖆 👗 🖷 🛍	🤊 🐏 📓 😰 🛛 Current Directory: /home/sept 🛛 👻 🖻
Current Directon	Command Window
Base 👻	>> help addpath
Name ∠	ADDPATH Add directory to search path.
	ADDPATH DIRNAME prepends the specified directory to the current
	matlabpath. Surround the DIRNAME in quotes if the name contains a
	space. If DIRNAME is a set of multiple directories separated by path
	separators, then each of the specified directories will be added.
	ADDPATH DIR1 DIR2 DIR3 prepends all the specified directories to
	the path.
	ADDPATHEND appends the specified directories.
	ADDPAINBEGIN prepends the specified directories.
	ADDPAINFRUZEN disables directory change detection for directorie
	heing added and thereby conserves Windows change
	being added and thereby conserves a moust change
Co III 7 X	notification resources (Windows only).
Co I+ □ ₹ × help f▲	notification resources (Windows only).
へ Co I+ □ z × help p help a	notification resources (Windows only).



### SCRIPT M-FILES

Introduction to Scientific and Engineering Computing, BIL108E



## SCRIPT M-FILES

#### Introduction to Scientific

and

#### rmpath





#### Introduction to Scientific and Engineering Computing, BIL108E





## SCRIPT M-FILES

Introduction	
to Scientific	
and	
Engineering	

Computing, BIL108E

#### SCRIPT M-FILES

- Not really programs
- No input / output parameters
- Script variables are part of workspace
- Useful for tasks that never change
- Use a script to run function for specific parameters required by the assignment

## SCRIPT M-FILES

Introduction to Scientific and Engineering Computing, BIL108E

EXAMPLE:

```
% SIMPLE SCRIPT FILE
theta = linspace (1.4, 4.6);
tandata = tan(theta);
plot (theta, tandata);
xlabel('\theta');
ylabel('tangent');
grid;
% File named as ex_05_03.m and run with
% the name of the filename without the ".m".
```

## SCRIPT M-FILES

#### Introduction to Scientific and

Engineering

Computin BIL108E

A	MATLAB 7.6.0 (R2008a)	ii:
Eile Edit Debug	Desktop Window Help	
106 8 10 10	🤊 💌 谢 🗊 🗐 🥹 Current Directory. /media/Transcend/source 💌 📖 🔞	
Current Directory ++	□ • × Wol Command Window	-0
🖻 🖆 🖪 🖥 🔹	>> type ex_05_03.m	
AIFIRE 2 delzero.m delzero.m dex.05.01.m ex.05.02.m ex.05.04.m dex.05.04.m dex.05.06.m fact.m FtoC.m tot.m testvar.m destvar.m commad History type ex. cl c	<pre>M-File M-fi</pre>	
▲ <u>S</u> tart		



Introduction to Scientific and Engineering Computing BIL108E

#### EXAMPLE:





## SCRIPT M-FILES

#### Introduction to Scientific and Engineering

Computing, BIL108E

#### SCRIPT M-FILES

- All variables created in a script file are added to the workplace.
- This may have undesirable effects because variables already existing in the workspace may be overwritten

## **FUNCTION M-FILES**

Introduction to Scientific and Engineering Computing, BIL108E

FUNCTION M-FILES

- Matlab has many built-in functions.
- Use type *functionname* to verify the functions.
- Function m-files differ from a script file in that it communicates with the MATLAB workspace only through specially designated **input** and **output** arguments.

## **FUNCTION M-FILES**



BIL108E

#### FUNCTION M-FILES

- Functions use input and output parameters to communicate with other functions and the command window
- Functions use local variables that exist only while the function is executing. Local variables are distinct from variables of the same name in the workspace or in other functions.
- Input parameters allow the same calculation procedure (same algorithm) to be applied to different data. Thus, function m-files are reusable.
- Functions can call other functions.



Introduction to Scientific and Engineering Computing, BIL108E

#### BASIC RULES

... outarg2;

## FUNCTION M-FILES

EXAMPLE:

Introduction to Scientific and Engineering Computing, BIL108E

1	MATLAB 7.6.0 (R2008a)	
<u>File Edit View Graphics Debu</u>	g Desktop Window Help	
🗋 🧉 🛦 🐃 🛅 🤊 (° ) 🚵 🖻	🖞 🗐 🥹 Current Directory. /media/Transcend/source 👻 😰	
Directory Workspace += D 7 >	Command Window	** 🗆 * ×
Name ∠ Value Handata <1×100 dc theta <1×100 dc	<pre>&gt;&gt; type FtoC.m function C=FtoC(F) % Celsius=FtoC(Fahrenheit) % Converts Fahrenheit temperatures to Celsius C=5*(F-32)/9; &gt;&gt; help FtoC Celsius=FtoC(Fahrenheit) Converts Fahrenheit temperatures to Celsius</pre>	
Command History	>>	
<u> ≪ S</u> tart		



## FUNCTION M-FILES

#### Introduction to Scientific and Engineering

Computing,

BIL108E

#### EXAMPLE:

 Write a function called FtoC (FtoC.m) to convert Fahrenheit temperatures into Celsius.

function C=FtoC(F)
% Celsius=FtoC(Fahrenheit)
% Converts Fahrenheit temperatures to Celsius
C=5\*(F-32)/9;

## FUNCTION M-FILES

oduction	
cientific	

to S



#### SUMMARY OF INPUT AND OUTPUT ARGUMENTS

- Values are communicated through input arguments and output arguments.
- Variables defined inside a function are local to that. function I ocal variables are invisible to other functions. and to the command environment.
- The number of return variables should match the number of output variables provided by the function.



## **FUNCTION M-FILES**

#### Introduction to Scientific and Engineering Computing,

BIL108E

#### SCOPE

#### LOCAL VARIABLES

- Any variable defined inside a function is inaccessible outside it
- Such variables are referred to as local. They exist only inside the function, which has its own workspace separate from the base workspace of variables defined in the Command Window



## **EUNCTION M-FILES**

Introduction to Scientific and Engineering Computing, BIL108E

### SCOPE

#### **GLOBAL VARIABLES**

- Variables defined in the base workspace are not normally accessible inside functions.
- Their scope is restricted to the workspace itself unless they have been declared global: global VARIABLENAME
- If several functions, and possibly the base workspace, declare particular variables as global, then they all share single copies of them.



## **FUNCTION M-FILES**

#### Introduction to Scientific and

Engineering

Computing, BIL108E

#### SCOPE

GLOBAL VARIABLES cont'd.

- Matlab recommends that global variables be typed in capital letters to remind you that they are global.
- The function isglobal(VARNAME) returns 1 if A is global, and 0 otherwise.
- The command who global gives a list of global variables.
- The command clear global makes all variables nonglobal. Example: clear VARNAME makes VARNAME nonglobal.

#### PERSISTENT VARIABLES

Persistent variables remain in existence between function calls. Example:

function test
persistent count
if isempty(count)
count = 1
else
count = count + 1
end

# THE STATE OF STATE OF

## FUNCTION M-FILES

Current Directory [/media/Transcend/source
Current Directory /media/Transcend/source 💌 😰
nd Window 🕴 🗆 🕹
/pe test.m
A set associate
tion test
istent count
sempty(count)
count = 1
count = count + 1
· · ·
-
1
1

## FUNCTION M-FILES

Introduction to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:

±A.		MATLAB: 7.6.0 (R2008a)	
<u>File Edit Debug</u>	g <u>D</u> esktop <u>W</u> inde	ow Help	
1 6 8 1 6	200 300	🖹 🥝 Current Directory. /media/Transcend/source 👻 🛄 🔞	
Directory Worl	kspace 🖛 🗖 🛪 🗙	Command Window	× * 🗆 *
	- 8ase - *	>> test	-
Name ∠ ⊞ans ⊞tandata	Value 37.2222 <1x100 dot	count =	
<b>⊞</b> theta	<1×100 doi	1	
		>> test	
		count =	
		>> test	
Command History	× 5 0 +1	count =	
-clc -type te	est.m	3	
▲ <u>Start</u>		>>	0



## FUNCTION M-FILES

#### Introduction to Scientific and Engineering Computing, BIL108E

#### FUNCTIONS WITHOUT RETURN VALUE

Omit the equal sign and output arguments in the function definition line.

EXAMPLE:

function stars(n)
asterisk = char(abs('\*')\*ones(1,n));
disp(asterisk)



Introduction to Scientific and Engineering Computing, BIL108E





and

## FUNCTION M-FILES

Introduction to Scientific EXAMPLE: Engineering Computing, BIL108E File Edit Debug Desktop Window Help Workspace 🕶 🗖 🛪 🗙 >> type stars Name ∠ Value 🗄 ans 42 <1x100 do 🖽 tandata <1×100 do H theta disp(asterisk) >> stars(9) \*\*\*\*\*\*\* >> abs('\*') ans =

### ATLAB 7.6.0 (R2008a) 🎦 ظ 👗 🐂 🛅 🤊 (\* ) 🏟 💅 🖹 🕜 Current Directory, /media/Transcend/source 💌 ... 😰 function stars(n) asterisk = char(abs('\*')\*ones(1,n)); 42 Command History \* 5 7 X type stars stars(9) abs('\*') **▲** Start

Introduction to Scientific and Engineering Computing, BIL108E

## FUNCTION M-FILES

## EXAMPLE:

A	MATLABI 7.6.0 (R2008a)	
<u>File Edit Debug D</u> eskto	b <u>Window H</u> elp	
10 B # 10 0 0	🐌 🗊 🗐 🖉 Current Directory. /media/Transcend/source 👻 🛄 🔞	
◀ Directory Workspace ₩	Command Window	× 5 🗆 **
🛅 📷 😓 🍇 🔤 - Base	- »	-
Name ∠ Value	function stars(n)	
👪 ans 🛛 '*'	asterisk = char(abs('*')*ones(1,n));	
🗄 tandata <1×1	<sup>00 dou</sup> disp(asterisk)	
theta <1×1	<sup>00 dol</sup> >> stars(9)	
	*****	
	>> abs('*')	
	ans =	
		1001
	42	
	24	
	sy char(are)	
	>> chai (ans)	
4		
Command History 🗰		
stars(9)		
abc(1*1)	×	
char(ans)	• >>	
📣 Start		014



## **FUNCTION M-FILES**

roduction
Scientific
and
ginooring

Computing, BIL108E

#### **VECTOR ARGUMENTS**

- Input and output arguments can be defined as vectors.
- A vector is initialized each time the function is called.

#### EXAMPLE:

dice function generates a vector of n random rolls of a die.

function d= dice(n) d = floor(6 \* rand(1, n) + 1);



Introduction to Scientific and Engineering Computing, BIL108E

	MATLAB 7.6.0 (R2008a)	
Eile Edit Debug Desktop Winde	w Help	
) 🖆 🖌 🍡 🗂 🤊 (* ) 🏜 🖆	🖹 🥝 Current Directory: /media/Transcend/source 💌 🛄 🔞	
Directory Workspace 🏎 🗖 🔻 🗙	Command Window	** 🗆 * ×
🗃 📷 🔚 🍇 🚾 - Base 🔹 👋	>> type dice.m	
Jame ∠ Value		
ans '*'	function d= dice(n)	
tandata <1×100 doi	d = floor(6 * rand(1, n) + 1):	
Etheta <1X100 dot	>>	
Command History 🗰 🖬 🛪 🗙		
char (ans)		
clc		
-clc		



## FUNCTION M-FILES

Introduction to Scientific and EXAMPLE: Engineering Computing, BIL108E Eile Edit Debug Desktop Window Help 🎦 ظ 👗 ங 🛱 🤊 🥲 🐞 💅 🗐 🕢 Current Directory. /media/Transcend/source 🕶 ... 🔞 ◀ Directory Workspace ₩ □ ₹ × 🐚 🖬 🕼 🙀 🚾 - Base >> >> type dice.m Name ∠ Value [5,6,1] <1×100 doi 🗄 ans function d= dice(n) tandata theta d = floor(6 \* rand(1, n) + 1); <1×100 dou >> dice(3) ans = 5 61 >> 4 Command History III 7 X clc type dice.m dice(3)

## FUNCTION M-FILES

Introduction to Scientific Engineering Computing, BIL108E

#### EXAMPLE:

<b>A</b>	MATLAB 7.6.0 (R200Ba)	
Eile Edit Debug Desktop	<u>Window</u> Help	
🗋 🖆 🖉 🖉 🖉 🖉	👔 💼 🥹 Current Directory: /media/Transcend/source 👻 🛄 🔞	
◆ Directory Workspace I □	X Command Window	- 「 
1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	<pre>&gt;&gt; type dice.m</pre>	
ans 4	function d= dice(n)	
Heta <1x100	do d = floor(6 * rand(1, n) + 1);	
	>> dice(3)	
	20.10	
	ans =	
	5 6 1	
	>> floor(4.7)	
	ans =	
•		
Command History	4	
tuna di ca m		
cype dice.m	>>	
arce(3)		
tloor(4.7)		
🚸 <u>S</u> tart		0

📣 <u>S</u>tart

## FUNCTION M-FILES

Introduction	
to Scientific	
and	

Engineering

Computi

Bill 108E       Bile Days Deston Window Help         Karaman       Image:	1	MATLAB 7.6.0 (R2008a)	A	Computing,
Karaman       Image: Imag		w Help	<u>File Edit Debug Desktop Windo</u>	BIL108E
Command Window       >> dice(3)         Name        Value         ans       4         tandata       <1x100 dot         theta       <1x100 dot         5       6         1       >> floor (4.7)         ans =       4         >> floor (4.2)         ans =       4         ans =       4         >> floor (4.2)       ans =		🗊 🥹 Current Directory /media/Transcend/source 🕶 😰	1) 🖆 🛦 🐂 🖻 🤊 (e) 🚵 🚮	Karaman
>>> dice(3) ans = tandata <1x100 dot theta <1x100 dot theta <1x100 dot >> floor (4.7) ans = 4 >> floor (4.2) ans =	×	Command Window 🔤 🖉		Naraman
Name // Value       ans =         ans 4       5       1         brandata       <1x100 dot       5       1         >> floor (4.7)       ans =       4          4       >> floor (4.2)       ans =          4       >> floor (4.2)       ans =           4       >> floor (4.2)         ans =       4       >> floor (4.2)       ans =	-	>> dice(3)	🛅 📷 🔚 🍇 📷 - Base 🔹 🛸	
ans       4         anstandata       <1x100 dot         theta       <1x100 dot         5       6         y       Floor (4.7)         ans =       4         y       floor (4.2)         ans =       4         y       floor (4.2)         ans =       4			Name 🗠 🛛 Value	
Image: command History       **       Floor (4. 2)         Image: command History       **       **		ans =	🖽 ans 🛛 4	
<pre></pre>			🗄 tandata <1×100 doi	
<pre>&gt;&gt; floor(4.7) ans =</pre>		5 6 1	theta <1×100 doi	
>> floor(4.7) ans = 4 >> floor(4.2) ans =				
ans = 4 >> floor(4.2) ans =		>> floor(4.7)		
ans = 4 >> floor(4.2) ans =				
4 >> floor(4.2) ans =		ans =		
<pre>&gt;&gt; floor(4.2) ans =</pre>		4		
<pre>&gt;&gt; floor(4.2) ans =</pre>		1		
Command History III III X	-	>> floor(4.2)		
ans =				
		ans =	Command History III 7 X	
			dice(3)	
+		4	floor(4,7)	
			floor(4.7)	
······································	-	>>	-1100r(4.2)	
A Start	0		📣 Start	

#### PASSING AN ARGUMENT

- Passing an argument by value: An input argument is passed by value only if a function modifies it.
- Passing an argument by reference

function y = delzero(x)y = x(x = 0);

% x = delzero(x)

## FUNCTION M-FILES

Introduction to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:





## FUNCTION M-FILES

Introduction to Scientific and Engineering	EXAMPLE:
Computing,	MATLAB 7.6.0 (R2008a)
DILIUOE	Elle Edit Debug Desktop Window Help
Karaman	🗋 🗃 👗 🖏 🛍 🤊 (*) 🏟 📆 🛃 🥥 Current Directory (media/Transcend/source 💌 🛄 😨
	<pre>     Command History ** C **     Command History ** C **     Floor(4.2)     Command History ** C **     Floor(4.2)     Clc     the datagram are are are are are are are are are are</pre>

## FUNCTION M-FILES

## Introduction

to Scientific

and

Engineering

Computing,

BIL108E

#### **EXAMPLE:**

📣 <u>S</u>tart

# File Edit Debug Desktop Window Help

The set y man			
		Composed Window	
	· Base · »	>> type delzero.m	
tandata tandata theta	[2,4,9,12] <1×100 dou <1×100 dou [2,4,9,12]	<pre>function y = delzero(x) y = x(x ~= 0); % x = delzero(x) &gt;&gt; x = delzero(x) x =</pre>	
		2 4 9 12 >>	
Command History -clc -type de x = del	Izero.m           zero(x)		
📣 Start			

MATLAB 7.6.0 (R2008a)



### NUMBER OF ARGUMENTS OF A FUNCTION

Use nargin and nargout functions to display the number of input and output arguments.

EXAMPLE:

function y = myfunc1(a, b, c)
disp(nargin)
y = nargin;



## FUNCTION M-FILES

Computing,	A MATLAB 7.6.0 (R2008a)	ie.
BIL108E	Elle Edit Debug Desktop Window Help	
Karaman	🗋 🖆 👗 🐂 🛱 🤊 (*   🐌 🗊 🖹 🔮 Current Directory /media/Transcend/source 👻 🖻	
rtaraman	Orrectory Workspace **      Command Window	- 1
	<pre>&gt;&gt; type myfunc1.m</pre>	
	Name 4 Value	
	$\frac{1}{2} tandata < 1 \times 100 \text{ dot}$	
	theta <1×100 dou disp (nargin)	
	$\exists x = [2,4,9,12]$ $y = nargin;$	
	2	
	ans =	
	2	



#### Introduction to Scientific and Engineering Computing, BIL108E

## FUNCTION M-FILES

#### SUBFUNCTIONS

- An M-file may contain the code for more than one function.
- The first one in the file is the **primary function** and is invoked with the M-file name.
- Additional functions are called **subfunctions** and are visible only to the primary function and to other subfunctions.
- Each subfunction begins with its own function definition line.
- Subfunctions follow each other in any order after the primary function.



## **RECURSIVE FUNCTIONS**

roduction	
Scientific	
and	

Computing, BIL108E

#### **RECURSIVE FUNCTIONS**

MATLAB allows functions to call themselves in a process called recursion.

Recursive function can cost too much time and memory.

#### Example:

start

Factorial function can be written in a recursive ".m" file.

$$n! = n \times (n-1)!$$



## **RECURSIVE FUNCTIONS**

Introduction to Scientific and Engineering Computing, BIL108E

EXAMPLE:

function y = fact(n)% Factorial Recursive definition of n! disp(n) if n > 1y = n \* fact(n-1);else y = 1; end



## **RECURSIVE FUNCTIONS**

Introduction to Scientific and EXAMPLE: Engineering Computing, MATLAB 7.6.0 (R2008a) BIL108E File Edit Debug Desktop Window Help 🎦 ظ 👗 🐂 🛅 🤊 (\* ) 🏟 💅 🖹 🕜 Current Directory, /media/Transcend/source 💌 ... 😰 Workspace 🕶 🗖 🛪 🗙 >> type fact.m Name ∠ Value 🗄 ans function y = fact(n)<1×100 do 🖽 tandata % Factorial Recursive definition of n! <1×100 doi [2,4,9,12] **H**theta disp(n) HV if n > 1 y = n \* fact(n-1); else = 1: v end Command History 🗰 🖬 🛪 🗙 myfunc1(1,2) clc type fact.m **▲** Start

## **RECURSIVE FUNCTIONS**

Introduction to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:

A		MATLAB 7.6.0 (R2008a)	•
<u>File Edit Debug D</u> esktop	Window	Help	1
1 6 × 1 2 0 0	🍋 🗹 🖻	🛿 Current Directory: /media/Transcend/source 👻 🛄 🛐	
	2 × 5 [	ommand Window	··· □ ㅎ ×
🛅 📷 🝓 🝇 🚾 - Base 👒	-	> fact(7)	-
Name 🗠 🛛 Value		7	
ans 5040			
tandata <1×10	0 dou	6	
theta <1x10	121		
L2,4,0,	121	5	
		4	
		3	
		2	
		1	
1			
Command History III	X		
clc	Ê a	ns =	
type fact.m			
fact(7)		5040	-
A Start			101



## **FUNCTION HANDLES**

#### Introduction to Scientific and Engineering Computing, BIL108E

#### FUNCTION HANDLES

Function handles points the defined function.

- A handle for a function is created with @. A function may be represented by its handle. In particular, the handle may be passed as an argument to another function.
- feval evaluates a function whose handle is passed to it as an argument.



## FUNCTION HANDLES

#### Introduction to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:

function myplotfunc(funchand, limit1, sample)
% plot the given function
% with the given limits
x = linspace(limit1(1), limit1(2),sample);
y = feval(funchand, x);
plot(x,y)

#### % fhand = @sin % myplotfunc(fhandle, [0, pi/2], 100) % fhand = @cos % myplotfunc(fhandle, [0, pi/2], 100)

# ALL AND AL

## FUNCTION HANDLES

Introduction to Scientific and EXAMPLE: Engineering Computing, MATLAB 7.6.0 (R2008a BIL108E File Edit Debug Desktop Window Help 🎦 ظ 👗 🐂 🛅 🤊 (\* ) 🐉 🛒 🗐 🕜 Current Directory: /media/Transcend/source 💌 ... 💽 Current Directory 🏎 🖬 🛪 🗙 🐨 🖻 🖆 🖪 🖥 🔹 • >> type myplotfunc.m All Files L Type ex\_05\_01.m M-file function myplotfunc(funchand, limit1, sample) ex\_05\_02.m M-file % plot the given function 🖺 ex\_05\_03.m M-file % with the given limits 🖺 ex\_05\_04.m M-file x = linspace(limit1(1), limit1(2), sample); ex\_05\_05.m M-file ]ex\_05\_05.m~ Editor y = feval(funchand, x);ex\_05\_06.m M-file plot(x,y) ) ex\_05\_06.m~ Editor fact.m M-file % fhand = @sin FtoC.m M-file % myplotfunc(fhandle, [0, pi/2], 100) 1 myfunc1.m M-file myplotfunc.m M-file % fhand = @cos M-file 1 stars.m % myplotfunc(fhandle, [0, pi/2], 100) M-file test.m \* • • × ommand Hist clc type myplotfunc **▲** Start

### FUNCTION HANDLES

#### Introduction to Scientific and Engineering Computing, BIL108E

# ring EXAMPLE:

File Edit Debug Desktop Window Help 🚹 🖆 🔏 🖻 🤊 🍽 谢 🗊 🖹 🥘 Current Directory: /media/Transcend/source 💌 ... 😰 Current Directory I 🖬 🔻 🗙 Work 🖻 🗂 🖪 🖥 🔹 🔹 >> fhand = @sin All Files L Type 🖺 ex\_05\_01.m M-file fhand = 🖺 ex\_05\_02.m M-file ex\_05\_03.m M-file @sin M-file ex\_05\_04.m M-file 🖺 ex\_05\_05.m >> myplotfunc(fhand, [0, pi/2], 100) ex\_05\_05.m~ Editor 4 Mex 05 06.m M-file >> ex\_05\_06.m~ Editor A fact.m M-file FtoC.m M-file 1 myfunc1.m M-file M-file myplotfunc.m M-file 🖺 stars.m test.m M-file \* 5 7 × CIC fhand = @sin myplotfunc(fhand A Start

# The second secon

## FUNCTION HANDLES

#### Introduction to Scientific and

Engineering

Computing,

BIL108E





## FUNCTION HANDLES

Introduction to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:

File Edit Debug I	Decktop Mind	au Lisia	
Elle Edit Debug E	Zeskipp Mind		
	? (*   🍋 🗹	💼 🥝 Current Directory /media/Transcend/source 💌 🛄 🖻	
Current Directory + [	4 IOW X 5 🗆	Command Window	
🖻 🖆 🖪 💆 •		>> fhand = @sin	
All Files ∠	Type		
<pre> ex_05_01.m </pre>	M-file	fhand -	
<pre>ex_05_02.m</pre>	M-file	Thand -	
🖺 ex_05_03.m	M-file		
🖺 ex_05_04.m	M-file	Øsin	
<pre>ex_05_05.m</pre>	M-file		
ex_05_05.m~	Editor A	>> mvplotfunc(fhand, [0, pi/2], 100)	
🖺 ex_05_06.m	M-file	$\rightarrow$ fhand = 0 cos	
ex_05_06.m~	Editor A		
🖺 fact.m	M-file		
🖺 FtoC.m	M-file	Thand =	
🖺 myfunc 1.m	M-file		
1 myplotfunc.m	M-file	Øcos	
🕙 stars.m	M-file 🚽		
💼 test.m	M-file 💌	>> mynlotfunc(fhand [0 ni/2] 100)	
<ul> <li>■ 30002</li> </ul>		// myprocrane((nana, [0, p1/2], 100)	
Command History	× 5 🗆 +1	>>	
тургостип	c(Tnanc.		
fhand = @	cos		
mynlotfun	c (fhan		
a myprocrum	c (mante		
	1.63		



## **FUNCTION HANDLES**

Introduction to Scientific

and

BIL108E



## FUNCTION HANDLES

Introduction to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:



## **FUNCTION HANDLES**

Introduction to Scientific and

Engineering

Computing,

BIL108E





#### Introductio to Scientific and Engineering Computing, BIL108E

#### EXAMPLE:

The function sin(x) can be written as a Taylor series by

$$sin(x) = \sum_{k=1}^{\infty} \frac{(-1)^k x^{2k+1}}{(2k+1)!}$$

Write a user-defined function file that calculates sin(x) by using Taylor's series. For the function name and arguments use y = Tsin(x, n). The input arguments are the angle x in degrees, and n the number of terms in the series. Use the function to calculate  $sin(150^\circ)$  using 3 and 7 terms.



Introduction to Scientific

and

Engineering

Computing, BIL108E

## **STRUCTURES**

#### **STRUCTURES**

Arrays can store variables that may be all numeric or character. With structure different data types can be stored as one variable within a structure.

#### EXAMPLE:

Create a structure called student with one field for a student's name, a second for his/her student ID number, and a third for all her marks to date.

student.name = 'Can Ozgur'; student.id = 'N010080090'; student.marks=[80, 60, 40]; % student % student.marks(2)

## **STRUCTURES**

Introduction to Scientific and Engineering Computing, BIL108E

### **STRUCTURES**

Use subscripts to add more elements EXAMPLE:

student(2).name = 'Ergun Yilmaz'; student(2).id = 'N010080091'; student(2).marks=[70, 30, 90]; % student % student(2).marks(2)



to S

Cor

## **STRUCTURES**

cientific and neering	EXAMP
109E	A
-109E	<u>Eile Edit Debug I</u>
	1 6 × h 6 ×
	Orrectory Workspa
	•
	Command History
	clear

4	MATLAB 7.6.0 (R2008a)	
Eile Edit Debug Desktop Win	dow Help	
🗋 🧉 🛦 🖷 🛍 🤊 (° ) 🏟 🖆	🛿 🗾 🥝 Current Directory: /media/Transcend/source 👻 🗔 🔞	
	Command Window	** 🗆 *
🛅 📷 🗃 🍇 🌆 - Base 🔹 🔺	>> type ex_05_05	
Name ∠ Value		
	student.name = 'Can Ozgur';	
	student.id = 'N010080090';	
	student.marks=[80, 60, 40];	
	% student	
	% student.marks(2)	
	>> ex_05_05	
Command History +		
clear		
CIC		



## **STRUCTURES**

Introduction to Scientific and Engineering Computing, BIL108E





and

BIL108E

## **STRUCTURES**

Introduction to Scientific EXAMPLE: Engineering Computing, Eile Edit Debug Desktop Window Help 🎦 😂 👗 ங 🖏 🤊 🤨 🐞 💅 🖹 🥥 Current Directory. /media/Transcend/source 🕶 🗔 🔞 Workspace It D 7 X Command

1 1 2 1 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2	- Base - >>	% student	
Name ∠	Value	% student.marks(2)	
🗄 ans	80		
🗄 student	<1×1 struct	>> ex 05 05	
		>> student	
		student =	
		name: 'Can Ozgur'	
		id: 'N010080090'	
		marks: [80 60 40]	
		>> student.marks(1)	
4		ans =	
Command History	X 5 🗆 🕂		
ex_05_05	5 🗎	80	
student	100		
-	(a)*	>>	
A Start			10

## **STRUCTURES**

Introduction to Scientific and Engineering Computing, BIL108E

## EXAMPLE:



# **STRUCTURES**

#### Introduction

to Scientific

and





## STRUCTURES

Introduction to Scientific and Engineering Computing, BIL108E





## STRUCTURES

BE Elle an C C Name B ar El st	Edit Debug Desktop Win b & 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	dow Help  P Q Current Directory [/media/Transcend/source : Command Window  name i d	** = x ×
IN Contraction of the second s	3         3         10         10         10         10           Cotory         Workspace         #         1         2         2           1         10         10         10         2         2           2         Value         2         90         2	Image: Second	× 5 🗆 10
Name ∎ ar ∎ st	Workspace     Image: The state sta	Command Window  Command Windo	** 🗆 * ×
™∎ Name ⊞ar ≝st	Image: Second	name id	-
Name Bar Bast	∠ Value s 90	id	
ar E st	s 90	manka	
te st		marks	
	Jaent <1x2 struc		
		<pre>&gt;&gt; student(2)</pre>	
		ans =	
		name: 'Ergun Yilmaz'	
		id: 'N010080091'	
		marks: [70 30 90]	
		terioda un prisonale - Dagemario - Hansini, Kasha Murtu	
		<pre>&gt;&gt; student(2).marks(3)</pre>	
4		ans =	
Com	nand History 🗰 🖬 🛪 🗙		
	student		



## M-FILE DEBUGGING

Introduction to Scientific and Engineering Computing, BIL108E

### M-FILE DEBUGGING

The Editor / Debugger enables you to get inside a function, while it is running.



## References

Introduction to Scientific and Engineering Computing, BIL108E



#### References for Week 5

**1** Brian Hahn, Daniel T.Valentine, Essential Matlab for Engineers and Scientists, Elsevier, 2010.