Introduction to Matlab ECE 111 Introduction to ECE Week #1

Please visit Bison Academy for corresponding lecture notes, homework sets, and solutions

Becoming familiar with MATLAB

- Using the command window
- Using scripts
- Plotting with Matlab
- Random numbers in Matlab
- If-Statements
- For-Loops
- While-Loops
- Monte-Carlo Simulations



General environment and the console

Startup Screen:

• I usually close everything down except the command window



Command Window

Matlab works like a calculator

- Solve
- (2+3)*5
- Solve

```
Y = 5 * \cos(10 * \tan(3))
```

You type it the way it looks:

-# N	MATLAB 7.12.0 (R2011a)
File	Edit Debug Desktop Window Help
: 🎦	🚰 👗 ங 🛍 🤊 🥲 🖓 😭 👔 🗊 📄 🎯 Documents and Settir
: Sh	ortcuts 🖪 How to Add 🛛 💽 What's New
	>> (2 + 3) * 5
	ans =
	25
	>> Y = 5*cos(10*tan(3))
	Υ =
	0 7241
	0.7241
fx	>>
J.÷	

Matlab Function Names

π

 e^{x}

10^{*x*}

ln(x)

 $\log_{10}(x)$

 $\cos(x)$

tan(x)

 $\arcsin(x)$

 $\arccos(x)$

 $\arctan(y/x)$

sin(x) units = radians

11

11

"

"

- pi
- exp(x)
- 10^x
- log(x)
- log10(x)
- sin(x)
- $\cos(x)$
- tan(x)
- asin(x)
- acos(x)
- atan(y,x)
- + many more

File	Edit Debug Desktop Window Help
1) 🚰 👗 ங 🛍 🤊 🥲 à 📸 🛃 🥝 C:\Documents and Settings
sŀ	ortcuts 🛃 How to Add 💽 What's New
	>> exp(2)
	ans =
	7.3891
	>> log(10)
	10 10g(10)
	ans =
	2.3026
	>> Sin(1)
	ans =
	0.8415
	>> cos(2)
	2019 =
	ans –
	-0.4161
	>> asin(0.6)
	ans =
	0.6435
	0.0100
fx	>>
4 9	itart

Order of Operations

- 1st: ^
- 2nd: *,/
- 3rd: +, -

Equations are executed

- By order of operations, then
- Left to right

Paranthesis never hurt,

• They can aviod confusion

*	MATLAB 7.12.0 (R2011a)
File	Edit Debug Desktop Window Help
: *) 🚰 👗 ங 🛍 🤊 (🔍) ቅ 📝 🖹 🥝 C:\Documents and Settings\Ad
: Sh	ortcuts 🗷 How to Add 🛛 🗷 What's New
	>> A1 =4/2*3
	A1 =
	6
	>> A2 = $4/2/3$
	A2 =
	0.6667
	>> A3 = 4 / $(2*3)$
	A3 =
	0.6667
fx.	>>

Matlab as a Calculator

You can define variables as you go

- 1st character must be a letter
- 2nd onward can be letters or numbers
- Case sensitive

Example: Light Sensor

$$R = \left(\frac{2200}{(lux)^{0.6}}\right)\Omega$$
$$V = \left(\frac{R}{1000+R}\right)10V$$

Find R and V @ 10 Lux

*	MATLAB 7.12.0 (R2011a)
File	Edit Debug Desktop Window Help
: "] 🔁 👗 🖷 🖷 🤊 (♥ 🍑 🗊 🗐 🎯 ettings\Administr
: Sh	ortcuts 🛃 How to Add 🛛 Mhat's New
	>> lux = 10
	lux =
	10
	>> R = 2200 / (lux ^ 0.6)
	_
	R =
	552 6150
	552.6150
	>> $V = R / (1000 + R) \times 10$
	··· · · · · · · · · · · · · · · · · ·
	V =
	3.5593
fx;	>>

Doing Several Operations at Once

Matlab is a matrix language

[start of a matrix , next column (a space also works) ; next row] end of matrix. ' transpose

Multiplication, Power

- ◆ *, ^
- Matrix operations (coming soon)

Dot-Notation

- .* ./ ,^
- Element-by-element operations
- Allows you to do several operations at once

-	MATLAB 7.12.0 (R2011a)
File	Edit Debug Desktop Window Help
: *) 🚰 🐰 ங 🛍 🤊 (* 🐉 🗊 🖹 🕐 ettings\Administrator\My Do
: st	ortcuts 🗷 How to Add 💽 What's New
	>> lux = [10,20,30]'
	lux =
	10
	30
	>> R = 2200 ./ (lux .^ 0.6)
	R =
	552.6150
	364.5899
	285.8577
	>> V = R ./ $(1000 + R) \times 10$
	V =
	3.5593
	2.6718
	2.2231

Formatting Output

- Terminate a line with a semi-colon if you don't what the result displayed
- Leave off the semi-colon if you *do* want to see the result

format short pi

3.1416

format long pi

3.141592653589793

format shorteng
pi^30

821.2893e+012

```
📣 MATLAB 7.12.0 (R2011a)
File Edit Debug Desktop Window Help
🎦 🚰 👗 🐂 🛱 🤊 (🖻 🦣 🛒 🗐 🖉 ettings\Administrator\My
 Shortcuts 💽 How to Add 🛛 💽 What's New
  >> lux = [10,20,30]';
  >> format long
  >> R = 2200 ./ (lux .^ 0.6)
  R =
     1.0e+002 *
      5.526150149321077
      3.645899419073986
      2.858577216671873
  >> format short e
  >> V = R ./ (1000 + R) * 10
  V =
     3.5593e+000
     2.6718e+000
     2.2231e+000
fx >>
```

Matlab as a Graphing Calculator

Matlab has pretty good graphics This is useful if you want to know what happens over a range of values.

Example: Find R for 1 < lux < 100

 $R = \left(\frac{2200}{(lux)^{0.6}}\right)\Omega$

Linear spacing

Log spacing from 10^{-2} to 10^{3} with 100 points lux = logspace(-2, 3, 100)';



Matlab as a Graphing Calculator

Graphs are useful

- They show you how the two variables are related
- They allow you to determine V over a range of lux
- They allow you to determine lux if you know V

Example: If you read 2.00 Volts, what is the light level?

- Read it off the graph
- light = 36 lux (approx)

1	AATL/	AB 7.12.0 (R2011a)
File	Edit	Debug Desktop Window Help
Ľ) 🔁	👗 ங 💼 🤊 😢 巐 🗊 🖹 🕜 🗦 jettings\Administrator\My Documents\MATLAB
sh	ortcut	s 🗷 How to Add 🗷 What's New
	>>	lux = [1:0.1:100]';
	>>	R = 2200 ./ (lux .^ 0.6);
	>>	V = R . / (1000 + R) * 10;
	>>	<pre>plot(lux,V);</pre>
	>>	<pre>xlabel('lux');</pre>
	>>	<pre>ylabel('Volts');</pre>
	>>	ylim([0,7])
fre	>>	grid
A.		
	otart	
	Figur	e 1 💼 🚺
File	Edit	View Insert Tools Desktop Window Help
2	6	a 🖕 📐 🔍 🖤 🕲 🐙 🔏 - 🗔 🔲 📰 💷
		7
		6
		6
		4
	olts	
	>	3
		2





Matlab Help

If you forget how to use a function, type help

```
📣 MATLAB 7.12.0 (R2011a)
File Edit Debug Desktop Window Help
🎦 🚰 🐇 🐚 👘 🤊 💌 훩 📆 📄 🥝 Current Folder: C:\Documents and Settings\Administrator\My Documents\MATLAB
Shortcuts 🛃 How to Add 🛛 🛃 What's New
  >> help plot
   PLOT Linear plot.
      PLOT(X,Y) plots vector Y versus vector X. If X or Y is a matrix,
      then the vector is plotted versus the rows or columns of the matrix,
      whichever line up. If X is a scalar and Y is a vector, disconnected
      line objects are created and plotted as discrete points vertically at
      х.
      PLOT(Y) plots the columns of Y versus their index.
      If Y is complex, PLOT(Y) is equivalent to PLOT(real(Y), imag(Y)).
      In all other uses of PLOT, the imaginary part is ignored.
      Various line types, plot symbols and colors may be obtained with
      PLOT(X, Y, S) where S is a character string made from one element
      from any or all the following 3 columns:
                    blue
                                          point
                                                                     solid
              b
                                    .
                                          circle
                                                                     dotted
                                                              :
              q
                    green
                                   0
              r
                    red
                                   Х
                                         x-mark
                                                              -.
                                                                     dashdot
              С
                                   +
                                          plus
                                                              ___
                                                                     dashed
                    cyan
                    magenta
                                   ×
                                          star
                                                            (none) no line
              m
```

Multiple Plots on the same graph:

plot(x1,y1, x2,y2, x3,y3)
plot(x, [y1,y2,y3])

Invest \$1000 for 10 years at...

- 1% interest
- 3% interest
- 6% interest
- 9% interest

```
t = [0:0.01:10]';
y1 = 1000 * exp(0.01*t);
y3 = 1000 * exp(0.03*t);
y6 = 1000 * exp(0.06*t);
y9 = 1000 * exp(0.09*t);
```

```
% Method #1
plot(t,y1,t,y3,t,y6,t,y9)
```

```
% Method #2
plot(t,[y1,y3,y6,y9])
```



Polynomials

poly([a,b,c])

- Give a polynomial with roots at (a, b, c) roots([a,b,c,d])
 - Find the roots of the polynomial

 $ax^3 + bx^2 + cx + d = 0$

poly([1,2,3])

1 -6 11 -6

$$y = x^3 - 6x^2 + 11x - 6$$

= (x - 1)(x - 2)(x - 3)

roots([1,-6,11,-6])

3.0000 2.0000 1.0000

```
📣 MATLAB 7.12.0 (R2011a)
File Edit Debug Desktop Window Help
 🎦 🚰 🐰 🐂 違 🤊 🕲 🐌 📆 📄 🕜 🕽 Jocuments and Settings\Administrator\My Documents\ 🔽 🛄 🖻
 Shortcuts 💽 How to Add 💽 What's New
   >> poly([1,2,3])
   ans =
          1 -6 11 -6
   >> roots([1,-6,11,-6])
   ans =
       3.0000000000000002
       1.99999999999999998
       1.00000000000000000
fx >>
📣 Start
```

Note: The roots are the zero crossings

```
• Roots = { 1, 2, 3 }
x = [0:0.01:4]';
y = x.^3 - 6*(x.^2) + 11*x - 6;
plot(x,y);
grid on
```



Change the Problem to Fit the Solution

roots() finds the zero crossings of a polynomial

 $0 = x^3 + 5x^2 + 7x + 2$

If you want to find a different answer, change the problem

 $6 = x^3 + 5x^2 + 7x + 2$

becomes

$$0 = (x^3 + 5x^2 + 7x + 2) - (6)$$

Note that

• $6 = 0x^3 + 0x^2 + 0x + 6$

• You have to use matricies with similar dimensions

```
MATLAB 7.12.0 (R2011a)
File Edit Debug Desktop Window Help
 🎦 🔂 👗 ங 💼 🤊 🕲 🚵 🛒 📄 🥝 🛛 Settings\Administrator\My D
 Shortcuts 💽 How to Add 🛛 💽 What's New
   >> roots([1, 5, 7, 2])
   ans =
       -2.6180
       -2.0000
       -0.3820
   >> roots([1,5,7,2] - [0,0,0,6])
   ans =
     -2.7144 + 1.4000i
     -2.7144 - 1.4000i
       0.4288
   >> x = ans(3)
   x =
        0.4288
   >> x^3 + 5^*x^2 + 7^*x + 2
   ans =
        6.0000
fr
```

Matlab Scripts

Instead of typing the same set of commands over-and-over again, you can place these Matlab commands in a file (a Matlab script)

- The file must have a .m extension
- You can execute this script using the green arrow
- You can execute this script by calling it from the command window

2	Editor - C:\Documents and Settings\Administrator\Wy Documents\WATLAB\E 🔳 🗖 🔀
File	Edit Text Go Cell Tools Debug Desktop Window Help 🛛 🛥 🔻 🗙
: 1) 🖆 🛃 👗 🖏 🤊 ୯ । 🍪 🗊 • 🚧 🖛 🔶 😥 • 🗄 🗶 🖷 💟 👋 🔍 💌
•	$\mathbb{G} = -1.0 + \div 1.1 \times \%^{*}_{+} \%^{*}_{-} \mathbb{O}_{-}$
1 -	- x = [-2:0.5:2]';
2 -	$- y = (x+1) \cdot (x-1) \cdot (x-3);$
3 -	- disp([x,y])
4 -	-
	script In 4 Col 1 OVR
29	command window
File	MATLAB 7.12.0 (R2011a)
: *	Coll Cooling Desicip Window Field
: sh	nortcuts 2 How to Add 2 What's New
	>> Script Demo
	-2.0000 -15.0000
	-1 5000 -5 6250
	-1 0000 0
	-0.5000 2.6250
	0 3.0000
	0.5000 1.8750
	1.0000 0
	1.5000 -1.8750
	2.0000 -3.0000
fx	>>
4 9	ovr.



1	- M	ATLAB 7.12.0 (R2011a)
	File	Edit Debug Desktop Window Help
	: 🎦	逽 🕉 🐂 🖏 🤊 🥙 🖣 🗊 🖹 🥝 Documents and Settings\Admir ⊻ 🖻
	: Sho	rtcuts 🗷 How to Add 🗷 What's New
		>> rand
		ans =
		U.6463
		>> randn(1,3)
		ans =
		1 0000 1 1000 0 0007
		1.0933 I.I093 -0.8637
		>> ceil(8*rand(1,3))
		ans =
		0 0 2
	c	
	fx.	>>
1	📣 St	art

If-Statement

```
if - end
if - else - end
if - elseif - end
```

Do a set of operations if a statement is true

Valid boolean statements:

(N	== 3)	N=3
(N	> 3)	<i>N</i> > 3
(N	>= 3)	$N \ge 3$
(N	!= 3)	$N \neq 3$
(N	>= 3)*(N <= 7)	$3 \le N \le 7$

Example: Roll a loaded die

- 20% of the time you always roll a 6
- The rest of the time it's a fair die
- Each time you run the script, you get a new die roll



For-Loops

```
for i=1:100
   Matlab commands
   end
```

Repeat a set of commands a fixed number of times

Terminate with an end statement

Example: Cast a level-8 Fireball

- y = 8d6
- Use loaded dice (20% chance of a 6)
- Each time you run the script, you get a different result



Monte-Carlo Simulations

- One extermely useful capability of Matlab is to run Monte Carlo simulations
- To find the probability of an event, repeat an experiment 100,000 times
- The probability is then roughly the percentage of the time the outcome happened

Procedure:

- Write a script to run an experiment one time
- Once that works, repeat 100,000 times
- (place the code inside a for-loop)

Example: Ice Storm

The Dungeons and Dragon's spell *Ice Storm* does 6-40 damage

- The sum of two 8-sided dice and four 6-sided dice
- y = 2d8 + 4d6

Determine...

- The probability of doing N damage
- The probability that N > 30

🖻 Editor - C:\Documents and Settings\Administrator\Wy Documents\MATLAB\E 🔲 🗖
File Edit Text Go Cell Tools Debug Desktop Window Help 🛛 🛥 🗖
: 🗋 🗃 📓 🕹 🐂 🖏 🤊 (°) 🍓 🖅 - 🛤 🖛 🔶 😥 - 🗟 🗶 🖷 😒 👘 💟
: *≣ ⊑≣ - 1.0 + ÷ 1.1 × ∞° ∞ 0.
1 - d8 = ceil(8*rand(1,2));
<pre>2 - d6 = ceil(6*rand(1,4));</pre>
3 - y = sum(d8) + sum(d6)
4
Script Demo m* x TreStorm m x
script Ln 4 Col 1 OVR
MATLAB 7.12.0 (R2011a)
File Edit Debug Desktop Window Help
Shortruits (a) How to Add (a) What's New
У =
32
Y =
0.5
27
Y =
10
10
fx >>
▲ Start

Solution: Step 1

- Create a file *IceStorm.m*
- Find y = 2d8 + 4d6

Note that every time you run this script, you get a different answer

• it's random

Figure - C:\Documents and Settings\Administrator\Wy Documents\MATLAB\F	
File Edit Text Go Cell Tools Debug Desktop Window Help	
: 🞦 😂 📰 👗 ங 🛍 🤊 (°) 🌭 🖅 - 🚧 🗢 🔶 🈥 - 😫 🗶 🖷 💎 🛛 »	
: *= ⊑= - 1.0 + ÷ 1.1 × ∞ ⁴ ∞ ⁴ ∞	
1 - d8 = ceil(8*rand(1,2));	
<pre>2 - d6 = ceil(6*rand(1,4));</pre>	
3 - y = sum(d8) + sum(d6)	
4	
Scrint Demo.m* x TreStorm.m x	
script In 4 Col 1	OVR
MATLAB 7.12.0 (R2011a)	
File Edit Debug Desktop Window Help	
	<u> </u>
, Shortcuts A How to Add A what's New	
у =	
32	
y =	
27	
y =	
16	
<i>fx</i> , >>	
🔦 Start	[

Solution: Step 2

- Repeat 100,000 times
- Keep track of how many times you did y damage

The odds of doing 30 damage

>> Damage(30) / 100000 ans = 0.0297

The odds of doing 30 or more damage

```
>> sum(Damage(30:40) / 100000)
ans = 0.0866
```



Monte-Carlo Example #2

A and B are playing a match

- A has a 60% chance of winning any given game.
- What is the probability that A will win the match?

Start by playing a single match

- A won the match 5-2
- Different result each time you run the script

	■ & • • • • • • • • • • • • • • • • • •
*=	-1.0 + $\div 1.1$ × $\%_{+}^{6}$ $\%_{+}^{6}$ 0
1 -	Matches = 0;
2 -	A = 0;
3 —	B = 0;
4 —	□ for i=1:7
5 —	if(rand < 0.6)
б —	A = A + 1;
7 —	else
8 —	B = B + 1;
9 —	end
10 -	end
11 -	if(A > B)
12 -	Matches = Matches + 1;
13 —	end
14 -	disp([A, B, Matches])
Script Dr	emo.m* x Trestorm.m x Forl.oop.m x
	Ln 14 Col 21 OVR
🗚 MATL	AB 7.12.0 (R2011a)
File Edit	Debug Desktop Window Help
: 🔟 🖨	👗 🐂 🔄 7 (*) 🖓 🛐 🗐 🖉 istor\My Documents\MATLAB\ECE111\01_Basic 💙
Shortcut	is 🛃 How to Add 🛃 What's New

Now repeat for 100,000 matches

- A wins 70,913 times in 100,000 matches
- A has roughly a 70.9% chance of winning any given match

🖻 Editor - C:Wocuments and Settings\Administrator\Wy Documents\WATLAB\E 🗐									
File Edit Text Go Cell Tools Debug Desktop Window Help									
: 🗋 😂 🔜 & 🐂 🖷 🤊 (° 🌭 🖅 - 🗛 🗢 🔶 fiz, 돈 - 🗟 🗶 🖷 🖂 - 👋									
*= [] - 1.0 + ÷ 1.1 × % ⁺ % ⁺ 0									
1 - Matches = 0;									
2 - for N=1:1e5									
3 - A = 0;									
4 - B = 0;									
5 for i=1:7									
6 - if(rand < 0.6)									
- A = A + 1;									
8 - else									
9 - B = B + 1;									
10 - end									
11 - end									
12 - 1f(A > B)									
13 - Matches = Matches + 1;									
14 – end									
18 - Matches									
/ Swint Dama mit w Tracham m w Eastann m w									
script_benoun × testorman × Porcopan ×									
MAILAB 7.12.0 (R2011a)									
The Cuit Debug Desktop Willow Hep The Cuit Debug Debug Desktop Willow Hep The Cuit Debug Desktop Willow Hep The Cuit Debug D									
Shortcuts 🖉 How to Add 🖉 What's New									
Matches =									
70913									
$f_{x} >>$									
A Start									

While-Loop

while(statement is true)
 do the following
 end

Example: Count how many times you roll a die until you get a 1

🖻 E	dito	r - C:\Documents and Settings\Administrator\My Doc	uments\MA	TLAB\Ma	tch	_ 🗆 🔀
File	Edit	Text Go Cell Tools Debug Desktop Window Help				X 5 K
: 🔟	8		≥ <u>∙</u> € *	1989		» 🗆 💌
: *	¢#	-1.0 + $\div 1.1$ × $\%^{+}_{+} \%^{-}_{-}$				
1	-	N = 0;				
2	_	d6 = 0;				
3		- while (d6 ~= 1)				
4	_	d6 = ceil(6*rand);				
5	_	N = N + 1;				
6	_	⊢end	(1			
7		disp(['# rolls = ',int2str	(N)]);			
8						
2					- 1 - 2	
		script		ILN 8	Col 1	JOVR
A M		AB 7.12.0 (R2011a)			- l	
File	Edit	Debug Desktop Window Help	nd Sattings) Ad	ministrator	Mu Dou 😽	
: L	ortcut	to a whether the weather the second state of t	na secongs(Aa	ministrator	huà Doi 🔽	
	#	rolls = 2				~
	" #	rolls = 4				_
	#	rolls = 19				
	" #	rolls = 2				
	"	rolls = 4				
	"	rolls = 3				
	" #	rolls = 2				
	"	rolls = 1				
fx	" >>					
J.,	<					>
📣 SI	tart					OVR

While-Loop (cont'd)

What is the chance that it will take 7 or more rolls to get a 1?

Repeat 100,000 times

Monte-Carlo Simulation

In 100,000 trials

- It took 7 or more rolls 33,343 times
- There is about a 33.34% chance it will take 7 or more rolls to get a 1



While-Loop (cont'd)

Player A and B are playing a match

- Player A has a 60% chance of winning any given game
- When a player is up 3 games, the match is over

What is the chance player A wins the match?

Solution: Play a single match

- If A wins, A gains 1 point.
- If A loses, A loses 1 point.
- Keep playing until A is up 3 or down 3

🖻 Edito	or - C:\Documents and Settings\Administrator\Wy Documents\MATLAB\Match 🗖 🗖 🗙
File Edil	t Text Go Cell Tools Debug Desktop Window Help 🛛 🛥 🛪 🗙
: 🎦 🖻	×□ × ∨ 11 @ @ @ & 0 < 0 0 0 0 0 0 0 0 0 0
: *8 6	$1 - 1.0 + \dot{+} 1.1 \times \% \% \% 0$
1 -	A = 0;
2 -	= while (abs(A) < 3)
3 —	if(rand < 0.6)
4 —	A = A + 1;
5 —	else
6 –	A = A - 1;
7 —	end
8 —	end
9 —	if(A == 3)
10 -	Wins = 1;
11 -	else
12 -	Wins = 0;
13 -	end
14 -	<pre>disp(['Wins = ',int2str(Wins)])</pre>
	script Ln 14 Col 30 OVR
📣 МАТІ	LAB 7.12.0 (R2011a)
File Edi	t Debug Desktop Window Help
: 🛄 🛄	ta 🗟 Herrita Add 🗊 What's New
. snorce	ns = 1
wi	ns = 0
Wi	ns = 0
Wi	ns = 1
Wi	ns = 1
Wi	ns = 1
Wi	ns = 0
Wi	ns = 0
fx >>	
<	
📣 Start	OVR ,

Win-by-3 (cont'd)

Now play 100,000 matches

- A wins about 77% of the time with this format
- TV hates this format since a match can take a very long time



While-Loop: Tennis

Assume A and B are playing a match

- A has a 60% chance of winning any given game
- If a player wins 4 games and is up by 2 games, the match is over.
- Otherwise, the match continues until a player is up two games.

Find the probability that A wins the match

Solution: Start with playing a single match

- A wins 4 1
- A wins 6 4
- A loses 8 6
- A wins 4 1

🖻 Editor	- C:\Docum	nents and Settings\Administrator\My Documents\MATLAB\E	. 💶 🗖 🛛
File Edit	Text Go	Cell Tools Debug Desktop Window Help	X 5 K
1 6 1		🛍 🤊 (°) 🍐 🖅 + 🛤 🖛 🔶 fiz 📔 🛃 🗐 😒	» 🗌 🖌
+=	- 1.0	+ \div 1.1 × $9\%_{+}^{\%}$ 9% 0	
1 —	A =	0;	
2 -	в =	0;	
3 —	[]whil	e(max(A,B) < 4)	
4 —		$if(rand < 0.6) \frac{A}{2} = A + 1;$	-
5 —		else $B = B + 1;$	
б —		end	
7 —	Lend		
8 —	-whil	e(abs(A - B) < 2)	
9 —		if(rand < 0.6) = A + 1;	-
10 -		else $B = B + 1;$	
11 -		end	
12 —	^L end		
13 —	disp	([A,B])	
	-		
WhileLoop	.m × ler	script Ip 13 Col	1 OVR
📣 MATLA	B 7.12.0	(R2011a)	
nie cuit	X h h	No Nation window Help	isics 💌 📖 🕅
Shortcuts	How to A	dd 💽 What's New	
	_		
	4	1	
	6	4	
	6	8	
	4	1	
<i>fx</i> ; >>			
<			>

Now repeat for 100,000 matches

Result is A wins about 73,500 times

• A has about a 73.5% chance of winning the match

Results will vary each time you run this script

• it's random

To find the actual odds, you need to use a student-t test (week #15 of ECE 111)



Summary

Matlab is a fairly friendly computer language

You can use the command window as a calculator

• Adds, subtracts, multiplies, divides

Scripts allow you to try & modify code as you write it

For-loops let you run code multiple times

• Monte-Carlo simulations...

If-statements allow you to check for conditions

• If the sum is 25 or more...

While-loops let you run code until an event happens

• repeat until you roll a 1

Matlab Commands

Display

- format short display results to 4 decimal places
- format long display results to 13 decimal places
- format short e display using scientific notation
- format long e display using scientific notation

Polynomials

- poly(x)
- roots(x)
- conv(x,y)

Analysis

• sqrt(x) square root of x

e^x

X

10^x

- log(x) log base e
- $\log 10(x)$ log base 10
- exp(x)
- exp10(x)

• unwrap(x)

- abs(x)
- round(x) round to the nearest integer
- floor(x) round down (integer value of x)
- ceil(x) round up to the next integer
- real(x) real part of a complex number
- imag(x) imaginary part of a complex number
- abs(x) absolute value of x, magnitude of a complex number
- angle(x) angle of a complex number (answer in radians)
 - remove the discontinuity at pi (180 degrees) for a vector of angles
- sum(x) sum the columns of x
- prod(x) multiply the columns of x

Trig Functions

- sin(x)sin(x) where x is in radians
- $\cos(x)$ $\cos()$
- tan(x)tan()
- asin(x) $\arcsin(x)$
- acos(x) $\arccos(x)$
- $\arctan(x)$ • atan(x)
- atan2(y,x)angle to a point (x,y)

x!

x!

Probability and Statistics

- factorial(x)
- gamma(x)
- create an nxm matrix of random numbers between 0 and 1 • rand(n,m)
- randn(n,m)
- length(x)
- mean(x)
- std()
- create an nxm matrix of random numbers with a normal distribution
- return the dimensions of x
- mean (average) of the columns of x
- standard deviation of the columns of x

Display Functions

- plot(x) plot x vs sample number
- plot(x,y) plot x vs. y
- semilogx(x,y) = log(x) vs y
- semilogy(x,y) x vs log(y)
- $\log\log(x,y)$ $\log(x) vs \log(y)$
- mesh(x) 3d plot where the height is the value at x(a,b)
- contour(x) contour plot
- bar(x,y) draw a bar graph
- xlabel('time') label the x axis with the word 'time'
- ylabel() label the y axis
- title() put a title on the plot
- grid() draw the grid lines

Useful Commands

- hold on don't erase the current graph
- hold off do erase the current graph
- diary create a text file to save whatever goes to the screen
- linepace(a, b, n) create a 1xn array starting at a, increment by b
- logspace(a,b,n) create a 1xn array starting at 10[^]a going to 10[^]b, spaced logarithmically
- subplot() create several plots on the same screen
- disp('hello') display the message *hello*

Utilities

- format set the display format
- zeros(n,m) create an nxm matrix of zeros
- eye(n,m) create an nxm matrix with ones on the diagonal
- ones(n,m) create an nxm matrix of ones
- help using different functions
- pause(x) pause x seconds (can be a fraction). Show the graph as well
- clock the present time
- etime the difference between two times
- tic start a stopwatch
- toc the number of seconds since tic