Lesson 1 _Leon van Dommelen_

Contents

Related Assignments 1

Comments 2

Emergency Stop 2

Startup 2

Simple computations 2

Exponential notation 3

Bad numbers 3

Accuracy 4

Trig functions 4

Complex numbers 5

Precedence 5

HW publishing example: hw1_format.m 6

Matlab help 7

VARIABLES 7

Computing with variables 8

Manipulating variables 8

Pi to a trillion digits is not enough? 9

End lesson 1 10

Related Assignments

Preread + participation activities 1.1-3; 2.1-2,4-6
After class challenge activities: 1.1-3; 2.1-2,4-6
Also: hw1c

Comments

How to document your code.

```
% Lines starting with a single % are explanatory comments
% that matlab ignores.

% Lines starting with %% are also ignored, and act as
% section headers.
```

Emergency Stop

```
% create an infinite loop
%while(1) , x = 1, end

% Use Ctrl+c to stop (Hold down Ctrl while hitting the c
% key.)
```

Startup

Things to do before an interactive section. (Emacs users also set fill column to C-x 57.)

```
% reduce needless whitespace
format compact
% reduce irritations
more off

% echo the lines (me only)
echo on

% start a diary
%diary lesson1.txt
```

Simple computations

```
2+3
2-3
2*3
2/3
2^3
1.5/.5
```

```
ans =  5
ans = -1
ans =  6
ans =  0.66667
ans =  8
ans =  3
```

Exponential notation

```
% exponential notation for Planck's constant
1.0546e-34
% or not
0.00000000000000000000000000000000010546

% square of Planck's constant
1.0546e-34^2
```

```
ans =    1.0546e-34
ans =    1.0546e-34
ans =    1.1122e-68
```

Bad numbers

```
% Inf(inity)
1/0

% N(ot)aN(umber)
0/0

% "underflow" can be dangerous
1.0546e-34^10

% "overflow" is at least as bad
1/1.0546e-34^10
```

```
warning: division by zero
ans = Inf
warning: division by zero
ans = NaN
ans = 0
warning: division by zero
ans = Inf
```

Accuracy

```
% Normally Matlab numbers have a "relative" error of
    about 10^-16
(1/3)+(1/3)+(1/3)-1

% oops, not intended to be that accurate, try again
(1/3)+(1/3)+(1/3)+(1/3)+(1/3)+(1/3)-2

% try it with bigger numbers
(1000/3)+(1000/3)+(1000/3)+(1000/3)+(1000/3)+(1000/3)
    -2000

% print out the "absolute" error in 1
eps(1)
% print out the "absolute" error in 1000
eps(1000)
```

```
ans = 0
ans =   -2.2204e-16
ans =   -2.2737e-13
ans =    2.2204e-16
ans =    1.1369e-13
```

Trig functions

```
% matlab (and all science) uses radians by default
sin(30)
sin(pi/6)

% avoid using degrees if not needed
sind(30)

% watch very large values of the argument of trig
    functions
sin(10*pi)
```

```
sin(10000000000000000*pi)
```

```
ans = -0.98803
ans =  0.50000
ans =  0.50000
ans =   -1.2246e-15
ans = -0.37521
```

Complex numbers

```
% i stands for the square root of -1 (i.e. i^2 = -1)
2i^2
% can use j instead of i if needed
2j^2

% (1+i)^2 = 1^2 + 2i + i^2 = 2i?
(1+i)^2

% Taking the complex conjugate of a number means
% replacing every i in it by -i.

% complex conjugates multiply to real positive:
(1+i)*(1-i)
sin(1+i)*sin(1-i)
```

```
ans = -4
ans = -4
ans =  0 + 2i
ans =  2
ans =  2.0892
```

Precedence

If no parentheses are used, the following order of precedence applies:
highest: ^
lower: *, /
lowest +, -

```
% without parentheses
2+3*4
% since * takes precedence over +, this is the same as
2+(3*4)
% and not the same as
(2+3)*4
```

```
% without parentheses
12/2*3
% since / and * have equal precedence, this is
(12/2)*3
% and not
12/(2*3)
```

```
ans =  14
ans =  14
ans =  20
ans =  18
ans =  18
ans =  2
```

HW publishing example: hw1_format.m

```
%% HW 1c due MM/DD/YY _Eager Student Name_
%
% This is a hypothetical example homework, showing how
% your hwX.m file should be formatted before publishing
% it.

% get rid of excessive whitespace
format compact


%% Question 1
%
% Evaluate various simple expressions.

% a) evaluate 2+(6/4) using minimal parentheses
2+6/4

% b) evaluate (2+6)/4 using minimal parentheses
(2+6)/4


%% Question 2
%
% Plot sin(_x_) from -pi to pi.

% create the x and y values of the curve
x=[-pi:.01:pi]';   % 600 x-values
y=sin(x);          % corresponding sin(x) values
```

```
% plot it
plot(x,y);


%% End of HW 1c
```

The shown format must be followed in all homeworks.

Matlab help

```
% error function (try Tab completion)
%help erf

% also try
%doc erf
```

VARIABLES

Variables are named storage locations.

```
% there is no variable named 'x' yet (no response)
who x

% The next statement (command) is *not* a question.  It
% tells matlab to create a variable named 'x', if it does
% not yet exist, (like now), and then put the value 3 in
% that storage location.

% create x and store 3 in it
x=3

% now we have a variable x
who x
whos x

% we can print our its value by invoking its name
x
```

```
x =  3
Variables in the current scope:
x
Variables in the current scope:
```

```
  Attr Name        Size                     Bytes
    Class
  ==== ====        ====                     =====
       =====
         x          1x1                         8
             double
Total is 1 element using 8 bytes
x =  3
```

Computing with variables

```
% we can compute with x
x=x+7

% In the above statement, the right hand side is
% evaluated *first*.  Then the result, 10, is put in the
% storage location named "x".  The old value, 3, is
% *lost*.

% we can double x
x=x+x
% try using the Up-Arrow key a few times
```

```
x =  10
x =  20
```

Manipulating variables

```
% always keep track of *what* is stored in a variable
x=1
y=2

% let's try to swap the values naively
y=x;
x=y;

% Note in the above that the trailing semi-colons prevent
% the new values of x and y to be printed.  We were
% keeping them secret.  But now look at the results.

% we did not correctly swap the values; the 2 got lost:
x
y
```

```
% lets try again
x=1
y=2

% This time we prevent the value of y from becoming lost
% by storing it in a temporary variable called 'temp'

% save the original value of y
temp=y
% now give y the value of x
y=x
% and give x the *saved* value of y
x=temp
```

```
x =  1
y =  2
x =  1
y =  1
x =  1
y =  2
temp =  2
y =  1
x =  2
```

Pi to a trillion digits is not enough?

```
% show pi
pi

% the Indiana pi bill would redefine pi as 3.2
pi=3.2
pi

% maybe not a good idea?
clear pi

% we have the old value back
pi

% list the defined variables now
%who

% clear all and look again
%clear
%who
```

```
ans =  3.1416
pi =  3.2000
pi =  3.2000
ans =  3.1416
```

End lesson 1