

UNIVERSITY OF UTAH
ELECTRICAL ENGINEERING DEPARTMENT

EE 1000

MATLAB™ TUTORIAL

```
»diary on
»help path
```

PATH Get/set search path.

PATH, by itself, prettyprints MATLAB's current search path. The initial search path list is set by PATHDEF, and is perhaps individualized by STARTUP.

P = PATH returns a string containing the path in P.

PATH(P) changes the path to P. PATH(PATH) forces the path cache to be rebuilt.

PATH(P1,P2) changes the path to the concatenation of the two path strings P1 and P2. Thus PATH(PATH,P) appends a new directory to the current path and PATH(P,PATH) prepends a new path. If P1 or P2 are already on the path, they are not added.

For example, the following statements add another directory to MATLAB's search path on various operating systems:

```
Unix: path(path, '/home/myfriend/goodstuff')
VMS:  path(path, 'DISKS1:[MYFRIEND.GOODSTUFF]')
DOS:  path(path, 'TOOLS\GOODSTUFF')
Mac:  path(path, 'HardDisk:Tools:GoodStuff:')
```

See also WHAT, CD, DIR, ADDPATH, RMPATH.

```
»clear all
»% for help, type help at the prompt
»% A comment is anything after a % sign.
»% Matlab does simple math.
»2+2
ans =
    4
»% Matlab does simple math and remembers the result.
»a = 2 + 2
a =
    4
»% To continue a command on the next line, use 3 periods.
»a = 2 ...
+ 2
a =
    4
»% Matlab is somewhat like C but with simpler syntax.
»for ind = 1 : 2 : 5
»    ind
»end
ind =
    1
ind =
    3
ind =
    5
```

```

»if a > 3
»  a = a + 4
»end
a =
    8
»% Use a semicolon to suppress output.
»a = a + 5;
»% To see the value of a variable, type its name.
»a
a =
    13
»% Matlab is specialized for handling arrays and matrices.
»avec = [1 3 a]
avec =
     1     3    13
»% To make an array vertical, take its transpose using .'
»bvec = avec.'
bvec =
     1
     3
    13
»% Use single quotes to define a string.
»wierd = 'hi'
wierd =
hi
»% Matlab can expand an existing array.
»avec = [avec 12 6]
avec =
     1     3    13    12     6
»% An array may be empty.
»zerovec = []
zerovec =
[]
»% To enter a matrix, place a semicolon at the end of each row.
»Amat = [avec; 2 2 4 3 1]
Amat =
     1     3    13    12     6
     2     2     4     3     1
»% You may build matrices from vertical arrays, too.
»Bmat = [avec.' [2 2 4 3 1].']
Bmat =
     1     2
     3     2
    13     4
    12     3
     6     1
»% Just use * to multiply matrices.
»prodmat = Amat * Bmat
prodmat =
    359    102
    102     34
»% Matlab handles complex numbers with ease.
»acomplex = 1.4e-2 + i * 0.089
acomplex =
    0.0140 + 0.0890i
»acomplex = 1.5e-2 + 0.086i
acomplex =
    0.0150 + 0.0860i

```

```

»% Use length to find the size of an array.
»length(avec)
ans =
     5
»% Use size to find the size of a matrix.
»size(avec)
ans =
     1     5
»length(avec.')
ans =
     5
»% Use an index in parentheses to extract one element of an array.
»avec(3)
ans =
    13
»% Note that indexes start at 1, NOT 0.
»avec(0)
??? Index exceeds matrix dimensions.

»avec(length(avec))
ans =
     6
»% The colon operator means make a list by adding one until
»% 2nd number exceeded.
»3 : 8
ans =
     3     4     5     6     7     8
»8 : 3
ans =
Empty matrix: 1-by-0
»3.4 : 6.7
ans =
    3.4000    4.4000    5.4000    6.4000
»% Use a second colon and the middle number is the increment.
»8 : -1 : 3
ans =
     8     7     6     5     4     3
»listvec = 1.1 : 0.1 : 1.4
listvec =
    1.1000    1.2000    1.3000    1.4000
»% A colon as an index means the entire row or column.
»Amat
Amat =
     1     3    13    12     6
     2     2     4     3     1
»Amat(:,2)
ans =
     3
     2
»Amat(2,:)
ans =
     2     2     4     3     1
»% The zeros() and ones() functions are useful.
»Cmat = [ones(4,1) [1:4].']
Cmat =
     1     1
     1     2
     1     3
     1     4
»Dvec = zeros(1,5)
Dvec =
     0     0     0     0     0

```

```

»% Basic plotting is simple.
»plot(avec)
»% This plot command will create a 'Figure 1' window containing
»% a line plot of the values in avec.
»% Use files with .m as the extension for scripts (a list of
»% Matlab commands just like you would type at the prompt),
»% and functions (user-defined functions you can call from
»% the command line).
»% .* and ./ do element-by-element * and /
»[1 2] .* [3 4]
ans =
     3     8
»[1 2] ./ [3 4]
ans =
    0.3333    0.5000
»% .^ does element-by-element ^
»[2 3 5] .^ [0 2 3]
ans =
     1     9    125
»diary off

```