# MATLAB Quick Reference

## Operators

| Matrix Operations | | Array or Element by Element | |
|---|---|---|---|
| **+** | **Addition** | | |
| **-** | **Subtraction** | | |
| **\*** | **Matrix Multiplication** | **.\*** | **Element by Element Multiplication** |
| **/** | **Right Matrix Division** $b/A=bA^{-1}$ | **./** | **Element by Element Right Division** |
| **\\** | **Left Matrix Division** $A\backslash b=A^{-1}b$ | **.\\** | **Element by Element Left Division** $A.\backslash B=B./A$ |
| **^** | **Raise Matrix to a power** | **.^** | **Raise each Element to a power** |
| **'** | **Transpose matrix (conjugate if complex)** $A'$ | **.'** | **Transpose matrix (no complex conjugation)** $A.'$ |

| Relational Operators | | Logical Operators | |
|---|---|---|---|
| **<** | **Less Than** | **&** | **And** |
| **<=** | **Less Than or Equal** | **\|** | **Or** |
| **>** | **Greater Than** | **~** | **NOT** |
| **>=** | **Greater Than or Equal** | | |
| **==** | **Equal** | | |
| **~=** | **Not Equal** | | |

## Special Number Symbols

| | | | | |
|---|---|---|---|---|
| **pi** | $\pi$ | **i** | $\sqrt{(-1)}$ |
| **inf** | $\infty$ | **j** | $\sqrt{(-1)}$ |
| **NaN** | Not a number  ie 0/0 | | |

# Entering Matrices

| | |
|---|---|
| **x = [ 1 2 3; 4 5 6; 7 8 9]** | Comma or space between elements. |
| **x = [ 1,2,3**<br>  **4,5,6**<br>  **7,8,9 ]** | Semicolon or return for new row.<br>Enclosed in square brackets. |
| **a = [ exp(0)  sqrt(4)  1+2 ]** | Each element can be an expression. |
| **C = [ A B]** | A and B are matrices with the same number of rows. |
| **C = [ A ; B]** | A and B are matrices with the same number of columns. |

# Subscripts

| Subscripting a Vector | |
|---|---|
| **variable(V)** | **V is a vector of indexes** |
| **Examples** | |
| **A(3)** | The third element in vector A. |
| **A([3 8 11])** | The third, eighth and eleventh elements of A. |
| **A(3:11)** | The third to eleventh elements of A. |
| **A(11:end)** | All elements from the eleventh to the last element |

| Subscripting a Matrix | |
|---|---|
| **variable(R,C)** | **R is a vector of rows and C a vector of Columns** |
| **Examples** | |
| **M(2,3)** | The element in the second row and third column of M. |
| **M(2, [3 8 11])** | Second row, third, eighth and eleventh column. |
| **M(4,3:11)** | Forth row, third to eleventh column. |
| **M(11:end,2)** | Eleventh to the last row, second column. |
| **M(5,:)** | Fifth row, all columns. |
| **M(11:end,:)** | Eleventh to the last row, all columns. |

# Output Display Format

| | | | |
|---|---|---|---|
| **format short** | Fixed point, 4 decimal places | **format long** | Fixed point,<br>15 decimal places |
| **format** | Same as above | | |
| **format short e** | Floating point,<br>4 decimal places | **format long e** | Floating point,<br>15 decimal places |
| **format short eng** | Engineering notation,<br>4 decimal places | **format long eng** | Engineering notation,<br>15 decimal places |

## Generating Vectors

| | |
|---|---|
| **\<start\> : \<end\>** | **x = 1 : 5**    generates x = [ 1 2 3 4 5] |
| **\<start\> : \<separation\> : \<end\>** | **y = 0:5:20** generates y = [ 0 5 10 15 20] |
| **linspace(start,end,n)**    n = number of elements | **linspace(0,10,5)** generates [ 0 2.5 5 7.5 10] |
| **logspace(d1,d2,n)**    n elements logarithmically spaced between $10^{d1}$ and $10^{d2}$ . | **logspace**(-1,2,4)   generates  [0.1 1 10 100] |

## Utility Matrices

| | |
|---|---|
| **zeros(n)** | n by n matrix where each element is zero. |
| **zeros(m,n)** | m by n matrix where each element is zero. |
| **zeros(a,b,c)** | 3 dimensional array, a by b by c. |
| **ones(m,n)** | m by n matrix where each element is one. |
| **rand(m,n)** | m by n matrix of random numbers. |
| **eye(n)** | n by n identity matrix. |

## Variable Control

| | |
|---|---|
| **who** | List all variables in memory. |
| **whos** | Same as above but with more information. |
| **clear** | Remove all variables from memory. |
| **clear *\<variable\>*** | Remove specified variables from memory. |

## File Control Commands

| | |
|---|---|
| **dir** | List contents of current directory. |
| **ls** | List contents of current directory. |
| **what** | List the Matlab files in the current directory. |
| **cd** *\<directory\>* | Change the current directory. |
| **type** *\<filename\>* | Display the contents of a text or .m file. |
| **delete** *\<filename\>* | Delete a file. |
| **diary** *\<filename\>* | Record all commands and results to a file. |
| **diary off** | Stop above. |

## Help

| | |
|---|---|
| **help** | Display help topics. |
| **help \<function\>** | Help on a particular function. |
| **lookfor \<word\>** | Look for word in function descriptions. |
| **doc \<function\>** | Full documentation on function. |

# In Built Functions

Hit *fx* icon next to prompt for function browser. Only selected functions shown here.

| | | | |
|---|---|---|---|
| **abs(x)** | The absolute value. Modulus | **round(x)** | Round to the nearest integer. |
| **sqrt(x)** | The square root. | **ceil(x)** | Round up. |
| **exp(x)** | The exponential base e. $e^x$ | **floor(x)** | Round down. |
| **log(x)** | The natural logarithm. $\log_e(x)$ | **fix(x)** | Round towards zero. |
| **log10(x)** | The log base 10. $\log_{10}(x)$ | **rem(x,b)** | Remainder of x divided by b. |

## *Trigonometry*

| | |
|---|---|
| **sin(x)** | Sine. x in radians. |
| **sind(x)** | Sine. x in degrees. |
| **asin(x)** | The arcsine. The inverse of sin(x). Radians |
| **asind(x)** | The arcsine. Degrees |
| **sinh(x)** | Hyperbolic Sine. |
| **asinh(x)** | The inverse Hyperbolic Sine. |

**The above variations are also available for the following functions.**

| | | | |
|---|---|---|---|
| **cos(x)** | Cosine | | |
| **tan(x)** | Tangent | **cot(x)** | Cotangent |
| **sec(x)** | Secant | **csc(x)** | Cosecant |

## *Complex Numbers*

| | | | |
|---|---|---|---|
| **real(z)** | The real part of z. | **imag(z)** | The imaginary part of z. |
| **abs(z)** | The modulus of z. | **angle(z)** | The phase angle of z. |
| **conj(z)** | The complex conjugate of z. | | |

## *Matrix*

| | | | |
|---|---|---|---|
| **det(A)** | Determinant | **sqrtm(A)** | The matrix square root. |
| **norm(A)** | Norm | **expm(A)** | The matrix exponential base e. |
| **inv(A)** | Inverse | **logm(A)** | The matrix natural logarithm. |
| **[v,d]=eig(A)** | d = Eigenvalues, v = Eigenvectors | | |

## *Statistics*

| | | | | | |
|---|---|---|---|---|---|
| **max(x)** | Maximum | **median(x)** | Median | **var(x)** | Variance |
| **min(x)** | Minimum | **mean(x)** | Average | **std(x)** | Standard Deviation |

## Polynomials

| | |
|---|---|
| p = [1 2 3 4 5]; can represent the polynomial $x^4 + 2x^3 + 3x^2 + 4x + 5$ | |
| **y = polyval(p,x)** | Evaluate polynomial for each value in x. |
| **roots(p)** | Roots of polynomial. |
| **p = poly( <roots>)** | Polynomial with given roots. |
| **p = polyfit(x,y,n)** | Best fit of **x,y** data points to n<sup>th</sup> order polynomial. |

## Saving, Exporting and Importing Data

| | |
|---|---|
| **save** | Save all variable to the file matlab.mat |
| **load** | Load in variables from the file matlab.mat |
| **save <*filename*>** | Save all variable to the file *filename*.mat |
| **load <*filename*>** | Load in variables from the file *filename*.mat |
| **save <*filename*> <*variable*>** | Save only the variable *variable* to the file *filename*.mat |
| **load <*filename*> <*variable*>** | Load in only the variable *variable* from the file *filename*.mat |
| **save <*filename*> <*variable*> -ascii** | Save *variable* to the text file *filename* |
| **load <*filename*>.<*ext*>** | Load from the text file, to a variable called *filename* |

## Misc

| | |
|---|---|
| **x = input( 'What is x ? ')** | Ask the user for a number. |
| **[x,y] = ginput(1)** | Graph coordinates of a clicked on point. |
| **pause** | Wait for the user to press a key. |
| **pause(5)** | Wait for 5 seconds |
| **! command** | Execute an operating system command |
| **display(a)** | Suppose a = 5. This would print "a = 5"; |
| **disp(a)** | Same as above, but with out the "a = ". |
| **fprinf(' x = %f \n',x)** | Formatted print |

## fprintf Formatting

| | | | | | |
|---|---|---|---|---|---|
| **%d** | Integer | **%5f** | 5 characters wide | **\n** | New line |
| **%f** | Fixed point | **%5.2f** | 2 decimal places | **\t** | Horizontal tab |
| **%e** | Exponential Notation | **%-5.2f** | Left justify | **\\** | Back slash |
| **%s** | String of Characters | **%+5.2f** | Print sign (+ or -) | **%%** | Percent character |

# Graph Commands

| | |
|---|---|
| **plot(y)** | Plot y against index number. |
| **plot(x,y)** | Plot y against x |
| **plot(x1,y1,x2,y2)** | Plot y1 against x1 and y2 against x2. |
| **plot(x,y.'r+')** | Plot y against x using red plus signs. |
| **plot(x1,y1,'r+',x2,y2,'go')** | Red plus signs for x1 and y1, Green circles for x2 y2. |

| Symbol | Line Type or Mark | Symbol | Colour |
|---|---|---|---|
| **.** | Point | **r** | Red |
| **o** | Circle | **g** | Green |
| **x** | X mark | **b** | Blue |
| **+** | Plus sign | **y** | Yellow |
| **\*** | Stars | **m** | Magenta |
| **-** | Solid line | **c** | Cyan |
| **:** | Dotted line | **w** | White |
| **-.** | Dash dot line | **k** | Black |
| **--** | Dash Line | | |

| Other Types of Plot | |
|---|---|
| **fill(x,y,'r')** | Red filled graph |
| **bar(x,y)** | Bar graph |
| **stem(x,y)** | Stem plot |
| **loglog(x,y)** | x & y log scale |
| **semilogx(x,y)** | x log, y linear |
| **semilogy(x,y)** | x linear, y log |
| **polar(theta,r)** | Polar plot |
| **surf(x,y,z)** | 3D surface |
| **mesh(x,y,z)** | 3D mesh |
| **plot3(x,y,z)** | 3D line plot |

| Other Graphics Commands | |
|---|---|
| **title('Title')** | Graph title. |
| **xlabel('X axis')** | Label the x-axis. |
| **ylabel('Y axis')** <br> **zlabel('Z axis')** | Label y and z axis. |
| **text(x,y,'My Text')** | Place text at coordinates x,y. |
| **grid** | Place a grid on the graph. |
| **hold on** | Add any new plot to the current graph. |
| **hold off** | Replace current plot with any new plot. |
| **subplot(r,c,n)** | Split figure into r by c subplots. <br><br> subplot(2,3,1)  subplot(2,3,2)  subplot(2,3,3) <br> subplot(2,3,4)  subplot(2,3,5)  subplot(2,3,6) |
| **h = figure** | New graphics window. |
| **figure(h)** | Change to plotting in figure h. |
| **delete(h)** | Delete figure h. |
| **clf** | Clear current figure. |
| **drawnow** | Force the graph to update now. |

# Programming

## Enumerated Loops (for)

The general form of a **for** loop is :-

```
for <variable> = <vector>
    <statement>
    <statement>
        etc
end
```

Examples

```
A = rand(1,5);
for a = A
   disp(a)
end
```

```
for k = [ 1 7 3 pi i]
   disp(k)
end
```

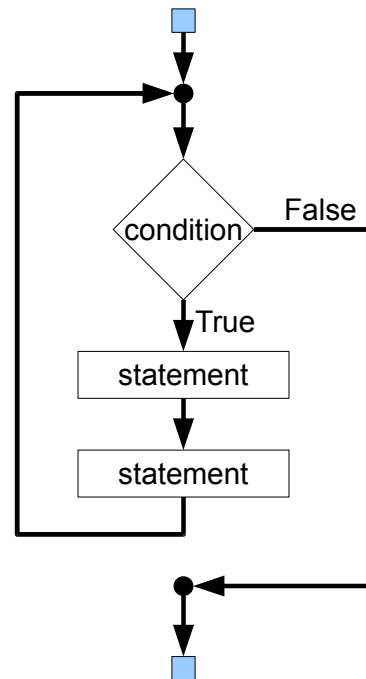## Precondition Loops (while)

Below is the general form of a **while** loop.

```
while(<condition>)
    <statement>
    <statement>
        etc
end
```

Example

```
A = 7;      %Find the square root of A.
x=1;        %First guess
err=1;      %Set error to get started

Newton Raphson Iteration
while(err>0.0001)
    x = (x.^2+A)./(2*x);
    % calculate the error
    err = abs(A-x.^2);
end

disp(x);
```

# Conditional Execution (if)

| | |
|---|---|
| General form of a simple **if** statement.<br><br>    **If** *<condition>*<br>      *<statement>*<br>      *<statement>*<br>         *etc*<br>    **end** | |
| The form of **if** statement with **else**<br><br>    **if**  *<condition>*<br>      *<statement1>*<br>      *<statement2>*<br>         *etc*<br>    **else**<br>      *<statement3>*<br>      *<statement4>*<br>         *etc*<br>    **end** | |
| You can also have as many **elseif** parts as you like.<br><br>    **if** *<condition1>*<br>      *<statement1>*<br>         *etc*<br>    **elseif** *<condition2>*<br>      *<statement2>*<br>         *etc*<br>    **else**<br>      *<statement3>*<br>         *etc*<br>    **end** | |

# Switch

Execution depends on the value of a variable.

```
switch <variable>
 case 1
   <statement1>
      etc
 case 2
   <statement2>
      etc
 case 3
   <statement3>
      etc
 otherwise
   <statement4>
      etc
 end
```
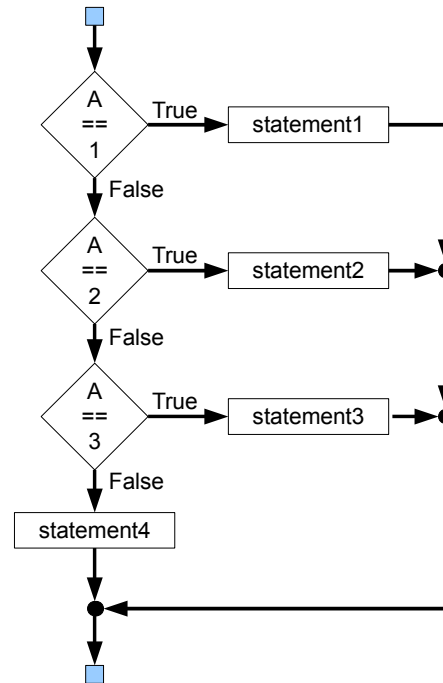
The case values can be any value that the switch variable can take. You can also put multiple values after the case.

```
case {1,2,3,4,5}
   <statement>
      etc
```

On the right A is the switch variable.



# Functions

| Function Definition | Using the Function |
|---|---|
| `function sum = myadd(a,b)`<br><br>`% The first block of comments`<br>`% defines what is printed out when`<br>`% you type    help myadd`<br><br>`% Comment not part of help`<br>`sum = a + b;` | **s = myadd(<expression>,<expression>)** |
| `function [sum,diff] = add_sub(a,b)`<br><br>` sum = a + b;`<br>` diff = a - b;` | **[s,d] = add_sub(<expression>,<expression>)** |

# Unix Commands

Linux and MAC OSX operating systems are both based on UNIX.

## *File and Directory Paths*

| /var/tmp | Absolute path from root | |
|---|---|---|
| **p5computing/exercise1** | Relative path from current working directory. | |
| **.** | The current directory | **cp   /tmp/myfile  .** |
| **..** | The directory above. | **cd   ../exercise2** |
| | | **cd  ../../..** |
| **~** | Your home directory | **cp   /tmp/myfile  ~** |
| | | **cd  ~/p5Computing** |

## *Commands*

| **ls** | List the contents of the current directory. |
|---|---|
| **ls -a** | List current directory, showing hidden files. |
| **ls -l** | List current directory, long format. More information. |
| **ls   <directory path>** | List the contents of the specified directory. |
| **ls -al   <directory path>** | As above showing hidden files and long format. |
| **mkdir  <directory name>** | Make a new directory with the given name. |
| **cd  <directory path>** | Change the current working directory. |
| **pwd** | Print the current working directory. |
| **cp  <file path>  <new file name>** | Copy a file to a new file. |
| **cp  <file path>  <directory path>** | Copy a file into the specified directory. |
| **mv  <file path>  <new file name>** | Change the name of a file to a new file name. |
| **mv  <file path>  <directory path>** | Move a file into the specified directory. |
| **rm  <list of files>** | Remove all files in the list. |
| **rm -i <list of files>** | Remove all files in the list, asking for confirmation. |
| **rm -R <directory path>** | Remove a directory and its contents. |
| **rmdir  <directory path>** | Remove an empty directory. |
| **cat  <file path>** | Type file to screen. |
| **more  <file path>** | Type file to screen a page at a time. |
| **man  <command>** | Display manual pages for the command. |

## *Wild Cards*

| **?** | A single character | **rm  prog?.m** | Remove prog1.m, prog2.m etc |
|---|---|---|---|
| **\*** | A character string | **cp \*.m MatlabFiles** | Copy all files ending .m to a directory. |

# DOS Commands

The commands you can use in a windows command prompt.

## *File and Directory Paths*

| Drive:<path><filename> | For Example C:\TEMP\mydirectory\myfile.txt | | | |
|---|---|---|---|---|
| **Drive:** | For example | | | |
| | **A:** | Floppy drive | **D:** | DVD |
| | **C:** | The main hard disc. | **F:** | USB Memory Stick |
| | If omitted, use the current drive. | | | |
| **<path>** | Absolute path from the root of the drive.   \TEMP\mydirectory\ <br> or the path from the current directory.    p5computing\exercise2\myfunction.m <br> If directory names contain spaces, use double quotes.  "Program Files\MATLAB" <br> If omitted, use current working directory. | | | |
| **.** | The current working directory | | | |
| **..** | The directory above. | | | |

## *Commands*

| | |
|---|---|
| **dir** | List the contents of the current directory. |
| **dir /w** | List current directory using wide format. |
| **dir  <directory path>** | List the contents of the specified directory. |
| **mkdir  <directory name>** | Make a new directory with the given name. |
| **cd  <directory path>** | Change the current working directory. |
| **copy  <file path>  <new file name>** | Copy a file to a new file. |
| **ren  <file path>  <new file name>** | Rename a file to a new file name. |
| **move  <file path>  <directory path>** | Move a file into the specified directory. |
| **del  <list of files>** | Delete all files in the list. |
| **del <list of files> /P** | Delete all files in the list, asking for confirmation. |
| **rmdir  <directory path>** | Remove an empty directory. |
| **type  <file path>** | Type file to screen. |
| **help  <command>** | Display help on command. |

## *Wild Cards*

| | | | |
|---|---|---|---|
| **?** | A single character | **del prog?.m** | Remove prog1.m, prog2.m etc |
| **\*** | A character string | **move \*.xls  mydir** | Copy all files ending .xls to a directory. |

Note:  In windows, directories are also called folders.