

## IMPORTING AND EXPORTING DATA

When you get into serious programming you will often need to store data on a disk. The process of moving data between MATLAB and disk files is called importing (from disk files) and exporting (to disk files). Data are saved in disk files in one of two formats: *text or binary*. In text format, data values are ASCII codes, and can be viewed in any text editor. In binary format, data values are not ASCII codes and cannot be viewed in a text editor. Binary format is *more efficient in terms of storage space required*. This lecture provides a brief summary of the main ways in which MATLAB imports and exports data. For full details consult MATLAB Help by searching the topic Importing and Exporting Data. Also search for the topic import wizard (or type, in the Command Window, help uiimport).

### **The load and save commands**

If you want to save data between MATLAB sessions the save and load commands are probably the best ones to use.

### **Exporting text (ASCII) data**

ASCII stands for American Standard Code for Information Interchange. Computers can only understand numbers, so an ASCII code is the numerical representation of a character such as 'a' or '@' or an action of some sort.

## Examples

% Save all variables from the workspace to test.mat:

```
save test.mat
```

% Save two variables, where FILENAME is a variable:

```
savefile = 'pqfile.mat';
```

```
p = rand(1, 10);
```

```
q = ones(10);
```

```
save(savefile, 'p', 'q');
```

To export (save) the array

A =

```
1  2  3  
4  5  6
```

in “delimited” ASCII format in the file *myData.txt* use the

command, If you view *myData.txt* in a text editor (or type it in the

Command Window) it looks like this:

```
1.0000000e+000  2.0000000e+000  3.0000000e+000  
4.0000000e+000  5.0000000e+000  6.0000000e+000
```

Delimiters are the characters used to separate the data values in the file—spaces by default. You can use tabs instead of spaces by specifying the *-tabs* qualifier instead of *-ascii*. If you save character arrays (strings) in this way, the ASCII codes of the characters are written to the file.

## Importing text (ASCII) data

The load command is the reverse of save, but has a curious twist. If the array A has been saved in *myData.txt* as above, the command,

`load myData.txt`

creates a variable in the workspace with the same name as the file, minus the extension, i.e., *myData*. If you don't want the filename as the variable name use the functional form of the command, e.g.,

```
A = load('myData.txt')
```

Data imported in this way doesn't have to be created by MATLAB. You can create it in a text editor, or it could be created by any other program that exports data in ASCII format.

### **Exporting binary data**

The command,

```
save filename x y z
```

saves the variables *x*, *y*, and *z* in the file *filename.mat* in MATLAB proprietary binary format, i.e., such a MAT-file can only be used by MATLAB.

Note:

- If no variables are listed the entire workspace is saved.
- The extension *.mat* is the default—you can specify a different extension.
- Seek help for all the save options.

### **Importing binary data**

The command,

```
load filename
```

loads all the variables from *filename.mat* into the workspace; see help for all the load options.

## fprintf() Function

With this function we can display formatted output as controlled by the following:

Value Type	Conversion	Details
Integer, signed	%d or %i	Base 10 values
Integer, unsigned	%u	Base 10
	%o	Base 8 (octal)
	%x	Base 16 (hexadecimal), lowercase letters a-f
	%X	Same as %x, uppercase letters A-F
Floating-point number	%f	Fixed-point notation
	%e	Exponential notation, such as 3.141593e+00
	%E	Same as %e, but uppercase, such as 3.141593E+00
	%g	The more compact of %e or %f, with no trailing zeros
	%G	The more compact of %E or %f, with no trailing zeros
Characters	%c	Single character
	%s	String of characters

Action	Flag	Example
Print sign character (+ or -).	'+'	%+5.2f
Pad with zeros.	'0'	%05.2f

Two commands that are frequently used to generate output are: *disp* and *fprintf*. The main differences between these two commands can be summarized as follows

- disp*
  - . Simple to use.
  - . Provide limited control over the appearance of output
- fprintf*
  - . Slightly more complicated than *disp*.
  - . Provide total control over the appearance of output

Some examples of *fprintf* function:

□ `fprintf('%0.3f',12345.6789);`

➤ 12345.679

□ `fprintf('%012.3f',12345.6789)`

➤ 00012345.679

□ `fprintf('%012.5f',12345.6789);`

➤ 012345.67890

□ `fprintf('%0.3g and %0.3f',7.36019,7.36019);`

➤ 7.36 and 7.360 (g: round to 10)

□ `fprintf('%0.3f , %06.1f',7.36,-1.25);`

➤ 7.360 , -001.3

□ `fprintf('78 in Hexa=%X while in Octa=%o.',78,78);`

➤ 78 in Hexa=4E while in Octa=116.

□ `fprintf('Planck's constant=%0.6G Js',6.6260688e-34);`

➤ Planck's constant=6.62607E-034 Js

□ `fprintf('Planck's constant=%0.6E Js',6.6260688e-34);`

➤ Planck's constant=6.626069E-034 Js

□ `fprintf('Name:\t%s\n Age:\t%d', 'Rami',28);`

➤ Name: Rami

Age: 28

□ `R=52;V=12;`

`fprintf('when the Voltage=%g Volt\n and Resistor=%g Ohm\n then  
the Total Power=%0.3g Watt',V,R,V^2/R);`

➤ when the Voltage=12 Volt  
and Resistor=52 Ohm  
then the Total Power=2.77 Watt

## Probability

Here is a simple example of probability in MATLAB.

### Spinning Coins

When a fair (unbiased) coin is spun, the probability of getting heads or tails is 0.5 (50%). Since a value returned by rand is equally likely to anywhere in the interval [0, 1) we can represent heads, say, with a value less than 0.5, and tails otherwise.

Suppose an experiment calls for a coin to be spun 50 times, and the results recorded. In real life you may need to repeat such an experiment a number of times; this is where computer simulation is handy. The following script simulates spinning a coin 50 times:

Q/ Write a MATLAB program to simulate spinning a coin 50 times. Use rand function.

```
for i = 1:50
    r = rand;
    if r < 0.5
        fprintf( 'H' )
    else
        fprintf( 'T' )
    end
end
fprintf( '\n' )           % newline
```