Input/Output in MATLAB

CHEN 1703

See also:

- The class wiki page notes on I/O
- Your text book, §3.4 and Appendix C

THE UNIVERSITY OF UTAH

VERY Basic MATLAB File I/O

Saving variables to files & Loading variables from files

- **`save`** `filename x y -ASCII`
  - `filename` is the name of the file that you want to write data to.
  - `x`, `y` are variables to be written to the file.
    - If omitted, all variables are written.
  - `-ASCII` tells Matlab to write the data in a format that you can read.
    - If omitted, data will be written in binary format.
      - best for large amounts of data

- **`load`** `filename x y`
  - This is the complimentary command to **`save`**.
  - Reads variables `x` and `y` from file `filename`
    - If variables are omitted, all variables are loaded...

```
clear;
x = linspace(-pi,pi);
y = cos(x);
save myVariables x y;

clear;
load myVariables;
who;  plot(x,y);
```

Formatted Output in Matlab

- **`disp(x)`** - prints the contents of variable `x`.
- **`fprintf(...)`** - use for formatted printing
  - Allows much more control over output
  - Syntax: **`fprintf`**`(`*'text & formatting'*`,`*variables*`)`**`;`**
  - Text formatting:
    - `%a.bc`
      - a - *minimum* width of output buffer
      - b - number of digits past decimal point
      - c - formatting scheme
        - f - floating point (typical format) `12.345`
        - e - scientific notation - `1.2345e1`
        - s - string format

Control Code	Description	Example
\n	Begin a new line	`fprintf('hello.\n');`
\t	Insert a "tab"	`fprintf('\thello.\n');`
\\	insert a backslash	`fprintf('\\hello.\\\n');`
''	Insert a single quote	`fprintf('''hello.''\n');`
%%	Insert a % sign	`fprintf('%%%1.2f\n',95.6);`

```
x = [1.1 2.2 3.3 4.4];
y = 2*x;
fprintf('Hello. (%1.3f,%1.3f), (%1.1f,%1.0f)\n',...
    x(1),y(1),x(3),y(3));
```

```
Hello. (1.100,2.200), (3.3,7)
```

THE UNIVERSITY OF UTAH

Formatted Output - Examples

```
fprintf('%6s%8s\n','index','value');
fprintf('--------------\n');

n = 5;
a = zeros(5,1);
for( i=1:5 )
   a(i) = 2*i+1;
   fprintf('%6.0f%8.1f\n',i,a(i));
end
```

Repeat temperature conversion example using **`fprintf`** rather than **`disp`**.

File Output in MATLAB

- Three steps:
  - Open the file
    - `fid = fopen(filename,'w');`
    - `'w'` tells matlab that we want to WRITE to the file.
    - see "help fopen" for more information.
  - Write to the file
    - `fprintf(fid,format,variables);`
  - Close the file
    - `fclose(fid);`

Example:
Temperature conversion example - write results to a file called "tempTable.dat"

File Input in MATLAB

- Import wizard "File→Import Data"
  - Allows you to import data from delimited files (spreadsheets, etc)
- Importing "spreadsheet" data
  - dlmread - import data from a delimited file (you choose the delimiter)
  - xlsread - import data from Excel.
- General file input - three steps:
  - `fid=fopen(filename,'r')` - open a file to allow detailed input control.
    - `'r'` tells matlab that we want to READ from the file.
  - `a=fscanf(fid,format,size);`
    - Works like file writing, but use **`fscanf`** rather than **`fprintf`**.
    - `fid` - file id that you want to read from
    - `format` - how you want to save the information (string, number)
      - **`'%s'`** to read a string, **`'%f'`** to read a floating point number, **`'%e'`** to read scientific notation.
    - `size` - how many entries to read.
    - **`feof`**`(fid)` - returns true if end of file, false otherwise.
  - **`fclose(fid);`**

File Input - Example 1

General form of an n^{th} order polynomial:

$$p(x) = \sum_{i=0}^{n} a_i \, x^i$$

For a quartic (n=4) we have:

$$p(x) = \sum_{i=0}^{4} a_i \, x^i = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$$

```
clear; clc;

% open a file to read - first line contains
% the order of the polynomial.  Second line
% contains the polynomial coefficients.
fid = fopen('poly.dat','r');

% read the order of the polynomial
n = fscanf(fid,'%f',1);

% read all of the polynomial coefficients
a = fscanf(fid,'%f',n+1);
```

"poly.dat"

```
4
1.0  2   0.02  4.0  0
```

"poly.dat"

```
2
0   1  2.3
```

File Input - Example 2

Have the user specify the number of elements in a molecule in a file. Read the file and then output the molecular weight. Include H C O N S.

Convention: input lines look like:

```
Element Number
```

Example:

```
C    1
H    4
```

Steps:

1. Set up the MW vector: `mw=[mwH mwC mwO mwN mwS];`
2. Set up the composition vector: `nAtoms=zeros(1,5);`
3. Open the file
4. While we aren't at the end of the file
    1. read a line.
    2. Assign the proper entry in nAtoms
5. Calculate the mixture molecular weight & output it.

Excel - I/O

- Reading delimited data
  - File→Open (may need to select file type to be "all files")
  - An import wizard opens, allowing you to select delimiters