Writing data to a text file

There are three steps to writing data to a file: (1) open the file, (2) write the data, and (3) close the file. Below are two examples for writing data to a plain text file; the first example deals with characters and the second deals with numeric data.

```
function n = charArray2File(A, fname)
% Write the characters in char array A to a text file.
% fname is a string that names a plain text file.  If the file already
%   exists, it is overwritten.
% n is the number of lines written to the file.

n= 0;                   % No. of lines written so far
if isempty(A) || ~ischar(A)
    return;
end
fid= fopen(fname, 'w'); % fid is the identifier for the file (opened for writing)
[n,nc]= size(A);
for r= 1:n
    fprintf(fid, '%s\n', A(r,:));
end
fclose(fid);
```

The built-in functions `fopen` and `fclose` open and close a text file, as suggested by the function names. When you open a file, you need to specify the filename as well as how the file will be used: ‘r’ for read; ‘a’ for append, i.e., append to an existing file; or ‘w’ for write, i.e., create a new file or replace any text if the file already exists. The built-in function `fprintf` is used to print (write) data to the file. By specifying a first argument that is the file identifier (variable `fid` above) in calling `fprintf`, you specify that the printing is to be done to the file instead of the default output—screen.

```
function n = numArray2File(A, fname)
% Write the numeric values in array A to a text file using the number
%   format %10.4f
% fname is a string that names a plain text file.  If the file already
%   exists, it is overwritten.
% n is the number of lines written to the file.

n= 0;  % No. of lines written so far
if isempty(A) || ~isnumeric(A)
    return;
end
fid = fopen(fname,'w'); % fid is the identifier of the file (opened for writing)
[n,nc]= size(A);
for i=1:n
   str = sprintf('%10.4f', A(i,:));
   fprintf(fid, '%s\n', str);
end
fclose(fid);
```

The built-in functions `ischar` and `isnumeric` tests whether an array contains characters and numbers (e.g., type `double`), respectively. In the example on numeric data above, note that the statement

```
   str = sprintf('%10.4f', A(i,:));
```

applies the number format `%10.4f` to *every* number on the $\texttt{i}^{th}$ row of array `A`. Function `sprintf`, like `fprintf`, creates a string using the specified format sequences, if any, but `sprintf` does not print to the screen or to a file.