These tables document the MATLAB functions used in *BlackBox1*.

MATLAB function	*Output Variables & Types*	*Input Variables & Types*
str = **input**(*prompt*, 's')	*str* is a string variable	*prompt* is a string
disp(*X*)	(There is no output variable.)	*X* is a variable or constant (including a string)
L = **length**(*X*)	*L* is a numerical variable	based on your experience, *X* is a string or string variable
s = **num2str**(*A*)	*s* is a string variable	based on your experience, *A* is a number or numerical variable
tf = **strcmp**(*s1*, *s2*)	*tf* is a logical variable	*s1* and *s2* are strings or string variables

MATLAB function	*What the Function Does*
str = **input**(*prompt*, 's')	prompts the user for input and stores that input as a string in the designated string variable *str* (that's what the 's' is encoding)
disp(*X*)	displays *X*, if *X* is a constant, or the value of *X*, if *X* is a variable
L = **length**(*X*)	stores in the variable *L* the number of characters in the string (stored in) *X*
s = **num2str**(*A*)	stores in variable *s* the character string of numerals depicting the numerical value to which *A* evaluates
tf = **strcmp**(*s1*, *s2*)	stores a logical 1 in the variable *tf* if *s1* and *s2* evaluate to the same string; stores a logical 0 in *tf* otherwise

MyFirstScript Solutions

MyFirstScript script should be a modification of *BlackBox2* in that it prompts you to answer "Y," "N," or "Maybe" to the question "Am I right?"

In creating this script, you will need to figure out a response to "Maybe," and also how to have MATLAB produce that response.

There are two different solutions that people have found for this problem in the past. Below I highlight in red just the lines that need to be changed from *BlackBox2*.

(You may have an equivalent solution using different variable names than I used. Note that I also changed the comment lines to reflect the program changes.)

Solution 1 (make two comparisons with **strcmp**):

```
% Display a message telling the user how many letters their name has
% and prompting them to agree, disagree, or not commit to either.
% Their response should be the character 'Y,' the character 'N,' or the string 'Maybe,'
% and is stored in the variable "YNM".

YNM = input('Am I right? (Y/N/Maybe):','s');

% Compare the character stored in "YNM" with the character 'Y.'
% If it agrees, store logical 1 in the variable "TFY", signalling 'true.'
% Otherwise, store logical 0 in the variable "TFY", signalling 'false.'

TFY = strcmp(YNM, 'Y');

% Compare the character stored in "YNM" with the string 'Maybe'.
% If it agrees, store logical 1 in the variable "TFM", signalling 'true.'
% Otherwise, store logical 0 in the variable "TFM", signalling 'false.'

TFM = strcmp(YNM, 'Maybe');

% Add the integer 1 to TFY + 2*TFM.
% This automatically interprets the logical values in "TFY" and "TFM" as numbers.
% The step is needed because the variable "Select" is going to be used to
% select one of three responses, which must be identified in the cell array
% "Response" using the integers 1, 2 or 3, not 0, 1 and 2.

Select = 1 + TFY + 2*TFM;

Response = { 'Too bad!' 'Great!'  'So you don't trust me!'};
```

Solution 2 (choose the response using **length**):

```
% Display a message telling the user how many letters their name has
% and prompting them to agree, disagree, or not commit to either.
% Their response should be one of the strings 'Yes,'  'No,' or 'Maybe,'
% and is stored in the variable "YNM".

YNM = input('Am I right? (Yes/No/Maybe):','s');

% Determine the length of the reply string.

RL = length(YNM);

% Subtract the integer 1 from RL.
% This step is needed because the variable "Select" is going to be used to
% select one of three responses, which must be identified in the cell array
% "Response" using the integers 1, 2 or 4, not 0, 1 and 3.

Select = RL – 1;

Response = { 'Too bad!' 'Great!'  '(filler)' 'So you don't trust me!'};
```

Further Remarks

If you have some experience programming in another language, you may have wondered why we couldn't have used some "if .. then…" statements to handle situations where the program should respond differently under different conditions.

MATLAB does have such *conditional* statements, and you will learn about them soon. But this exercise was about "making do" with the few MATLAB functions you have learned about so far!

Answers to the "Wrapping Up" Questions

1. How do you start MATLAB for this course on the lab computers?

Ans: Click on the "Start" icon at the lower left corner of your screen, and select:

All Programs → Cognitive Science → MATLAB Cog Sci 242

2. How do you open a MATLAB file (an "m file") from within MATLAB in order to edit it?

Ans: Go to the menu bar at the top of the "Home" tab and select **Open → Open …,** then scroll through the directories in the window at the left to find the file.

3. What is a "string," and how do you indicate in MATLAB that a particular sequence of characters is to be treated as a string?

Ans: A sequence of adjacent characters (which may include whitespace characters), treated as a single object, is called a *string*. You can designate a sequence of characters as a string by placing the sequence between single quotes.

4. What is a "logical variable"? What values can it have (in MATLAB)?

Ans: A logical variable is a variable that has one of only two values, denoted in MATLAB by **false** and **true**, or by the numerals 0 and 1 .

5. What does "case sensitive" mean? Is MATLAB case sensitive?

*Ans***:** A system that works with alphabetic symbols is "case sensitive" if it distinguishes between lower-case and upper-case letters.
MATLAB is case sensitive.

6. How would you code the following instruction in MATLAB?

"Assign the value 3 to the variable X." *Ans*: X = 3

7. Which MATLAB functions would you use for the following purposes?

- comparing two strings to see whether or not they are the same: **strcmp**
- prompting the user for input from the keyboard and assigning that input to a variable: **input**
- displaying a string on the screen: **disp**
- converting a number to its corresponding string of numerals: **num2str**
- determining the length of a string: **length**

8. How do you prevent the result of a MATLAB command from being displayed?

Ans: place a semicolon (;) at the end of the line containing the command

9. How do you indicate a line in a MATLAB program is a "comment"?

Ans: place a percent sign (%) at the start of the line