**MATLAB Special Characters, Reserved Words, and Functions**

**APPENDIX A**

## Special Characters Description



<...> Used to indicate template parameters—data to be supplied

% Indicates a comment in an m-file

{...} Defines a cell array

[ ] The empty vector

[...] Concatenates data, vectors, and arrays

( ) Used to override operator precedence

( ) Used to identify the formal and actual parameters of a function

(...) Used to index an array

(<variable>) Used to allow a variable to be used as a structure field

'abc' Encloses a literal character string

‘ Transposes an array

; Suppresses output when used in commands

; Separates rows in an array definition

: Specifies a vector in the form <from:incr:to>

: Used in slicing vectors and arrays

. Used to access fields of a structure

... Used to continue a MATLAB command to the next line

## Mathematical Operators Description

= Assignment operator—assigns a value to a variable (memory location); not the same as an equality test

+ Scalar and array addition

**−** Scalar and array subtraction

− Unary negation

* Matrix multiplication

.\* Element-by-element multiplication

/ Matrix division

./ Element-by-element division

^ Matrix exponentiation

.^ Element-by-element exponentiation

**A–1**

**A–2 Appendix A** MATLAB Special Characters, Reserved Words, and Functions

## Logical Operators Description

< Less than

<= Less than or equal to

> Greater than

>= Greater than or equal to

== Equal to

˜= Not equal to

& Element-by-element logical AND

&& Short-circuit logical AND (scalar)

| Element-by-element logical OR (vectors)

|| Short-circuit logical OR (scalar)

˜ Unary not

## Logical Functions Description

all(a) True if all the values in a (a logical vector) are true

and(a, b) True if both a and b are true

any(a) True if any of the values in a (a logical vector) are true

not(a) True if a is false; false if a is true

or(a, b) True if either a or b is true

## File Input and Output Description

[nums txt raw] = Read comma-separated text files

csvread(<file>)

csvread(<file>) Read comma-separated text files

csvwrite(<file>, Write comma-separated text files

<data>)

dlmread (<file>, Read text files separated by the given delimiting character(s)

< dlm>)

dlmwrite(<file>, Write text files separated by the given delimiting character(s)

<data>, <dlm>) fclose(<handle>) Close a text file

fgetl(<handle>) Read a line omitting the new-line character

fgets(<handle>) Read a line including the new-line character fh = fopen Open a text file for reading or writing (<handle>,

<why>)

fprintf Write to the console, or to plain text files (when <handle> is present

(<handle>, ...)

imread(<file>) Read an image file

imwrite(<data>, Write an image file

<file>, <format>)

load <file> Load the current workspace from a file

# **Appendix A** MATLAB Special Characters, Reserved Words, and Functions **A–3**

save <file> Save workspace variables in a file

[tk rest] = Extract a token from a string and return the remainder of the string

strtok(<str>,

<dlm>)

ca = textscan Acquire and scan a line of text according to a specific format and

(<handle>, save the data in a cell array

<format>)

[data Fs nb] = Read a sound file in .wav format

wavread(<file>)

wavwrite(<data>, Write a sound file in .wav format

<Fs>, <nb>,<file>)

[nums, txt, Read an Excel spreadsheet

raw] = xlsread(<file>)

xlswrite(<file>, Write an Excel spreadsheet in a specific

<data>, <sheet>, row/column range

<range>)

## Format Control Description

%<m>.<n>e Exponential notation

%<m>.<n>f Fixed point or decimal notation

%<m>.<n>g Fixed point or exponential notation

%q A quoted string delimited by double quotes

%<n>s Character string

\b Backspace

\n New Line

\t Tab

## Display Formatting Description

format compact Set format to compact form format long Set format to 14 decimal places format long e Set format to 14 exponential places

format loose Set format back to default, non-compact form format short Set format back to default, 4 decimal places format short e Set format to 4 exponential places

## User Interface

**Management Description**

ans Default variable name for results of calculations

clc Clear the interactions window

clear <selection> Remove all (or slected) variables from the workspace

clf Clear the current figure

*continued on next page*

**A–4 Appendix A** MATLAB Special Characters, Reserved Words, and Functions

close all Close all graphics windows

exit Terminate the user interface system

help <topic Invoke the help utility

or function>

load <file> Load the current workspace from a file

quit Terminate the user interface system

save <file> Save workspace variables in a file

who List variables in the workspace

whos List variables and their sizes

## Special Constants Description

eps Smallest possible difference between two floating point numbers

false Logical false

inf Infinity

NaN Not a number

pi Ratio of the circumference of a circle to its diameter

true Logical true

## Basic Mathematical

**Functions Description**

abs(x) Compute the absolute value

ceil(x) Round x to the nearest integer toward positive infinity

cross(a, b) Vector cross product

exp(x) Compute e to the power x

fix(x) Round x to the nearest integer toward zero

floor(x) Round x to the nearest integer toward minus infinity

log(x) Compute the natural log of x

log10(x) Compute the log base 10 of x

mod(x, a) Compute the remainder when x is divided by a

rem(x, a) Compute the remainder when x is divided by a

round(x) Round x to the nearest integer

sqrt(x) Calculate the square root of x

## Trigonometry Description

acos(x) Compute the inverse cosine (arcsine) of x

asin(x) Compute the inverse sine (arcsine) of x

atan(x) Compute the inverse tangent (arctan) of x

atan2(y, x) Compute the inverse tangent given the x and y values (4 quadrant

resolution)

cos(x) Compute the cosine of x

sin(x) Compute the sine of x

tan(x) Compute the tangent of x

# **Appendix A** MATLAB Special Characters, Reserved Words, and Functions **A–5**

## Vector, Array, and Matrix

**Operations Description**

cumsum(v) Compute a cumulative sum of the values in v

deal(...) Distribute cell array results among variables

det(a) Compute the determinant of a matrix

diag(a) Extract the diagonal from a matrix or (if provided a is a vector) construct a matrix with a as the diagonal

eye(n) Generate the identity matrix of size n x n

find(<logical a>) Compute a linear list of the locations of the true values in a logical array

fliplr(a) Flip a matrix from left to right

inv(a) Compute the inverse of a matrix

length(a) Determine the largest dimension of an array

linspace(from, Define a linearly spaced vector

to, n)

magic(n) Generate a magic square of size n x n [v,in] = max(a) Find the maximum value and its position in a mean(a) Compute the average of the elements in a meshgrid(x, y) Map each of two vectors into separate 2-D arrays [v,in] = min(a) Find the minimum value and its position in a ones(r, c) Generate an array filled with the value 1

prod(x) Compute the product of all the items in x

rand(r, c) Calculate an r x c array of evenly distributed random numbers in

the range 0…1

randn(r, c) Calculate an r x c array of normally distributed random numbers in the range 0…1

size(a) Determine the dimensions of an array

sparse Define a sparse matrix

[v,in] = sort(v) Sort the vector v (a vector or a cell array of strings)

sum(a) Find the sum of an array

zeros(r, c) Build an array filled with the value 0

## 2-D Plotting Description

bar Generate a bar graph

barh Generate a horizontal bar graph

contour Generate a contour plot

hist Draw a histogram

loglog Generate an x-y plot, with both axes scaled logarithmically

pie Generate a pie chart

plot Create an x-y plot

polar Create a polar plot

semilogx Generate an x-y plot, with the x-axis scaled logarithmically

semilogy Generate an x-y plot, with the y-axis scaled logarithmically

**A–6 Appendix A** MATLAB Special Characters, Reserved Words, and Functions

## 3-D Plotting Description

bar3 Generate a 3-D bar graph

barh3 Generate a horizontal 3-D bar graph

gplot Plot a graph

mesh Generate a mesh plot of a surface

meshc Generate a mesh plot of a surface with contours

meshz Generate a mesh plot of a surface with a skirt

meshgrid(r, c) Create a plaid for 3-D plots

peaks Create a sample matrix used to demonstrate graphing functions

pie3 Generate a 3-D pie chart

plot3 Generate a 3-D line plot

sphere Example function used to demonstrate graphing

surf Generate a surface plot

surfc Generate a combination surface and contour plot

waterfall Generate a mesh plot of a surface with one skirt edge

## Plot Appearance Line

**Type Control Description**

* + Solid

: Dotted

-. Dash-dot

* + - Dashed

. Point

o Circle

x x-mark

+ Plus

\* Star

s Square

d Diamond

ˇ Triangle down

^ Triangle up

< Triangle left

> Triangle right

p Pentagram

h Hexagram

## Color Control Character Description

1. Blue
2. Cyan

g Green

k Black

m Magenta

# **Appendix A** MATLAB Special Characters, Reserved Words, and Functions **A–7**

r Red

w White

y Yellow

## Figure Control Description

axis Freezes the current axis scaling for the current plot or specifies the axis dimensions

figure <n> Open a new figure window. If present, <n> specifies a figure number

grid off/on Turn the grid off or on

hold off/on If hold is not set, erase figure contents before the next plotting instruction

legend(ca) Add a legend to a graph

shading <value> Shade a surface plot with one color per grid section subplot(plts, n) Divide the graphics window up into sections available for plotting text(x,y,str) Add text to a plot

title(str) Add a title to a plot

xlabel(str) Add a label to the x-axis

ylabel(str) Add a label to the y-axis

zlabel(str) Add a label to the z-axis

## Color Map Values Description

autumn yellow, orange, and red colors

bone shades of gray

colorcube multiple multi-color bands

cool light blue to purple

copper shades of red-brown

flag multiple red, white, and blue bands

hot deep red through orange to white

hsv single spectrum from red to purple

jet (default) rainbow from blue to red

pink from dark to light pink

prism multiple bands of spectrum colors

spring from purple to yellow

summer from dark green to yellow

white all white

winter from dark blue to light green

## String Operations Description

disp(...) Display matrix or text

fprintf(...) Print formatted information

input(...) Prompt the user to enter a value and parse the result

int2str(a) Convert an integer to its numerical representation

*continued on next page*

**A–8 Appendix A** MATLAB Special Characters, Reserved Words, and Functions

num2str(a,n) Convert a number to its numerical representation with n

decimal places

sprintf(...) Format a string result

sscanf(...) Formatted input conversion

strcmp(s1, s2) Compare two strings—returns true if equal

strcmpi(s1, s2) Compare two strings without regard to case—returns true if equal

textscan Scan a text string

## Time-Related Functions Description

clock Determine the current time on the CPU clock

etime Find elapsed time

pause Pause the execution of a program, either until any key is hit or for a specified number of seconds

tic Start a timing sequence

toc Stop a timing sequence and returns the elapsed time

## Numerical Methods Description

diff(v) Compute the differences between adjacent values in a vector

interp1 Compute linear and cubic interpolation

interp2 Compute linear and cubic interpolation

interp3 Compute linear and cubic interpolation polyfit(x, y, n) Compute a least-squares polynomial polyval(c, x) Evaluate a polynomial

spline(x, y) Spline interpolation

## Program Control Description

break A command within a loop module that forces control to the statement following the innermost loop

case A specific value alternative within a switch statement

catch End of a suspect code block where the exception is trapped

continue Skip to the end of the innermost loop, but remains inside it

else Within an if statement, begin the code block executed when the condition is false

elseif Within an if statement, begin a subsequent test when the result of

<expression> the previous test is false

end Terminate a function specification or an if, switch, for, while, or catch block.

end When indexing, the value of the last element in an index vector

for var = v A code block repeated as many times as there are elements in the

vector v

function Identify an m-file as a function or begin a helper function within a

function file

# **Appendix A** MATLAB Special Characters, Reserved Words, and Functions **A–9**

error(str) Throw an exception to announce an error with the string provided

global var Define a variable as globally accessible

if <expression> Begin a conditional module—the following code block is executed if the logical expression is true

lasterror Provide a structure describing the environment from which an

exception was thrown

nargin Determine the number of input parameters actually supplied by a function’s caller

nargout Determine the number of output parameters actually requested by a function’s caller

otherwise Catch-all code block at the end of a switch statement

switch <variable> Begin a code module selecting specific values of the variable

(must be countable)

try Begin a block of suspect code from which an exception might be thrown

while <expression> A code module repeated as long as the logical expression is true

## Data Class Operations Description

char(...) Cast to a character type class(<object>) Determine the data type of an object double(a) Cast a to type double

int8/16/32/64(a) Cast a to integer type with the specified number of bits uint8/16/32/64(a) Cast a to unsigned integer type with the specified number of bits isa(obj, str) Test for a given data type

ischar(ch) Determine whether the given object is of type char iscell(...) Determine whether the given object is a cell isempty(a) Test for the empty vector [ ]

islogical(a) Determine whether the given object is of type logical isnumeric(a) Determine whether the given object is of type double isspace(a) Test for the space character

isstruct(a) Determine whether the given object is a structure

## Structure Operations Description

fieldnames(str) Return a cell array containing strings that are the names of the fields in the structure

getfield(str, Extract the value of the field

field)

isfield(str, field) Return true if the string is a field in the specified structure

str = rmfield Return a copy of the given structure with the given field removed

(str, field)

str = setfield Construct a structure in which the value of the field has been changed

(str, field, to the given value

value)

struct(...) Construct a structure from <fieldname> <value> pairs of

parameters