		X	Y	Z
33	!	separator		
34	"	.' (transpose) / permute	rot90	system	full
35	#	for	repmat	repelem (run-length decoding)	blanks
36	$	specify outputs	display stack (debug)		fopen, fwrite, fclose
37	%	specify inputs		char(vpa(...))	fopen, fread, fclose
38	&	comment	class	cast	typecast
39	'		intersect	and	bitand
40	(	*Not used. String delimiter*		run-length encoding	now / clock
41	)	( ) assignment indexing / split	{ } assignment indexing	( ) assignment ind. with final : / split	( ) assignment ind. with initial : / split
42	*	( ) reference indexing	{ } reference indexing	( ) reference ind. with final :	( ) refererence ind. with initial :
43	+	.*	kron	matrix product	Cartesian product
44	,	+	conv	conv2	conv2(..., 'same')
45	-		cos	sin	tan
46	.	-	setdiff	deconv	
47	/	break	continue	pause	bitget
48	0	./	angle	matrix /	unwrap
49	1	*Not used*	predefined literals	predefined literals	
50	2	*Not used*	predefined literals	predefined literals	
51	3	*Not used*	predefined literals	predefined literals	
52	4	*Not used*	predefined literals	predefined literals	
53	5	*Not used*	predefined literals		
54	6	*Not used*	predefined literals		
55	7	*Not used*	predefined literals		
56	8	*Not used*	predefined literals		
57	9	*Not used*	predefined literals		
58	:	colon (function)	linearize array	comma-separated list	bitset
59	;		acos	asin	atan2
60	<	<	min	cummin	
61	=	==	isequal	strcmp	strncmp
62	>	>	max	cummax	
63	?	if		why	sparse
64	@	push "for" value / "while" index	push "for" index	perms	randperm
65	A	all	all(..., 1)	dec2base. Larger base, any symbols	base2dec. Larger base, any symbols
66	B	logical(dec2bin(...)-'0')	bin2dec(char(...+'0'))	dec2bin	bin2dec
67	C		histcounts	im2col	im2col(..., 'distinct')
68	D	disp(num2str(..., ...))	disp(num2str(...))	sprintf / fprintf	disp
69	E	multiply by 2	replace elements in array		
70	F	*Not used. False (literal)*			
71	G	Paste from clipboard G (user-input)	plot	imwrite / imagesc / image / imshow	appearance of graphics / format
72	H	Paste from clipboard H	Copy to clipboard H		
73	I	Paste from clipboard I	Copy to clipboard I		
74	J	Paste from clipboard J	Copy to clipboard J		
75	K	Paste from clipboard K	Copy to clipboard K		
76	L	Paste from clipboard L (multi-level)	Copy to clipboard L (multi-level)	gallery	
77	M	Paste from clipboard M (function-input)	mode		
78	N	stack size	nchoosek (first input: array)	NaN	isnan
79	O	zeros	datestr	datenum	datevec
80	P	flip	flipud	pi	pdist2
81	Q	increment by 1	accumarray		polyval
82	R	triu	triu(...,1)	tril	tril(...,-1)
83	S	sort	sortrows	circshift	sign
84	T	*Not used. True (literal)*		toeplitz	
85	U	str2num / string to array	str2double		
86	V	num2str			
87	W	2 raised to input			
88	X	*Not used*	regexp	regexprep	
89	Y	*Not used*		inf	isinf
90	Z	*Not used*			
91	[	*Not used. Array delimiter*	ind2sub		
92	\	mod	mod(...-1)+1	matrix \	
93	]	end (loops or conditional branches)	sub2ind		
94	^	.^	sqrt	matrix ^	Cartesian product
95	_	unary minus			
96	`	do...while	while	tic	toc
97	a	any	any(..., 1)	padarray	base2base
98	b	bubble		strsplit	
99	c	char (also for cell array)	cat	strcat	strjoin
100	d	diff	diag / spdiags	blkdiag	gcd
101	e	reshape / squeeze			exp
102	f	find	strfind	factor	
103	g	logical / cell2mat	ndgrid		gammaln
104	h	horzcat	{..., ...}	hankel	hypergeom
105	i	input	urlread	imread	
106	j	input(...,'s')	real	imag	conj
107	k	lower / floor	upper / ceil	closest values	
108	l	ones		log. With two inputs, specifies base	log2
109	m	ismember	ismember(...,'rows')	mean	lcm
110	n	numel	nchoosek (first input: numbers)	interp1	
111	o	double / cell array to numeric array	int64	round / change case	fix
112	p	prod	prod(..., 1, ...)	cumprod	isprime / totient function
113	q	decrement by 1	quantile	n-th prime / next prime	primes
114	r	rand	randn	randi	randsample
115	s	sum	sum(..., 1, ...)	cumsum	std
116	t	duplicate elements			strrep
117	u	unique	unique(...'rows')		strjust
118	v	vertcat	remove all blanks	strtrim	deblank
119	w	swap			
120	x	delete from stack	clc		
121	y	duplicate element	eye	hypot	size
122	z	nnz	nonzeros		
123	{	*Not used. Cell array delimiter*	num2cell	mat2cell	mat2cell(x,ones(size(x,1),1),size(x,2))
124	|	abs / norm / determinant	union	or	bitor
125	}	else			split array
126	~	Not	setxor	xor	bitxor