

# GNU APL Reference Card

(for GNU APL version 1.8)

## Emacs mode

### Interaction mode:

beginning of defun	C-M-a
end of defun	C-M-e
find function at point	M-.
apropos symbol	C-c C-a
edit function	C-c C-f
show help for symbol	C-c C-h
finnapl list	C-c TAB
show keyboard	C-c C-k
plot line	C-c RET
edit variable	C-c C-v
trace	C-c C-.

### Edit mode:

go to beginning of defun	C-M-a
go to end of defun	C-M-e
find function at point	M-.
apropos symbol	C-c C-a
interactive send current function	C-c C-c
help for symbol	C-c C-h
finnapl list	C-c TAB
show keyboard	C-c C-k
interactive send buffer	C-c C-l
interactive send region	C-c C-s
switch to interactive	C-c C-z
trace	C-c C-.
indent	C-M-q

## System

Notation for commands:

F	filename	L	library	P	path
G	logging facility	O	object	S	symbol
W	workspace				

### APL standard commands

check workspace intergity	)CHECK
clear workspace	)CLEAR
save workspace as CONTINUE and exit	)CONTINUE
copies objects from given workspace	)COPY [L] W [0 ...]
remove W	)DROP [L] W
dump W (readable, HTML escaped)	)DUMP-HTML [[L] W]
dump W (readable APL)	)DUMP [[L] W]
dump W (readable APL, verbose)	)DUMPV [[L] W]
erase symbol(s)	)ERASE S ...
show functions	)FNS [from-to]
help	)HELP [primitive]
history	)HIST [CLEAR]
runs command on host	)HOST command ...
loads workspace (IBM .atf format)	)IN F [0 ...]
show libraries and paths	)LIBS [[L] path]
show saved workspaces	)LIB [L P] [from-to]
load workspace W	)LOAD [L] W

show more error info	)MORE
lists symbols matching name	)NMS [from-to]
quit APL	)OFF
show operators	)OPS [from-to]
dump workspace (IBM .atf format)	)OUT name [0 ...]
protects during copying	)PCOPY [L] W [0 ...]
protects during loading	)PIN F [0 ...]
quiet load	)QLOAD [[L] W]
reset state indicator	)RESET
save workspace as W	)SAVE [[L] W]
clear suspended functions	)SIC
see suspended functions and locals	)SINL
see suspended functions	)SIS
state indicator	)SI
show symbol count	)SYMBOLS [count]
show values in use by interpreter	)VALUES
show variables	)VARS [from-to]
get/set workspace ID	)WSID [W]

### GNU extension commands (mostly for debugging)

toggles boxing of values when printing	]BOXING [OFF num]
toggle colored output	]COLOR [ON OFF]
dump W in HTML file	]DOXY [path]
expected error count in test suite	]EXPECT error_count
help	]HELP [primitive]
show keyboard layout	]KEYB
as )LIB, but shows fil extensions	]LIB [L P] [from-to]
show/set logging facilities	]LOG [G [ON OFF]]
next testcase file	]NEXTFILE
FIXME:	]OWNERS
performance statistics	]PSTAT [CLEAR SAVE]
as )SIS, with more details	]SIS
as )SI, with more details	]SI
shared variables	]SVARS
describe internal details of symbol S	]SYMBOL S
define user command	]USERCMD [ ... ]
toggle output coloring on console	]XTERM [ON OFF]

### System variables:

character input/output	<input type="checkbox"/> I
evaluated input/output	<input type="checkbox"/>
account information	<input type="checkbox"/> AI
command line arguments	<input type="checkbox"/> ARG
atomic vector	<input type="checkbox"/> AV
comparison tolerance	<input type="checkbox"/> CT
event message	<input type="checkbox"/> EM
event type	<input type="checkbox"/> ET
format control	<input type="checkbox"/> FC
index origin (indexes start: 1, can be set to 0)	<input type="checkbox"/> IO
left argument	<input type="checkbox"/> L
line counters	<input type="checkbox"/> LC
latent expression (executed when workspace is loaded)	<input type="checkbox"/> LX
print precision (number of digits)	<input type="checkbox"/> PP
print style	<input type="checkbox"/> PS
print width (max characters in each printed line)	<input type="checkbox"/> PW
right argument	<input type="checkbox"/> R
random link	<input type="checkbox"/> RL

shared variable event	<input type="checkbox"/> SVE
system limits	<input type="checkbox"/> SYL
terminal control characters	<input type="checkbox"/> TC
time stamp (current time)	<input type="checkbox"/> TS
time zone (offset from GMT)	<input type="checkbox"/> TZ
user load	<input type="checkbox"/> UL
axis argument	<input type="checkbox"/> X
workspace available (bytes for workspace)	<input type="checkbox"/> WA
dfn axis argument	X
dfn result	λ
dfn left value arg	α
dfn left function arg	α
dfn right value arg	ω
dfn right function arg	ω

### System functions:

atomic function	<input type="checkbox"/> AF
attributes	<input type="checkbox"/> AT
char representation	<input type="checkbox"/> CR
delay	<input type="checkbox"/> DL
D. Knuth's dancing links	<input type="checkbox"/> DLX
execute alternate	<input type="checkbox"/> EA
execute both	<input type="checkbox"/> EB
execute controlled	<input type="checkbox"/> EC
environment	<input type="checkbox"/> ENV
event simulate	<input type="checkbox"/> ES
expunge	<input type="checkbox"/> EX
fast Fourier transform	<input type="checkbox"/> FFT
file I/O	<input type="checkbox"/> FIO
FX	<input type="checkbox"/> FiX
Gtk GUI	<input type="checkbox"/> GTK
MAP ravel elements	<input type="checkbox"/> MAP
input from script	<input type="checkbox"/> INP
name association	<input type="checkbox"/> NA
name class	<input type="checkbox"/> NC
name list	<input type="checkbox"/> NL
plot a graph	<input type="checkbox"/> PLOT
regular expression, regex <input type="checkbox"/> RE string	<input type="checkbox"/> RE
random APL value	<input type="checkbox"/> RVAL
state indicator	<input type="checkbox"/> SI
SQL functions	<input type="checkbox"/> SQL
shared variable control	<input type="checkbox"/> SVC
shared variable offer	<input type="checkbox"/> SVO
shared variable query	<input type="checkbox"/> SVQ
shared variable retraction	<input type="checkbox"/> SVR
shared variable state	<input type="checkbox"/> SVS
STOP vector	<input type="checkbox"/> STOP
transfer form	<input type="checkbox"/> TF
TRACE vector	<input type="checkbox"/> TRACE
unicode character	<input type="checkbox"/> UCS

## Notation

comment	⍝
statement separator	◇
assignment	A← ...
assignment	(A B C)← ... ..
function definition	▽
zilde (empty vector)	⊖

a	+ a
a + b	a + b
- a	- a
a - b	a - b
magnitude of a	a
b mod a	a   b
signal (-1, 0, +1)	× a
ab	a × b
1/a	÷ a
a/b	a ÷ b
floor of a	⌊a
min(a,b)	a⌊b
ceiling of a	⌈a
max(a,b)	a⌈b
$e^a$	* a
$a^b$	a * b
log(a)	⊗ a
log <sub>b</sub> (a)	b ⊗ a
first <i>n</i> non-negative integers	ι n

a = b	a = b
a < b	a < b
a > b	a > b
a ≤ b	a ≤ b
a ≥ b	a ≥ b
expression max depth	≡ a
match (value and type)	a ≡ b
expression min depth	≠ a
not match	a ≠ b
not a	≈ a
a or b	a ∨ b
a and b	a ∧ b
a nor b	a ∸ b
a nand b	a ⋈ b
a ∈ b ?	a ∈ b
find a in b (binary index)	a ⊆ b ?

a!	!a
$\binom{b}{a}$	a!b
aπ	⊙a
circle (trig) function	a ⊙ b
random integer in [1,a]	?a
a distinct random integers in [1,b]	a?b

makes a vector out of A	, A
append B to A	A,B
number of components in each dimension of A	ρ A
array with shape A and data elements B	Aρ B
inverse matrix of A	$\begin{bmatrix} \cdot \\ \cdot \end{bmatrix} A$
$B^{-1}A$ (solution to $Bx = A$ )	A $\begin{bmatrix} \cdot \\ \cdot \end{bmatrix} B$
reverse elements of A (1 <sup>st</sup> index)	⊖A
rotate B by A positions	A⊖B
reverse elements of A (last index)	⊖A
rotate B by A positions (last index)	A⊖B
drop first A elements of B	A↓B
select first A elements of B	A↑B
intersection	A∩B

set (remove duplicates)	⋃A
union	A⋃B
identity	⊢A
take right hand side (B)	A⊢B
null	¬A
take left hand side (A)	A¬B
i-th element of A	A[i]
elements of A with indices i, j, k, ...	A[i j k ...]
element of A w/indices i, j, ... in 1 <sup>st</sup> dimension, k, l, ... in second, ...	A[i ...; k ...; ...]

transpose of A	⌊A
transpose of B, axes ordered by A	A⌊B
maps A: 1 for a∈ B, 0 for a∉ B	A∈B
grade up A	⬆A
grade up B with elements of A as top priority	A⬆B
grade down A	⬇A
grade down B with elements of A as low priority	A⬇B
transpose of A	⌊A
enclose A	⊂ A
enclose B with selected elements given the binary vector A	A⊂ B
disclose A	⊂ A
recursively pick elements of B given the indices in A	A⊂ B

Decode single digits of B with respect to base A	A⊥B
Encode B with respect to bases given by A	A⊤B

line label A	A:
branch to line A	→A

execute APL expression A	⌘A
format A as chars	⌞A

user input	□
------------	---

system var/function	□
---------------------	---

reduce op over array A	op/A
compress: select B using A as mask	A/B
A/B on last dimension	A≠B
expand: insert zeros in B using A as mask	A\B
A\B on last dimension	A↖B
inner product with functions f, g	Af.gB
outer product with function f	A○.fB
for each b∈B, apply: Ab	A”B
axis: AfC, over Bth axis	Af[B]C
duplicate/commute	
compose	A○B

## Circle function

A	A○B	A	A○B
---	-----	---	-----

$$0 \quad \sqrt{1-B \times B}$$

¬1	arcsin B	1	sin B
¬2	arccos B	2	cos B
¬3	$\arctan B$	3	tan B
¬4	$\sqrt{-1+B \times B}$	4	$\sqrt{1+B \times B}$
¬5	arcsinh B	5	sinh B
¬6	arccosh B	6	cosh B
¬7	arctanh B	7	$\tanh B$
¬8	¬(8○B)	8	$\pm\sqrt{-1+B \times B}$
¬9	B	9	real part of B
¬10	+B	10	B
¬11	0J1×B	11	imag part of B
¬12	*0J1×B ( $e^{iB}$ )	12	arc B (phase of B)

For A= 8, the sign before the square root is opposite of *B*.

## Function Definition

Example:  $f(d,v) = (v_1^d + \dots + v_n^d)^{1/d}$

### Dynamic function definition (dfn):

α is the left argument, ω is the right argument.

f ← { ( +/ω\*α ) \* (÷α) }

### Traditional function definition (tradfn):

∇: begin/end defun. “∇R ← A f B ;U ;V” is “f takes left arg A, right arg B, has local vars U, V, and returns result in R”.

∇res ← d f v ;sq ;sum

sq ← v \* 2

sum ← +/sq

res ← sum\*(÷d)

∇

Copyright © 2020 Jeronimo Pellegrini, Lucas S. Vieira  
designed by Jeronimo Pellegrini, Lucas S. Vieira and Stephen Gildea,  
for GNU APL

Released under the terms of the GNU General Public License version 3 or later. For more APL documentation, check the GNU APL site. For the T<sub>E</sub>X source for this card, see:

<https://www.github.com/jpellegrini/gnu-apl-refcard>