

⌈ The Co-dfns Compiler

⌈ High-performance, Parallel APL Compiler

⌈ Copyright © 2011-2017 Aaron W. Hsu arcfide@sacrideo.us

⌈

⌈ This program is free software: you can redistribute it and/or modify it under the terms of the GNU Affero

⌈ General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

⌈

⌈ This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the

⌈ implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Affero

⌈ General Public License for more details.

⌈

⌈ You should have received a copy of the GNU Affero General Public License along with this program.

⌈ If not, see <http://www.gnu.org/licenses/>

:Namespace CODFNS

```
⌈IO ⌈ML ⌈WX ← 0 1 3
VERSION ← 2018 1 0
AFΔPREFIX ← '/usr/local'
AFΔLIB ← ''
VSΔPS ← '\2017\o, "Enterprise' Professional' Community', " C\VC\Auxiliary\Build'
VSΔPS ,← C' 14.0\VC'
VSΔPS , "← C\Program Files (x86)\Microsoft Visual Studio'
VSΔPS , "← C\vcvarsall.bat'

Cmp ← { _ ← 1 ⌈DELETE α, soext θ
      _ ← (⌈opsys'vsc'gcc'clang') α ⌈ (α, '.cpp') put~ gc tt ⊃ a n ← ps ω
      _ ← ⌈NEXISTS α, soext θ ⌈ _ : n
      'COMPILE ERROR' ⌈SIGNAL 22}

MkNS ← {NS ⌈ α ⌈ {NS.⌈α mkf ω} (1 = 1 ⌈ ⌈ω) ≠ 0 ⌈ ⌈ω ⌈ NS ← #.⌈NS θ}
Fix ← {α MkNS α Cmp ω}
Xml ← {⌈XML (0 ⌈ ⌈ω), (⌈⌈⌈ 2 ↑ 1 ↓ ⌈ω), (C"), ⌈(C(⌈3 + ≠⌈ω) ↑, "nrsgvyel'), ⌈⌈⌈ ↓ ⌈⌈, "⌈ 3 ↓ ⌈ω}
MKA ← {mka ⊂ ω ⌈ 'mka' ⌈NA 'P ', (α, soext θ), '|mkarray <PP'}
EXA ← {exa θ ω ⌈ 'exa' ⌈NA (α, soext θ), '|exarray >PP P'}
FREA ← {frea ω ⌈ 'frea' ⌈NA (α, soext θ), '|frea P'}

opsys ← {ω ⊃ "Win'Lin'Mac' ⊂ 3 ↑ ⊃ '.' ⌈WG'APLVersion'}
soext ← {opsys'.dll'.so'.dylib'}
tie ← {0 :: ⌈SIGNAL ⌈EN ⌈ 22 :: ω ⌈NCREATE 0 ⌈ 0 ⌈NRESIZE ω ⌈NTIE 0}
put ← {s ← (⌈128 + 256 | 128 + 'UTF-8' ⌈UCS α) ⌈NAPPEND (t ← tie ω) 83 ⌈ 1 : r ← s ⌈ ⌈NUNTIE t}
mkf ← {fn ← (α, soext θ), '|', (⌈'R'⌈'⌈ω), '_dwa '
      f ← ω, '←{⌈'⌈'dya'⌈'NA'⌈', fn, '>PP <PP <PP'⌈' ⌈ '
      f, '←'⌈'mon'⌈'NA'⌈', fn, '>PP P <PP'⌈' ⌈ '
      f, '0=⌈NC'⌈'α'⌈:mon 0 0 ω ⌈ dya 0 α ω} ⌈ 0}
```

```

ccf ← { ' -o ' ', ω, ' ', α, ' ' ' ', ω, ' .cpp ' ' -laf', AFΔLIB, ' > ', ω, ' .log 2>&1' }
cci ← { '-I' ', AFΔPREFIX, '/include' ' ' -L ' ', AFΔPREFIX, '/lib' ' ' }
cco ← '-std=c++11 -Ofast -g -Wall -fPIC -shared'
ucc ← { ωω (□SH αα, ' ', cco, cci, ccf) ω }
gcc ← 'g++' ucc 'so'
clang ← 'clang++' ucc 'dylib'
vsco ← { z ← '/W3 /wd4102 /wd4275 /Gm- /O2 /Zc:inline /Zi /Fd"', ω, ' .pdb" '
z, ← '/errorReport:prompt /WX- /MD /EHsc /nologo '
z, '/I"%AF_PATH%\include" /D "NOMINMAX" /D "AF_DEBUG" ' }
vslo ← { z ← '/link /DLL /OPT:REF /INCREMENTAL:NO /SUBSYSTEM:WINDOWS '
z, ← '/LIBPATH:"%AF_PATH%\lib" /DYNAMICBASE "af', AFΔLIB, ' .lib" '
z, '/OPT:ICF /ERRORREPORT:PROMPT /TLBID:1 ' }
vsc0 ← { ~v ≠ b ← □NEXISTS VSΔPS: 'MISSING VISUAL C++' □SIGNAL 99 ◇ '"', '" amd64', ~ ⊃ b ≠ VSΔPS }
vsc1 ← { ' && cd "', (⊃ □CMD 'echo %CD%'), '" && cl', (vsco ω), '/fast "', ω, ' .cpp" ' }
vsc2 ← { (vslo ω), '/OUT:"', ω, ' .dll" > "', ω, ' .log"' }
vsc ← { □CMD ('%comspec% /C ', vsc0, vsc1, vsc2) ω }

```

```

fΔ ← 'ptknrsgvyeld' ◇ NΔ ← 'ABEFGMLNOPVZ'
□FX0 □0 fΔ, " ← { 'o', " (□1 ≠ fΔ), " C' > ω }'
□FX0 □0 NΔ, " m ← { 'o', " (□1 ≠ NΔ), " C' = tω }'
□FX0 □0 'GLM', " ← { ; / (0 'o', " (□1 NΔ i 'GLM'), " C' 0 0 ), 1 + @0 ◇ tω }'
□FX0 □0 'ABEFO', " ← { ; / (0 'o', " (□1 NΔ i 'ABEFO'), " C' αα 0 ), 1 + @0 ◇ tω }'
□FX0 □0 'NPVZ', " ← { 0 (NΔ i ' 'o', " 'NPVZ', " ' ' 'o', " 0 (±ω) ' 0 (cω) ' αα (cω) ' 1 (cω) ' , " }'
□FX0 □0 NΔ, " C's ← { ω }' ◇ at ← { α α ω ω } ◇ new ← { ω } ◇ wrap ← { ω }

```

```

Display ← { α ← 'Co-dfns' ◇ W ← wnew Cα ◇ 777 :: wdel W
wdel W ⊢ W αα { wclose α : ⊕ □SIGNAL 777' ◇ α α α } * ω ω ⊢ ω }
LoadImage ← { α ← 1 ◇ □loading θ ω α }
SaveImage ← { α ← 'image.png' ◇ saveimg (□ω) α }
Image ← { ~ 2 3 v. = ≠ ρω : □SIGNAL 4 ◇ (3 ≠ 2 ⊃ 3 ↑ ρω) ∧ 3 = ≠ ρω : □SIGNAL 5 ◇ ω ⊢ wimg (□ω) α }
Plot ← { 2 ≠ ≠ ρω : □SIGNAL 4 ◇ ~ 2 3 v. = 1 ⊃ ρω : □SIGNAL 5 ◇ ω ⊢ wplot (□ω) α }
Histogram ← { ω ⊢ whist ω, α }

```

```

▽ r ← List
r ← □NS " 1 ρ C ⊃ θ ◇ r.Name ← , " C'Compile' ◇ r.Group ← C 'CODFNS'
r[0].Desc ← 'Compile an object using Co-dfns'
r.Parse ← C '2S -af=cpu opencl cuda '

```

▽

```

Convert ← { α (□SE.SALT.Load '[SALT]/lib/NStoScript -noname').ntgennscode ω }
Run ← { C I ← ω ◇ in out ← I.Arguments ◇ AFΔLIB ◇ ← I.af " ⊃ ~ I.af ≡ 0
S ← (C ':Namespace ', out), 2 ↓ 0 0 0 out Convert ##.THIS. ⊕ in
'Compile' ≡ C : { } { _ ← { ##.THIS. ⊕ out, ' ← ω } out Fix S ⊢ □EX'##.THIS.', out
□CMD 'copy "%CUDA_PATH%\nvvm\bin\nvvm64*" /Y' / ~ (I.af ≡ 'cuda') ∧ opsys 1 0 0 } θ }
Help ← { Usage: <object> <target> [-af={cpu,opencl,cuda}] }

```

```

▽ Z ← GfxΔInit S
'wnew' □NA 'P ', (S, soext θ), '|wnew <C[ ]'
'wclose' □NA 'I ', (S, soext θ), '|wclose P'
'wdel' □NA (S, soext θ), '|wdel P'
'wimg' □NA (S, soext θ), '|wimg <PP P'
'wplot' □NA (S, soext θ), '|wplot <PP P'
'whist' □NA (S, soext θ), '|whist <PP F8 F8 P'
'loading' □NA (S, soext θ), '|loading >PP <C[ ] I'
'saveimg' □NA (S, soext θ), '|saveimg <PP <C[ ]'
Z ← 0 0 ρ θ

```

▽

```
_o ← {0 ≥ ⊃ c a e r ← p ← ααω : p ∘ 0 ≥ ⊃ c a e r2 ← p ← α ω ω : p ∘ c a e(r ↑ ~ - [ / ≠ ·· r r2) }
_s ← {0 < ⊃ c a e r ← p ← ααω : p ∘ 0 < ⊃ c2a2e r ← p ← e ω ω r : p ∘ (c [ c2)(a, a2)e r}
_noenv ← {0 < ⊃ c a e r ← p ← ααω : p ∘ c a α r}
_env ← {0 < ⊃ c a e r ← p ← ααω : p ∘ c a(e ω ω a) r}
_then ← {0 < ⊃ c a e r ← p ← ααω : p ∘ 0 < ⊃ c a e _ ← p ← e(ω ω _ s eot) a : p ∘ c a e r}
_not ← {0 < ⊃ c a e r ← ααω : 0 a α ω ∘ 2 a α ω}
_as ← {0 < ⊃ c a e r ← ααω : c a e r ∘ c(⊂ ω ω a)e r}
_t ← {0 < ⊃ c a e r ← ααω : c a e r ∘ e ω ω a : c a e r ∘ 2 θ α ω}
_ign ← {c a e r ← ααω ∘ c θ e r}
_peek ← {0 < p ← ⊃ ααω : p ∘ 0 θ α ω}
_yes ← {0 θ α ω}
_opt ← {α(αα_o_yes) ω}
_any ← {α(αα_s ∇ _o_yes) ω}
_some ← {α(αα_s(αα_any)) ω}
_set ← {(0 ≠ ≠ ω) ∧ (∩ ω) ∈ αα : 0( , ∩ ω ) α (1 ↓ ω) ∘ 2 θ α ω}
Tk ← {(≠ , αα) ↑ ω) ≡ , αα : 0(⊂ , αα) α ((≠ , αα) ↓ ω) ∘ 2 θ α ω}
_eat ← {0 = ≠ ω : 2 θ α ω ∘ 0(αα ↑ ω) α (αα ↓ ω)}


ws ← (' ', ⬤UCS9)_set
aws ← ws_any_ign
awslf ← (⬤UCS1013)_set_o ws_any_ign
gets ← aws_s('←'_tk)_s aws_ign
him ← '-'_set
dot ← '.'_set
jot ← '°'_set
lbrc ← aws_s('{ '_set)_s aws
rbrc ← aws_s('}'_set)_s aws
lpar ← aws_s('('_tk)_s aws_ign
rpar ← aws_s(')'_tk)_s aws_ign
lbrk ← aws_s('[ '_tk)_s aws_ign
rbrk ← aws_s(']'_tk)_s aws_ign
semi ← aws_s(';'_tk_as('ª' V o, o ∩ ))_s aws
grd ← aws_s(':', _tk)_s aws_ign
egrđ ← aws_s('::_tk)_s aws_ign
alpha ← 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyzΔ'_set
digits ← '0123456789'_set
prim ← (prims ← '+-÷×|*⦿[ !<≤=≧≥>^v~̣?ρ, ;ϕθΘεξζιο~≢⇌→/÷\ʎτ↑↓↯n♠▽◻')_set
mop ← "'/' ÷ \ " _set
dop₁ ← '. * °'_set
dop₂ ← '¨ ¨ * °'_set
dop₃ ← '°'_set
eot ← aws_s{" ≡ ω : 0 θ α " ∘ 2 θ α ω}_ign
digs ← digits_some
odigs ← digits_any
int ← aws_s_digs_s(him_opt)_s aws
float ← aws_s(odigs_s_dot_s_int_o(digs_s_dot))_s aws
name ← aws_s(alpha_o(digits_some_s_alpha)_some)_s aws
aw ← aws_s('αω'_set)_s aws
aaww ← aws_s(('αα'_tk)_o('ωω'_tk))_s aws
sep ← aws_s (('♦', ⬤UCS1013)_set_ign)_s aws
nss ← awslf_s(:Namespace'_tk)_s aws_s(name_opt)_s awslf_ign
nse ← awslf_s(:EndNamespace'_tk)_s awslf_ign
```

$Sfn \leftarrow aws_s (('TFF\Box'_{tk})_o ('TFFI\Box'_{tk}))_s aws_as \{P \Phi \in \omega\}$
 $Prim \leftarrow prim_as P$
 $Vt \leftarrow (\vdash \text{ } \vdash 0 \supset \neg) \supset \neg 1, \sim 1 \supset \neg$
 $Var \leftarrow \{\alpha (aaww_o aw_o (name_as \Phi)_t (\alpha \alpha = Vt)_as (\alpha \alpha V_o, o \supset)) \omega\}$
 $Num \leftarrow float_o int_as (N_o \Phi)$
 $Strand \leftarrow 0 Var_s (0 Var_some)_as (3 A_o \Phi)$
 $Pex \leftarrow \{\alpha (rpar_s Ex_s lpar) \omega\}$
 $Atom \leftarrow Strand_o (0 Var_as (1 A))_o (Num_some_as (0 A_o \Phi))_o Pex$
 $Brk \leftarrow rbrk_s \{\alpha (Ex_opt_s (semi_s (Ex_opt)_any)) \omega\}_s lbrk_as (3 E_o \Phi)$
 $Idx \leftarrow Brk_s (_yes_as \{P, '[]'\})_s Atom_as (2 E_o \Phi)$
 $Blrp \leftarrow \{\alpha (\alpha \alpha_s (\omega \omega Slrp \nabla)) \omega\}$
 $Slrp \leftarrow \{\alpha (\alpha \alpha_o (\omega \omega_s \nabla)_o ((1_eat)_s \nabla)) \omega\}$
 $Fa \leftarrow \{$
 $\quad e \leftarrow (\mathcal{C} ' \omega \omega ' \alpha \alpha ', ' \alpha \omega ')_o, o \mathcal{C} \downarrow \mathbb{Q} \neg 1 + 3 3 2 2 \top (6 4 4 \neq 1 5 9) + 2 \times \imath 14$
 $\quad a \leftarrow \downarrow \mathbb{Q} \uparrow (e, \text{ } \text{ } \mathcal{C} \alpha) Gex_o Ex_o Fex Stmts_then Fn \text{ } \mathcal{C} \omega$
 $\quad m \leftarrow (0 = \supset a) \wedge \wedge \neq (\vee \neg o, = \sim \imath 14) \vee o, \neq \sim 1 \supset a$
 $\quad \sim \vee \neq m : (\neg \supset a) \theta \alpha \omega$
 $\quad (1 = + \neq m) \wedge 2 > m \imath 1 : 0 (\mathcal{C} 0 (N \Delta \imath 'F') 1 0 \neg 1 + @_0 \supset \supset m \neq 1 \supset a) \alpha \omega$
 $\quad z \leftarrow \neg \uparrow (\mathcal{C} 0 (N \Delta \imath 'F') \neg 1 0), (\{1 (N \Delta \imath 'F') \omega 0\} \neg 1 + m \neq \imath 14) \neg (2 + @_0 \supset) m \neq 1 \supset a$
 $\quad 0 (\mathcal{C} z) \alpha \omega\}$
 $Fn \leftarrow \{$
 $\quad ns \leftarrow (n z) \neq m \leftarrow (F_m \wedge \neg 1 \in \sim k) \vdash z \leftarrow \neg \uparrow \omega \diamond 0 = \neq ns : 0 (\mathcal{C} z) \alpha \text{ } "$
 $\quad r \leftarrow \downarrow \mathbb{Q} \uparrow \alpha \circ Fa \text{ } ns \diamond 0 < c \leftarrow \neg \supset r : c \theta \alpha \omega$
 $\quad z \leftarrow (\mathcal{C} \text{ } z) ((\supset \neg) \neg @_{(m)}) \neg \downarrow (m \neq p z) + @_0 \mathbb{Q} \uparrow \supset \neg 1 \supset r$
 $\quad 0 (\mathcal{C} z) \alpha \text{ } "\}$
 $Pfe \leftarrow \{\alpha (rpar_s Fex_s lpar) \omega\}$
 $Bfn \leftarrow rbrk Blrp lbrk_as \{0 (N \Delta \imath 'F') \neg 1 (\mathcal{C} \Phi 1 \downarrow \neg 1 \downarrow \omega)\}$
 $Fnp \leftarrow Prim_o (1 Var)_o Sfn_o Bfn_o Pfe$
 $Mop \leftarrow \{\alpha ((mop_as P)_s Afx_as (1 O)) \omega\}$
 $Dop_1 \leftarrow \{\alpha ((dop_1_as P)_s Afx_as (2 O_o \Phi)) \omega\}$
 $Dop_2 \leftarrow \{\alpha (Atom_s (dop_2_as P)_s Afx_as (2 O_o \Phi)) \omega\}$
 $Dop_3 \leftarrow (dop_3_as P)_s Atom_as (2 O_o \Phi)_o (dot_s jot_as (P_o \Phi)_as (1 O))$
 $Bop \leftarrow \{\alpha (rbrk_s Ex_s lbrk_s (_yes_as \{P, '[]'\})_s Afx_as (2 O_o \Phi)) \omega\}$
 $Afx \leftarrow Mop_o (Fnp_s (Dop_1_o Dop_3_opt)_as (\neg / \imath o \neq + @_0 \mathbb{Q} o \uparrow o \Phi))_o Dop_2_o Bop$
 $Trn \leftarrow \{\alpha (Afx_s ((Afx_o Idx_o Atom)_s (\nabla_opt)_opt)) \omega\}_as (3 F_o \Phi)$
 $Bind \leftarrow \{\alpha (gets_s (name_as \Phi)_env (\neg \neg \sim \alpha \alpha, \sim o \mathcal{C} \vdash)_as (0 (N \Delta \imath 'B') \alpha \alpha, o \mathcal{C} \vdash)) \omega\}$
 $Asgn \leftarrow gets_s Brk_s (name_as \Phi_t (0 = Vt)_as (0 V_o, o \supset))_as (4 E_o \Phi)$
 $Fex \leftarrow Afx_s (Trn_opt)_s (1 Bind_any)_as (\neg / \imath o \neq + @_0 \mathbb{Q} o \uparrow o \Phi)$
 $App \leftarrow Afx_s (Idx_o Atom_s (dop2_not)_opt)_as \{(\neq \omega) E \Phi \omega\}$
 $Ex \leftarrow Idx_o Atom_s \{\alpha (0 Bind_o Asgn_o App_s \nabla_opt) \omega\}_as (\neg / \imath o \neq + @_0 \mathbb{Q} o \uparrow o \Phi)$
 $Gex \leftarrow Ex_s grd_s Ex_as (G_o \Phi)$
 $Nlrp \leftarrow sep_o eot Slrp (lbrk Blrp rbrk)$
 $Stmts \leftarrow \{\alpha (sep_any_s (Nlrp_then (\alpha \alpha_s eot o \Phi))_any_s eot) \omega\}$
 $Ns \leftarrow nss Blrp nse_then (Ex_o Fex Stmts_then Fn)_s eot_as M$
 $ps \leftarrow \{0 \neq \supset c a e r \leftarrow \theta \theta Ns \in \{\omega / \sim \wedge \setminus ' \mathfrak{A} ' \neq \omega\} \text{ } \omega, \text{ } \Box \text{ } UCS 10 : \Box \text{ } SIGNAL c$
 $\quad (\downarrow s (\neg) @_3 \uparrow \supset a) e (s \leftarrow 0 (' \omega ') (' \alpha ') ' \alpha \alpha ' ' \omega \omega ' (\neg, n \sim \neg) \supset a)\}$

⊙	A	B	E	F	G	L	M	N	O	P	V	Z
⊙	0	1	2	3	4	5	6	7	8	9	10	11

$tt \leftarrow \{d \ t \ k \ n \leftarrow \omega \diamond I \leftarrow \{(\subset \omega) \ \square \ \alpha\} \diamond U \leftarrow \{\omega \omega \star^{-1} \alpha \alpha \ \omega \omega \ \omega\} \diamond D \leftarrow \{\omega \ (\vdash - 1 + \bot) \ \alpha[(\bot \neq \alpha) \sim \omega]\}$
 $\quad \leftarrow 2 \ \{0 \dashv l[\omega[i]] \leftarrow \omega[-1 + i \leftarrow \bot 0, 2 = \neq i] \dashv p[\omega] \leftarrow \alpha[i \leftarrow \alpha \ \bot \ \omega]\} \neq \vdash \circ \sqsubset \exists \ d \dashv p \leftarrow l \leftarrow \bot \neq d$

- ⊙ Drop unnamed top-level functions
- ⊙ Box mutated names
- ⊙ Resolve names

$bi \leftarrow \bot \ t = 1 \dashv vi \leftarrow \bot \ t = 10$

$gf \leftarrow \{p \ I@ \{3 \neq t[\omega]\} \star \equiv p[\omega]\}$

$bv \leftarrow \{\omega[bi \ \bot \ \omega]\}@ \{1 = t[\omega]\} \star \equiv \{\omega[\Delta p[\omega]]\} \ \bot \ 1 = t[p]$

$gn \leftarrow \{(i[b]) \ ((gf \ f)[b]) \dashv n[a] \leftarrow bv[v[a]] \dashv a \ b \leftarrow \bot \ \{\omega(\sim \omega)\} \ (\neq bi) > v \leftarrow 1 \ \bot \ 1 \sim \supset n[i] \ f \wedge .(\circ, =) \ \alpha \dashv i \ f \leftarrow \omega\}$

$n \leftarrow bi \ D \sim n \dashv (n[bi]) \ (gf \ bi) \ gn \star \{0 = \neq \supset \alpha\} \ vi \ (gf \ vi)$

$p \leftarrow bi \ D \sim I@ \{1 = t[\omega]\} \star \equiv \sim p$

$t \leftarrow t[nb] \dashv k \leftarrow k[nb \leftarrow \bot \ t \neq 1]$

- ⊙ Lift guard test expressions
- ⊙ Flatten expressions
- ⊙ Label jumps
- ⊙ Lift functions
- ⊙ Inline functions
- ⊙ Propagate constants
- ⊙ Fold constants
- ⊙ Dead, useless code elimination
- ⊙ Allocate frames
- ⊙ Create Init function
- ⊙ Declare functions
- ⊙ Sort AST
- ⊙ Flatten AST

}

```

E1 ← { 'fn' gcl ((C n, o ⊃ v), e, y) ω }
E2 ← { 'fn' gcl ((C n, o ⊃ v), e, y) ω }
Ei ← { r l f ← ⊃ v ω ◊ ((C n ω) ('fn' var) ⊃ ⊃ e ω), '=' , ((C n ω) ('fn' var) 1 ⊃ ⊃ e ω), ';' , nl }
O1 ← { 'op' gcl ((C n, o ⊃ v), e, y) ω }
O2 ← { 'op' gcl ((C n, o ⊃ v), e, y) ω }
O0 ← { }
Of ← { 'EF' (, ('Δ' □R' ___' ⊃ n ω), ', ', (C n ω), ', ') ; ', nl }
Fd ← { 'FP' (, (C n ω), ', ') ; ', nl }
F0 ← { 'DF' (, (C n ω), ' _f ) { ', nl, 'A*env[] = { tenv } ; ', nl }
F1 ← { 'DF' (, (C n ω), ' _f ) { ', nl, ('env0' dnv ω), (fnv ω) }
G0 ← { v ← (C n ω) (" var) 1 ⊃ ⊃ e ω
      'if (1 != cnt('v, ')) err(5); if('v, 'v.as(s32).scalar < I > ()) { ', nl }
G1 ← { 'z = ', ((C n ω) (" var) ⊃ ⊃ e ω), '; goto L, (Φ ⊃ l ω), '; } ; ', nl }
L0 ← { 'z = ', a, '; L, (Φ ⊃ n ω), ': ', (a ← (1 ⊃ ⊃ v ω) (" var) 1 ⊃ ⊃ e ω), '= z ; ', nl }
Z0 ← { } ; ', nl, nl }
Z1 ← { } ; ', nl, nl }
Ze ← { } ; ', nl, nl }
M0 ← { (rth θ), ('tenv' dnv ω), nl, 'A*env[] = { ', ((0 ≡ ⊃ ω) ⊃ 'tenv' 'NULL'), ', ' ; ', nl, nl }
S0 ← { ( (' { 'rk0, srk, 'DO(i, prk) cnt* = sp[i] ; ', spp, sfv, slp) ω ) }
Y0 ← { ⊃, / ((1 ≠ ⊃ n ω) ((¬ sts'' (C l), '' ⊃ s), '}', nl, ¬ ste'' (C n) var'' ⊃ r) ω), '}', nl }

gc ← { ⊃, / { 0 = ⊃ t ω : C 5 p θ ◊ C (Φ ⊃ t ω), Φ ⊃ k ω) ω } ö 1 ⊢ ω }

syms ← , " ' + ' ' - ' ' x ' ' ÷ ' ' * ' ' @ ' ' | ' ' o ' ' l ' ' f ' ' ! '
nams ← ' add ' ' sub ' ' mul ' ' div ' ' exp ' ' log ' ' res ' ' cir ' ' min ' ' max ' ' fac '
syms ← , " ' < ' ' ≤ ' ' = ' ' ≥ ' ' > ' ' ≠ ' ' ~ ' ' ^ ' ' v ' ' ^ ' ' v '
nams ← ' lth ' ' lte ' ' eql ' ' gte ' ' gth ' ' neq ' ' not ' ' and ' ' lor ' ' nan ' ' nor '
syms ← , " ' [] ' ' [ ' ' l ' ' p ' ' , ' ' , ' ' ϕ ' ' ϕ ' ' e ' ' e ' ' a '
nams ← ' sqd ' ' brk ' ' iot ' ' rho ' ' cat ' ' ctf ' ' rot ' ' trn ' ' rtf ' ' mem ' ' dis '
syms ← , " ' ≡ ' ' ≠ ' ' h ' ' h ' ' T ' ' L ' ' / ' ' / ' ' \ ' ' \ ' ' ? '
nams ← ' eqv ' ' nqv ' ' rgt ' ' lft ' ' enc ' ' dec ' ' red ' ' rdf ' ' scn ' ' scf ' ' rol '
syms ← , " ' ↑ ' ' ↓ ' ' ... ' ' ... ' ' . ' ' . ' ' * ' ' * ' ' o ' ' u ' ' n '
nams ← ' tke ' ' drp ' ' map ' ' com ' ' dot ' ' rnk ' ' pow ' ' jot ' ' unq ' ' int '
syms ← , " ' Δ ' ' Φ ' ' o . ' ' ε ' ' c ' ' ⊞ ' ' □FFT ' ' □IFFT ' ' %u '
nams ← ' gdu ' ' gdd ' ' oup ' ' fnd ' ' par ' ' mdv ' ' fft ' ' ift ' ' "

nl ← □ UCS 13 10 ◊ fvs ← , ö 0 ( / ~ ) 0 ≠ ( ≠ o p'' ¬ ) ◊ cln ← ' - ' □ R ' - ' ◊ cnm ← (syms 1 C) ⊃ (nams, C)
lits ← { 'A(0, eshp, constant(' , (cln Φ ω), ', eshp, ', ('f64' 's32' ⊃ ~ ω = |ω), ')) }
litv ← { 'std::vector< , ('DI' ⊃ ~ ^ / ω = |ω), '> { ' , (cln ⊃ {α, ', ', ω} / Φ'' ω), ' } .data() }
lita ← { 'A(1, dim4(' , (Φ ≠ ω), ' ) , array(' , (Φ ≠ ω), ', ', (litv ω), ' ) ) }
lit ← { ' ' = ⊃ 0 p ω : (cnm ω), α ◊ 1 = ≠ ω : lits ω ◊ lita ω }
var ← { α ≡ , 'α' : , 'l' ◊ α ≡ , 'ω' : , 'r' ◊ 1 ≥ ⊃ ω : α α lit , α ◊ 'env[', (Φ ⊃ ω), ']' [', (Φ ⊃ Φ ω), ']' }
dnv ← { (0 ≡ z) ⊃ ('A ' , α, '[' , (Φ z ← ⊃ v ω), ']' ; ') ('A* , α, '=NULL ; ') }
fnv ← { z ← 'A*env[', (Φ 1 + ⊃ s ω), ']' = { ' , (C / (C'env0'), { ' , p[', (Φ ω), ']' }'' 1 ⊃ s ω), '}' ; ', nl }
gcl ← { z r l n ← ((3 p C'fn'), Cα) { ⊃ α var / ω }'' ↓ (C ω), 1 ⊃ ω ◊ n, '(' , (C {α, ', ', ω} / z l r ~ C'fn'), ', env ) ; ', nl }

```

$rth \leftarrow \{\triangleright, / (Cnl), \sim 2 \downarrow \sim 2 \downarrow c \downarrow \sim 1 + (C'rth') \sim 3 \uparrow \cdot c \leftarrow \square SRC \square THIS\}$

```

A #include <time.h>
A #include <stdint.h>
A #include <inttypes.h>
A #include <limits.h>
A #include <float.h>
A #include <math.h>
A #include <memory>
A #include <algorithm>
A #include <string>
A #include <cstring>
A #include <vector>
A #include <unordered_map>
A #include <arrayfire.h>
A using namespace af;
A
A #if AF_API_VERSION < 35
A #error "Your ArrayFire version is too old."
A #endif
A #ifdef _WIN32
A #define EXPORT extern "C" __declspec(dllexport)
A #elif defined(__GNUC__)
A #define EXPORT extern "C" __attribute__((visibility ("default")))
A #else
A #define EXPORT extern "C"
A #endif
A #ifdef _MSC_VER
A #define RSTCT __restrict
A #else
A #define RSTCT restrict
A #endif
A #define S struct
A #define Z static
A #define R return
A #define RANK(lp) ((lp)->p->r)
A #define TYPE(lp) ((lp)->p->t)
A #define SHAPE(lp) ((lp)->p->s)
A #define ETYPE(lp) ((lp)->p->e)
A #define DATA(lp) ((V*)&SHAPE(lp)[RANK(lp)])
A #define CS(n,x) case n:x;break;
A #define DO(n,x) {I i=0,_i=(n);for(;i<_i;++i){x;}}
A #define DOB(n,x) {B i=0,_i=(n);for(;i<_i;++i){x;}}
A
A typedef enum{APLNC=0,APLU8,APLTI,APLSI,APLI,APLD,APLP,APLU,APLV,APLW,APLZ,
A APLR,APLF,APLQ}APLTYPE;
A typedef long long L;typedef int I;typedef int16_t S16;typedef int8_t S8;
A typedef double D;typedef unsigned char U8;typedef unsigned U;
A typedef dim_t B;typedef cdouble DZ;typedef void V;typedef std::string STR;
A
A S{U f=3;U n;U x=0;wchar_t*v=L"Co-dfns";const wchar_t*e;V*c;}dmx;
A S lp{S{L l;B c;U t:4;U r:4;U e:4;U _:13;U _1:16;U _2:16;B s[1];}*p;};
A S dwa{B z;S{B z;V*(*ga)(U,U,B*,S lp*);V(*p[16])();V(*er)(V*);} *ws;V*p[4];};
A S dwa*dwafns;Z V derr(U n){dmx.n=n;dwafns->ws->er(&dmx);}
A EXPORT I DyalogGetInterpreterFunctions(dwa*p){
A if(p)dwafns=p;else R 0;if(dwafns->z<sizeof(S dwa))R 16;R 0;}
A Z V err(U n, wchar_t*e){dmx.e=e;throw n;}Z V err(U n){dmx.e=L"";throw n;}
A S A{I r;dim4 s;array v;A(I r,dim4 s,array v):r(r),s(s),v(v){}
A A():r(0),s(dim4()),v(array()){}};
A int isinit=0;dim4 eshp=dim4(0,(B*)NULL);std::wstring msg;

```

```

R #define NM(n,nm,sm,sd,di,mf,df,ma,da) S n##_f:FN{di;mf;df;ma;da;\
R n##_f(STR s,I m,I d):FN(s,m,d){}} n##fn(nm,sm,sd);
R #define OM(n,nm,sm,sd,mf,df) S n##_o:MOP{mf;df;\
R n##_o(FN&l,A*p[]):MOP(nm,sm,sd,l,p){}};
R #define OD(n,nm,sm,sd,mf,df) S n##_o:DOP{mf;df;\
R n##_o(FN&l,FN&r,A*p[]):DOP(nm,sm,sd,l,r,p){}\
R n##_o(const A&l,FN&r,A*p[]):DOP(nm,sm,sd,l,r,p){}\
R n##_o(FN&l,const A&r,A*p[]):DOP(nm,sm,sd,l,r,p){}};
R #define MT
R #define DID inline array id(dim4)
R #define MFD inline V operator()(A&,const A&,A*[])
R #define MAD inline V operator()(A&,const A&,D ax,A*[])
R #define DFD inline V operator()(A&,const A&,const A&,A*[])
R #define DAD inline V operator()(A&,const A&,const A&,D ax,A*[])
R #define DI(n) inline array n::id(dim4 s)
R #define ID(n,x,t) DI(n##_f){R constant(x,s,t);}
R #define MF(n) inline V n::operator()(A&z,const A&r,A*p[])
R #define MA(n) inline V n::operator()(A&z,const A&r,D ax,A*p[])
R #define DF(n) inline V n::operator()(A&z,const A&l,const A&r,A*p[])
R #define DA(n) inline V n::operator()(A&z,const A&l,const A&r,D ax,A*p[])
R #define SF(n,x) inline V n::operator()(A&z,const A&l,const A&r,A*p[]){\
R if(l.r==r.r&&l.s==r.s){\
R z.r=l.r;z.s=l.s;const array&lv=l.v;const array&rv=r.v;x;R;}\
R if(!l.r){\
R z.r=r.r;z.s=r.s;const array&rv=r.v;array lv=tile(l.v,r.s);x;R;}\
R if(!r.r){\
R z.r=l.r;z.s=l.s;array rv=tile(r.v,l.s);const array&lv=l.v;x;R;}\
R if(l.r!=r.r)err(4);if(l.s!=r.s)err(5);err(99);}
R #define FP(n) NM(n,"",0,0,MT,MFD,DFD,MT,MT);MF(n##_f){n##fn(z,A(),r,p);}
R #define EF(n,m) EXPORT V n##_dwa(lp*z,lp*l,lp*r){try{\
R A cl,cr,za;if(!isinit){Initfn(za,cl,cr,NULL);isinit=1;}\
R cpda(cr,r);if(l!=NULL)cpda(cl,l);m##fn(za,cl,cr,env);cpad(z,za);}\
R catch(U n){derr(n);}\
R catch(exception e){msg=mkstr(e.what());dmx.e=msg.c_str();derr(500);}}\
R EXPORT V n##_cdf(A*z,A*l,A*r){try{m##fn(*z,*l,*r,env);}catch(U n){derr(n);}\
R catch(exception x){msg=mkstr(x.what());dmx.e=msg.c_str();derr(500);}}
R
R S FN{STR nm;I sm;I sd;FN(STR nm,I sm,I sd):nm(nm),sm(sm),sd(sd){}
R FN():nm(""),sm(0),sd(0){}
R virtual array id(dim4 s){err(16);R array();}
R virtual V operator()(A&z,const A&r,A*p[]){err(99);}
R virtual V operator()(A&z,const A&r,D ax,A*p[]){err(99);}
R virtual V operator()(A&z,const A&l,const A&r,A*p[]){err(99);}
R virtual V operator()(A&z,const A&l,const A&r,D ax,A*p[]){err(99);}};
R FN MTFN;
R S MOP:FN{FN&ll;A**pp;
R MOP(STR nm,I sm,I sd,FN&ll,A*pp[]):FN(nm,sm,sd),ll(ll),pp(pp){}};
R S DOP:FN{I fl;I fr;FN&ll;A aa;FN&rr;A ww;A**pp;
R DOP(STR nm,I sm,I sd,FN&l,FN&r,A*p[])
R :FN(nm,sm,sd),fl(1),fr(1),ll(l),aa(A()),rr(r),ww(A()),pp(p){}
R DOP(STR nm,I sm,I sd,A l,FN&r,A*p[])
R :FN(nm,sm,sd),fl(0),fr(1),ll(MTFN),aa(l),rr(r),ww(A()),pp(p){}
R DOP(STR nm,I sm,I sd,FN&l,A r,A*p[])
R :FN(nm,sm,sd),fl(1),fr(0),ll(l),aa(A()),rr(MTFN),ww(r),pp(p){}};
R

```



```

R std::wstring mkstr(const char*s){B c=std::strlen(s);std::wstring t(c,L' ');
R mbstowcs(&t[0],s,c);R t;}
R I scm(FN&f){R f.sm;}I scm(const A&a){R 1;}
R I scd(FN&f){R f.sd;}I scd(const A&a){R 1;}
R B cnt(dim4 s){B c=1;DO(4,c*=s[i]);R c;}
R B cnt(const A&a){B c=1;DO(a.r,c*=a.s[i]);R c;}
R B cnt(lp*d){B c=1;DO(RANK(d),c*=SHAPE(d)[i]);R c;}
R array scl(I x){R constant(x,dim4(1),s32);}
R A scl(array v){R A(0,dim4(1),v);}
R dtype mxt(dtype at,dtype bt){if(at==c64||bt==c64)R c64;
R if(at==f64||bt==f64)R f64;
R if(at==s32||bt==s32)R s32;if(at==s16||bt==s16)R s16;
R if(at==b8||bt==b8)R b8;err(16);R f64;}
R dtype mxt(const array&a,const array&b){R mxt(a.type(),b.type());}
R dtype mxt(dtype at,const A&b){R mxt(at,b.v.type());}
R Z array da16(B c,dim4 s,lp*d){std::vector<S16>b(c);
R S8*v=(S8*)DATA(d);DOB(c,b[i]=v[i]);R array(s,b.data());}
R Z array da8(B c,dim4 s,lp*d){std::vector<char>b(c);
R U8*v=(U8*)DATA(d);DOB(c,b[i]=1&(v[i/8]>>(7-(i%8))))
R R array(s,b.data());}
R V cpad(lp*d,A&a){I t;B c=cnt(a);
R switch(a.v.type()){CS(c64,t=APLZ);
R CS(s32,t=APLI);CS(s16,t=APLSI);CS(b8,t=APLTI);CS(f64,t=APLD);
R default:if(c)err(16);t=APLI;}
R B s[4];DO(a.r,s[a.r-(i+1)]=a.s[i]);dwafns->ws->ga(t,a.r,s,d);
R if(c)a.v.host(DATA(d));}
R V cpda(A&a,lp*d){if(15!=TYPE(d))err(16);if(4<RANK(d))err(16);
R dim4 s(1);DO(RANK(d),s[RANK(d)-(i+1)]=SHAPE(d)[i]);B c=cnt(d);
R switch(ETYPE(d)){
R CS(APLZ,a=A(RANK(d),s,c?array(s,(DZ*)DATA(d)):scl(0)))
R CS(APLI,a=A(RANK(d),s,c?array(s,(I*)DATA(d)):scl(0)))
R CS(APLD,a=A(RANK(d),s,c?array(s,(D*)DATA(d)):scl(0)))
R CS(APLSI,a=A(RANK(d),s,c?array(s,(S16*)DATA(d)):scl(0)))
R CS(APLTI,a=A(RANK(d),s,c?da16(c,s,d):scl(0)))
R CS(APLU8,a=A(RANK(d),s,c?da8(c,s,d):scl(0)))
R default:err(16);}}
R

```

```

A NM(add,"add",1,1,DID,MFD,DFD,MT ,MT )NM(sub,"sub",1,1,DID,MFD,DFD,MT ,MT )
A NM(mul,"mul",1,1,DID,MFD,DFD,MT ,MT )NM(div,"div",1,1,DID,MFD,DFD,MT ,MT )
A NM(max,"max",1,1,DID,MFD,DFD,MT ,MT )NM(min,"min",1,1,DID,MFD,DFD,MT ,MT )
A NM(exp,"exp",1,1,DID,MFD,DFD,MT ,MT )NM(log,"log",1,1,MT ,MFD,DFD,MT ,MT )
A NM(fac,"fac",1,1,DID,MFD,DFD,MT ,MT )NM(res,"res",1,1,DID,MFD,DFD,MT ,MT )
A NM(and,"and",1,1,DID,MT ,DFD,MT ,MT )NM(lor,"lor",1,1,DID,MT ,DFD,MT ,MT )
A NM(lth,"lth",1,1,DID,MT ,DFD,MT ,MT )NM(lte,"lte",1,1,DID,MT ,DFD,MT ,MT )
A NM(gth,"gth",1,1,DID,MT ,DFD,MT ,MT )NM(gte,"gte",1,1,DID,MT ,DFD,MT ,MT )
A NM(eql,"eql",1,1,DID,MT ,DFD,MT ,MT )NM(neq,"neq",1,1,DID,MT ,DFD,MT ,MT )
A NM(nan,"nan",1,1,MT ,MT ,DFD,MT ,MT )NM(nor,"nor",1,1,MT ,MT ,DFD,MT ,MT )
A NM(cir,"cir",1,1,MT ,MFD,DFD,MT ,MT )NM(not,"not",1,0,MT ,MFD,DFD,MT ,MT )
A NM(rot,"rot",0,0,DID,MFD,DFD,MT ,MT )NM(rtf,"rtf",0,0,DID,MFD,DFD,MT ,MT )
A NM(red,"red",0,0,DID,MT ,DFD,MT ,MT )NM(rdf,"rdf",0,0,DID,MT ,DFD,MT ,MT )
A NM(scn,"scn",0,0,DID,MT ,DFD,MT ,MT )NM(scf,"scf",0,0,DID,MT ,DFD,MT ,MT )
A NM(enc,"enc",0,0,DID,MT ,DFD,MT ,MT )NM(dec,"dec",0,0,MT ,MT ,DFD,MT ,MT )
A NM(sqd,"sqd",0,0,MT ,MFD,DFD,MT ,MT )NM(brk,"brk",0,0,MT ,MFD,DFD,MT ,MT )
A NM(iot,"iot",0,0,MT ,MFD,DFD,MT ,MT )NM(rho,"rho",0,0,MT ,MFD,DFD,MT ,MT )
A NM(cat,"cat",0,0,MT ,MFD,DFD,MT ,DAD)NM(ctf,"ctf",0,0,MT ,MFD,DFD,MT ,MT )
A NM(trn,"trn",0,0,MT ,MFD,DFD,MT ,MT )NM(rol,"rol",1,0,MT ,MFD,DFD,MT ,MT )
A NM(dis,"dis",0,0,MT ,MFD,DFD,MT ,MT )NM(par,"par",0,0,MT ,MFD,DFD,MT ,MT )
A NM(eqv,"eqv",0,0,MT ,MFD,DFD,MT ,MT )NM(nqv,"nqv",0,0,MT ,MFD,DFD,MT ,MT )
A NM(rgt,"rgt",0,0,MT ,MFD,DFD,MT ,MT )NM(lft,"lft",0,0,MT ,MFD,DFD,MT ,MT )
A NM(tke,"tke",0,0,MT ,MFD,DFD,MT ,MT )NM(drp,"drp",0,0,MT ,MFD,DFD,MT ,MT )
A NM(unq,"unq",0,0,MT ,MFD,DFD,MT ,MT )NM(int,"int",0,0,MT ,MT ,DFD,MT ,MT )
A NM(gdu,"gdu",0,0,MT ,MFD,DFD,MT ,MT )NM(gdd,"gdd",0,0,MT ,MFD,DFD,MT ,MT )
A NM(mem,"mem",0,0,MT ,MFD,DFD,MT ,MT )NM(fnd,"fnd",0,0,MT ,MT ,DFD,MT ,MT )
A NM(fft,"fft",1,0,MT ,MFD,MT ,MT ,MT )NM(ift,"ift",1,0,MT ,MFD,MT ,MT ,MT )
A NM(mdv,"mdv",1,0,MT ,MFD,DFD,MT ,MT )
A
A ID(add,0,s32)ID(sub,0,s32)ID(mul,1,s32)ID(div,1,s32)ID(res,0,s32)
A ID(min,DBL_MAX,f64)ID(max,-DBL_MAX,f64)ID(exp,1,s32)ID(fac,1,s32)
A ID(and,1,s32)ID(lor,0,s32)ID(lth,0,s32)ID(lte,1,s32)ID(eql,1,s32)
A ID(gth,0,s32)ID(gte,1,s32)ID(neq,0,s32)ID(enc,0,s32)ID(red,1,s32)
A ID(rdf,1,s32)ID(scn,1,s32)ID(scf,1,s32)ID(rot,0,s32)ID(rtf,0,s32)
A
A OD(brk,"brk",scm(l),scd(l),MFD,DFD)
A OM(com,"com",scm(l),scd(l),MFD,DFD)
A OD(dot,"dot",0,0,MT,DFD)
A OD(jot,"jot",(scm(l)&&scm(r)),(scd(l)&&scd(r)),MFD,DFD)
A OM(map,"map",1,1,MFD,DFD)
A OM(oup,"oup",0,0,MT,DFD)
A OD(pow,"pow",scm(l),scd(l),MFD,DFD)
A OM(red,"red",0,0,MFD,DFD)
A OM(rdf,"rdf",0,0,MFD,DFD)
A OD(rnk,"rnk",scm(l),0,MFD,DFD)
A OM(scn,"scn",1,1,MFD,MT)
A OM(scf,"scf",1,1,MFD,MT)
A

```

```

A MF(add_f){z=r;}
A SF(add_f,z.v=lv+rv)
A SF(and_f,if(lv.isbool()&&rv.isbool())z.v=lv&&rv;
A else if(allTrue<I>(lv>=0&&lv<=1&&rv>0&&rv<=1))z.v=lv&&rv;
A else{A a(z.r,z.s,lv);A b(z.r,z.s,rv);
A lorfn(a,a,b,p);z.v=lv*(rv/((!a.v)+a.v));})
A MF(brk_f){err(16);}
A DF(brk_f){if(l.r!=1)err(16);
A z.r=r.r;z.s=r.s;z.v=l.v(r.v.as(s32));}
A MF(cat_f){z.r=1;z.s[0]=cnt(r);z.v=flat(r.v);}
A DA(cat_f){A nl=l,nr=r;I fx=(I)ceil(ax);
A if(fx<0||fx>r.r&&fx>l.r)err(4);
A if(ax!=fx){if(r.r>3||l.r>3)err(10);
A if(nl.r){nl.r++;DO(3-fx,nl.s[3-i]=nl.s[3-(i+1)]);nl.s[fx]=1;}
A if(nr.r){nr.r++;DO(3-fx,nr.s[3-i]=nr.s[3-(i+1)]);nr.s[fx]=1;}
A if(nl.r)nl.v=moddims(nl.v,nl.s);if(nr.r)nr.v=moddims(nr.v,nr.s);
A catfn(z,nl,nr,fx,p);R;}
A if(fx>=r.r&&fx>=l.r)err(4);
A if(l.r&&r.r&&std::abs((I)l.r-(I)r.r)>1)err(4);
A z.r=(l.r>=r.r)*l.r+(r.r>l.r)*r.r+(!r.r&&!l.r);
A dim4 ls=l.s;dim4 rs=r.s;
A if(!l.r){ls=rs;ls[fx]=1;}if(!r.r){rs=ls;rs[fx]=1;}
A if(r.r&&l.r>r.r){DO(3-fx,rs[3-i]=rs[3-(i+1)]);rs[fx]=1;}
A if(l.r&&r.r>l.r){DO(3-fx,ls[3-i]=ls[3-(i+1)]);ls[fx]=1;}
A DO(4,if(i!=fx&&rs[i]!=ls[i])err(5));
A DO(4,z.s[i]=(l.r>=r.r||i==fx)*ls[i]+(r.r>l.r||i==fx)*rs[i]);
A if(!cnt(l)){z.v=r.v;R;}if(!cnt(r)){z.v=l.v;R;}
A dtype mt=mxt(r.v,l.v);
A array lv=(l.r?moddims(l.v,ls):tile(l.v,ls)).as(mt);
A array rv=(r.r?moddims(r.v,rs):tile(r.v,rs)).as(mt);
A z.v=join(fx,lv,rv);}
A DF(cat_f){if(l.r||r.r){catfn(z,l,r,0,p);R;}
A A a,b;catfn(a,l,p);catfn(b,r,p);catfn(z,a,b,0,p);}
A MF(cir_f){z.r=r.r;z.s=r.s;z.v=Pi*r.v.as(f64);}
A SF(cir_f,array fv=rv.as(f64);
A if(!l.r){I x=l.v.as(s32).scalar<I>();if(abs(x)>10)err(16);
A switch(x){CS(0,z.v=sqrt(1-fv*fv))CS(1,z.v=sin(fv))CS(2,z.v=cos(fv))
A CS(3,z.v=tan(fv))CS(4,z.v=sqrt(1+fv*fv))CS(5,z.v=sinh(fv))
A CS(6,z.v=cosh(fv))CS(7,z.v=tanh(fv))CS(8,z.v=sqrt(fv*fv-1))CS(9,z.v=fv)
A CS(10,z.v=abs(fv))CS(-1,z.v=asin(fv))CS(-2,z.v=acos(fv))
A CS(-3,z.v=atan(fv))CS(-4,z.v=(fv+1)*sqrt((fv-1)/(fv+1)))
A CS(-5,z.v=asinh(fv))CS(-6,z.v=acosh(fv))CS(-7,z.v=atanh(fv))
A CS(-8,z.v=-sqrt(fv*fv-1))CS(-9,z.v=fv)CS(-10,z.v=fv)}R;}
A if(anyTrue<I>(abs(lv)>10))err(16);B c=cnt(z);std::vector<I> a(c);
A std::vector<D> b(c);lv.as(s32).host(a.data());fv.host(b.data());
A std::vector<D> zv(c);
A DOB(c,switch(a[i]){CS(0,zv[i]=sqrt(1-b[i]*b[i]))CS(1,zv[i]=sin(b[i]))
A CS(2,zv[i]=cos(b[i]))CS(3,zv[i]=tan(b[i]))CS(4,zv[i]=sqrt(1+b[i]*b[i]))
A CS(5,zv[i]=sinh(b[i]))CS(6,zv[i]=cosh(b[i]))CS(7,zv[i]=tanh(b[i]))
A CS(8,zv[i]=sqrt(b[i]*b[i]-1))CS(9,zv[i]=b[i])CS(10,zv[i]=abs(b[i]))
A CS(-1,zv[i]=asin(b[i]))CS(-2,zv[i]=acos(b[i]))CS(-3,zv[i]=atan(b[i]))
A CS(-4,zv[i]=(b[i]==-1)?0:(b[i]+1)*sqrt((b[i]-1)/(b[i]+1)))
A CS(-5,zv[i]=asinh(b[i]))CS(-6,zv[i]=acosh(b[i]))CS(-7,zv[i]=atanh(b[i]))
A CS(-8,zv[i]=-sqrt(b[i]*b[i]-1))CS(-9,zv[i]=b[i])CS(-10,zv[i]=b[i])})
A z.v=array(z.s,zv.data());}
A MF(ctf_f){dim4 sp=z.s;sp[1]=r.r?r.s[r.r-1]:1;sp[0]=sp[1]?cnt(r)/sp[1]:1;
A sp[2]=sp[3]=1;z.r=2;z.s=sp;z.v=!cnt(z)?scl(0):array(r.v,z.s);}
A DF(ctf_f){I x=l.r>r.r?l.r:r.r;if(l.r||r.r){catfn(z,l,r,x-1,p);R;}
A A a,b;catfn(a,l,p);catfn(b,r,p);catfn(z,a,b,0,p);}

```

```

A DF(dec_f){I ra=r.r?r.r-1:0;I la=l.r?l.r-1:0;z.r=ra+la;z.s=dim4(1);
A if(l.s[0]!=1&&l.s[0]!=r.s[ra]&&r.s[ra]!=1)err(5);
A DO(ra,z.s[i]=r.s[i])DO(la,z.s[i+ra]=l.s[i+1])
A if(!cnt(z)){z.v=scl(0);R;}
A if(!cnt(r)||!cnt(l)){z.v=constant(0,z.s,s32);R;}
A B lc=l.s[0];array x=l.v;if(lc==1){lc=r.s[ra];x=tile(x,(I)lc);}
A x=flip(scan(x,0,AF_BINARY_MUL,false),0);
A x=array(x,lc,x.elements()/lc).as(f64);
A array y=array(r.v,cnt(r)/r.s[ra],r.s[ra]).as(f64);
A z.v=array(matmul(r.s[ra]==1?tile(y,1,(I)l.s[0]):y,x),z.s);}
A MF(dis_f){z.r=0;z.s=eshp;z.v=r.v(0);}
A DF(dis_f){if(l.v.isfloating())err(1);if(l.r>1)err(4);
A B lc=cnt(l);if(!lc){z=r;R;}if(lc!=1||r.r!=1)err(4);
A if(allTrue<char>(cnt(r)<=l.v(0)))err(3);
A z.r=0;z.s=eshp;array i=l.v(0);z.v=r.v(i);}
A MF(div_f){z.r=r.r;z.s=r.s;z.v=1.0/r.v.as(f64);}
A SF(div_f,z.v=lv.as(f64)/rv.as(f64))
A MF(drp_f){if(r.r)err(16);z=r;}
A DF(drp_f){I lv[4];seq it[4];seq ix[4];B c=cnt(l);
A if(l.r>1||(c>r.r&&r.r))err(4);if(!c){z=r;R;}
A U rk=r.r?r.r:(U)l.s[0];z.r=rk;z.s=r.s;l.v.as(s32).host(lv);
A DO((I)c,{U j=rk-(i+1);I a=std:abs(lv[i]);
A if(a>=r.s[j]){z.s[j]=0;ix[j]=seq(0);it[j]=seq(0);}
A else if(lv[i]<0){
A z.s[j]=r.s[j]-a;ix[j]=seq((D)z.s[j]);it[j]=ix[j];}
A else{z.s[j]=r.s[j]-a;ix[j]=seq(a,(D)r.s[j]-1);it[j]=ix[j]-(D)a;}})
A if(!cnt(z)){z.v=scl(0);R;}z.v=array(z.s,r.v.type());z.v=0;
A z.v(it[0],it[1],it[2],it[3])=r.v(ix[0],ix[1],ix[2],ix[3]);}
A DF(enc_f){I rk=r.r+l.r;if(rk>4)err(16);dim4 sp=r.s;DO(l.r,sp[i+r.r]=l.s[i])
A if(!cnt(sp)){z.r=rk;z.s=sp;z.v=scl(0);R;}dim4 lt=sp,rt=sp;I k=l.r?l.r-1:0;
A DO(r.r,rt[i]=1)DO(l.r,lt[i+r.r]=1)array rv=tile(r.v,rt);z.r=rk;z.s=sp;
A array sv=flip(scan(flip(l.v,k),k,AF_BINARY_MUL),k);
A array lv=tile(array(sv,rt),lt);af::index x[4];x[k]=0;
A array dv=sv;dv(x[0],x[1],x[2],x[3])=1;I s[]={0,0,0,0};s[k]=-1;
A dv=shift(dv,s[0],s[1],s[2],s[3]);dv=tile(array(dv,rt),lt);
A z.v=(lv!=0)*rem(rv,lv)+(lv==0)*rv;z.v=(dv!=0)*(z.v/dv).as(s32);}
A SF(eql_f,z.v=lv==rv)
A MF(eqv_f){z.r=0;z.s=eshp;z.v=scl(r.r!=0);}
A DF(eqv_f){z.r=0;z.s=eshp;
A if(l.r==r.r&&l.s==r.s){z.v=allTrue(l.v==r.v);R;}z.v=scl(0);}
A MF(exp_f){z.r=r.r;z.s=r.s;z.v=exp(r.v.as(f64));}
A SF(exp_f,z.v=pow(lv.as(f64),rv.as(f64)))
A MF(fac_f){z.r=r.r;z.s=r.s;z.v=factorial(r.v.as(f64));}
A SF(fac_f,array lvf=lv.as(f64);array rvf=rv.as(f64);
A z.v=exp(log(tgamma(lvf))+log(tgamma(rvf))-log(tgamma(lvf+rvf))))
A MF(fft_f){z.r=r.r;z.s=r.s;z.v=dft(r.v.type()==c64?r.v:r.v.as(c64),1,r.s);}
A MF(ift_f){z.r=r.r;z.s=r.s;z.v=idft(r.v.type()==c64?r.v:r.v.as(c64),1,r.s);}
A DF(fnd_f){A t(r.r,r.s,array(r.s,b8));if(!cnt(t)){t.v=scl(0);z=t;R;}
A t.v=0;if(l.r>r.r){z=t;R;}DO(4,if(l.s[i]>r.s[i]){z=t;R;}
A if(!cnt(l)){t.v=1;z=t;R;}dim4 sp;DO(4,sp[i]=1+(t.s[i]-l.s[i]))
A seq x[4];DO(4,x[i]=seq((D)sp[i]))t.v(x[0],x[1],x[2],x[3])=1;
A DO((I)l.s[0],I m=i;
A DO((I)l.s[1],I k=i;
A DO((I)l.s[2],I j=i;
A DO((I)l.s[3],t.v(x[0],x[1],x[2],x[3])=t.v(x[0],x[1],x[2],x[3])
A &(tile(l.v(m,k,j,i),sp)
A ==r.v(x[0]+(D)m,x[1]+(D)k,x[2]+(D)j,x[3]+(D)i))))))
A z=t;}
A

```

```

A MF(gdd_f){if(r.r<1)err(4);z.r=1;z.s=dim4(r.s[r.r-1]);
A   if(!cnt(r)){z.v=r.v;R;}I c=1;DO(r.r-1,c*=(I)r.s[i]);
A   array mt,a=array(r.v,c,r.s[r.r-1]);z.v=iota(z.s,dim4(1),s32);
A   DO(c,sort(mt,z.v,flat(a(c-(i+1),z.v)),z.v,0,false))}
A DF(gdd_f){err(16);}
A MF(gdu_f){if(r.r<1)err(4);z.r=1;z.s=dim4(r.s[r.r-1]);
A   if(!cnt(r)){z.v=r.v;R;}I c=1;DO(r.r-1,c*=(I)r.s[i]);
A   array mt,a=array(r.v,c,r.s[r.r-1]);z.v=iota(z.s,dim4(1),s32);
A   DO(c,sort(mt,z.v,flat(a(c-(i+1),z.v)),z.v,0,true))}
A DF(gdu_f){err(16);}
A SF(gte_f,z.v=lv>=rv)
A SF(gth_f,z.v=lv>rv)
A DF(int_f){if(r.r>1||l.r>1)err(4);
A   if(!cnt(r)||!cnt(l)){z.v=scl(0);z.s=dim4(0);z.r=1;R;}
A   dtype mt=mxt(l.v,r.v);z.v=setIntersect(l.v.as(mt),r.v.as(mt));
A   z.r=1;z.s=dim4(z.v.elements());}
A MF(iot_f){if(r.r>1)err(4);B c=cnt(r);if(c>4)err(10);
A   if(c>1)err(16);
A   z.r=1;z.s=dim4(r.v.as(s32).scalar<I>());
A   z.v=z.s[0]?iota(z.s,dim4(1),s32):scl(0);}
A DF(iot_f){z.r=r.r;z.s=r.s;B c=cnt(r);if(!c){z.v=scl(0);R;}
A   B lc=cnt(l)+1;if(lc==1){z.v=scl(0);R;};if(l.r>1)err(16);
A   array rf=flat(r.v).T();dtype mt=mxt(l.v,rf);
A   z.v=join(0,tile(l.v,1,(U)c).as(mt),rf.as(mt))==tile(rf,(U)lc,1);
A   z.v=min((z.v*iota(dim4(lc),dim4(1,c),s32)+(!z.v)*lc).as(s32),0);
A   z.v=array(z.v,z.s);}
A MF(lft_f){z=r;}
A DF(lft_f){z=l;}
A MF(log_f){z.r=r.r;z.s=r.s;z.v=log(r.v.as(f64));}
A SF(log_f,z.v=log(rv.as(f64))/log(lv.as(f64)))
A SF(lor_f,if(rv.isbool()&&lv.isbool())z.v=lv||rv;
A   else if(lv.isbool()&&rv.isinteger())z.v=lv+(!lv)*abs(rv).as(rv.type());
A   else if(rv.isbool()&&lv.isinteger())z.v=rv+(!rv)*abs(lv).as(lv.type());
A   else if(lv.isinteger()&&rv.isinteger()){B c=cnt(z);
A     std::vector<I> a(c);abs(lv).as(s32).host(a.data());
A     std::vector<I> b(c);abs(rv).as(s32).host(b.data());
A     DOB(c,while(b[i]){I t=b[i];b[i]=a[i]%b[i];a[i]=t;})
A     z.v=array(z.s,a.data());}
A   else{B c=cnt(z);
A     std::vector<D> a(c);abs(lv).as(f64).host(a.data());
A     std::vector<D> b(c);abs(rv).as(f64).host(b.data());
A     DOB(c,while(b[i]>1e-12){D t=b[i];b[i]=fmod(a[i],b[i]);a[i]=t;})
A     z.v=array(z.s,a.data());}}}
A SF(lte_f,z.v=lv<=rv)
A SF(lth_f,z.v=lv<rv)
A MF(max_f){z.r=r.r;z.s=r.s;z.v=ceil(r.v).as(r.v.type());}
A SF(max_f,z.v=max(lv,rv))
A MF(mem_f){z.r=1;z.s=dim4(cnt(r));z.v=flat(r.v);}
A DF(mem_f){z.r=l.r;z.s=l.s;I lc=(I)cnt(z);if(!lc){z.v=scl(0);R;}
A   if(!cnt(r)){z.v=array(z.s,b8);z.v=0;R;}
A   array y=setUnique(flat(r.v));I rc=(I)y.elements();
A   array x=array(flat(l.v),lc,1);y=array(y,1,rc);
A   z.v=array(anyTrue(tile(x,1,rc)==tile(y,lc,1),1),z.s);}
A MF(mdv_f){if(r.r>2)err(4);if(r.r==2&&r.s[1]<r.s[0])err(5);if(!cnt(r))err(5);
A   if(r.s[0]==r.s[1]){z.r=r.r;z.s=r.s;z.v=inverse(r.v);R;}
A   if(r.r==1){z.v=matmulNT(inverse(matmulTN(r.v,r.v)),r.v);z.r=r.r;z.s=r.s;R;}
A   z.v=matmulTN(inverse(matmulNT(r.v,r.v)),r.v);z.r=r.r;z.s=r.s;
A   B k=z.s[0];z.s[0]=z.s[1];z.s[1]=k;z.v=transpose(z.v);}
A

```

```

A DF(mdv_f){if(r.r>2)err(4);if(l.r>2)err(4);if(r.r==2&&r.s[1]<r.s[0])err(5);
A  if(!cnt(r)||!cnt(l))err(5);if(r.r&&l.r&&l.s[l.r-1]!=r.s[r.r-1])err(5);
A  array rv=r.v,lv=l.v;if(r.r==1)rv=transpose(rv);if(l.r==1)lv=transpose(lv);
A  z.v=transpose(matmul(inverse(matmulNT(rv,rv)),matmulNT(rv,lv)));
A  z.r=(l.r-(l.r>0))+(r.r-(r.r>0));
A  if(l.r>1)z.s[0]=l.s[0];if(r.r>1)z.s[l.r>1]=r.s[0];}
A MF(min_f){z.r=r.r;z.s=r.s;z.v=floor(r.v).as(r.v.type());}
A SF(min_f,z.v=min(lv,rv))
A MF(mul_f){z.r=r.r;z.s=r.s;z.v=(r.v>0)-(r.v<0);}
A SF(mul_f,z.v=lv*rv)
A SF(nan_f,z.v=!(lv&&rv))
A SF(neq_f,z.v=lv!=rv)
A SF(nor_f,z.v=!(lv||rv))
A MF(not_f){z.r=r.r;z.s=r.s;z.v=!r.v;}
A DF(not_f){err(16);}
A MF(nqv_f){z.v=scl(r.r?(I)r.s[r.r-1]:1);z.r=0;z.s=dim4(1);}
A DF(nqv_f){z.r=0;z.s=eshp;I t=l.r==r.r&&l.s==r.s;
A  if(t)t=allTrue<I>(l.v==r.v);z.v=scl(!t);}
A MF(par_f){err(16);}
A DF(par_f){err(16);}
A DF(red_f){if(l.r>1)err(4);z.r=r.r?r.r-1;z.s=r.s;
A  if(l.r!=0&&l.s[0]!=1&&r.r!=0&&r.s[0]!=1&&l.s[0]!=r.s[0])err(5);
A  array x=l.v;if(cnt(l)==1)x=tile(x,(I)r.s[0]);
A  array y=r.v;if(r.s[0]==1)y=tile(y,(I)cnt(l));
A  z.s[0]=sum<B>(abs(x));if(!cnt(z)){z.v=scl(0);R;}
A  array w=where(x).as(s32);
A  if(z.s[0]==w.elements()){z.v=y(w,span);R;}
A  array i=shift(accum(abs(x(w))),1),d=shift(w,1);i(0)=0;d(0)=0;
A  array v=array(z.s[0],s32),u=array(z.s[0],s32);v=0;u=0;
A  array s=(!sign(x(w))).as(s32);array t=shift(s,1);t(0)=0;
A  v(i)=w-d;u(i)=s-t;z.v=y(accum(v),span);
A  z.v*=tile(accum(u),1,(I)z.s[1],(I)z.s[2],(I)z.s[3]);}
A MF(res_f){z.r=r.r;z.s=r.s;z.v=abs(r.v).as(r.v.type());}
A SF(res_f,z.v=rv-lv*floor(rv.as(f64)/(lv+(0==lv))))
A DF(rdf_f){if(l.r>1)err(4);I ra=r.r?r.r-1:0;z.r=ra+1;z.s=r.s;
A  if(l.r!=0&&l.s[0]!=1&&r.r!=0&&r.s[ra]!=1&&l.s[0]!=r.s[ra])err(5);
A  array x=l.v;array y=r.v;if(cnt(l)==1)x=tile(x,(I)r.s[ra]);
A  if(r.s[ra]==1){dim4 s(1);s[ra]=cnt(l);y=tile(y,s);}
A  z.s[ra]=sum<B>(abs(x));if(!cnt(z)){z.v=scl(0);R;}
A  array w=where(x).as(s32);af::index ix[4];if(z.s[ra]==w.elements()){
A  ix[ra]=w;z.v=y(ix[0],ix[1],ix[2],ix[3]);R;}
A  array i=shift(accum(abs(x(w))),1),d=shift(w,1);i(0)=0;d(0)=0;
A  array v=array(z.s[ra],s32),u=array(z.s[ra],s32);v=0;u=0;
A  array s=(!sign(x(w))).as(s32);array t=shift(s,1);t(0)=0;
A  v(i)=w-d;u(i)=s-t;ix[ra]=accum(v);z.v=y(ix[0],ix[1],ix[2],ix[3]);
A  dim4 s1(1),s2(z.s);s1[ra]=z.s[ra];s2[ra]=1;u=array(accum(u),s1);
A  z.v*=tile(u,(I)s2[0],(I)s2[1],(I)s2[2],(I)s2[3]);}
A MF(rgt_f){z=r;}
A DF(rgt_f){z=r;}
A MF(rho_f){I sp[4]={1,1,1,1};DO(r.r,sp[r.r-(i+1)]=(I)r.s[i]);
A  z.s=dim4(r.r);z.r=1;if(!cnt(z)){z.v=scl(0);R;}z.v=array(z.s,sp);}
A DF(rho_f){B cr=cnt(r);B cl=cnt(l);B s[4];if(l.r>1)err(11);if(cl>4)err(16);
A  l.v.as(s64).host(s);z.r=(I)cl;DO(4,z.s[i]=i>=z.r?1:s[z.r-(i+1)])B cz=cnt(z);
A  if(!cz){z.v=scl(0);R;}z.v=array(cz==cr?r.v:flat(r.v)(iota(cz)%cr),z.s);}
A MF(rol_f){z.r=r.r;z.s=r.s;if(!cnt(r)){z.v=r.v;R;}
A  array rnd=randu(r.v.dims(),f64);z.v=(0==r.v)*rnd+trunc(r.v*rnd);}
A

```

```

A DF(rol_f){if(cnt(r)!=1||cnt(l)!=1)err(5);
A D lv=l.v.as(f64).scalar<D>();D rv=r.v.as(f64).scalar<D>();
A if(lv>rv||lv!=floor(lv)||rv!=floor(rv)||lv<0||rv<0)err(11);
A I s=(I)lv;I t=(I)rv;z.r=1;z.s=dim4(s);if(!s){z.v=scl(0);R;}
A std::vector<I> g(t);std::vector<I> d(t);
A ((1+range(t))*randu(t)).as(s32).host(g.data());
A DO(t,I j=g[i];if(i!=j)d[i]=d[j];d[j]=i)z.v=array(z.s,d.data());}
A MF(rot_f){z.r=r.r;z.s=r.s;z.v=flip(r.v,0);}
A DF(rot_f){I lc=(I)cnt(l);if(lc==1){z.r=r.r;z.s=r.s;
A z.v=shift(r.v,-l.v.as(s32).scalar<I>());R;}
A if(l.r!=r.r-1)err(5);DO(l.r,if(l.s[i]!=r.s[i+1])err(5))
A std::vector<I> x(lc);l.v.as(s32).host(x.data());
A z.v=array(r.v,r.s[0],lc);z.r=r.r;z.s=r.s;
A DO(lc,z.v(span,i)=shift(z.v(span,i),-x[i]))z.v=array(z.v,z.s);}
A MF(rtf_f){z.r=r.r;z.s=r.s;z.v=r.r?flip(r.v,r.r-1):r.v;}
A DF(rtf_f){I lc=(I)cnt(l);I ra=r.r?r.r-1:0;I ix[]={0,0,0,0};
A if(lc==1){z.r=r.r;z.s=r.s;ix[ra]=-l.v.as(s32).scalar<I>();
A z.v=shift(r.v,ix[0],ix[1],ix[2],ix[3]);R;}
A if(l.r!=r.r-1)err(5);DO(l.r,if(l.s[i]!=r.s[i])err(5))
A std::vector<I> x(lc);l.v.as(s32).host(x.data());
A z.v=array(r.v,lc,r.s[ra]);z.r=r.r;z.s=r.s;
A DO(lc,z.v(i,span)=shift(z.v(i,span),0,-x[i]))
A z.v=array(z.v,z.s);}
A DF(scn_f){if(r.s[0]!=1&& r.s[0]!=sum<I>(l.v>0))err(5);
A if(l.r>1)err(5);array ca=max(1,abs(l.v)).as(s32);I c=sum<I>(ca);
A if(!cnt(l))c=0;A t(r.r?r.r-1:r.s,scl(0));t.s[0]=c;
A if(!cnt(t)){z=t;R;}t.v=array(t.s,r.v.type());t.v=0;
A array pw=0<l.v;array pa=pw*l.v;I pc=sum<I>(pa);if(!pc){z=t;R;}
A pw=where(pw);pa=scan(pa(pw),0,AF_BINARY_ADD,false);
A array si(pc,s32);si=0;si(pa)=1;si=accum(si)-1;
A array ti(pc,s32);ti=1;ti(pa)=scan(ca,0,AF_BINARY_ADD,false)(pw);
A ti=scanByKey(si,ti);t.v(ti,span)=r.v(si,span);z=t;}
A DF(scf_f){I ra=r.r?r.r-1:0;af::index sx[4];af::index tx[4];
A if(r.s[ra]!=1&& r.s[ra]!=sum<I>(l.v>0))err(5);
A if(l.r>1)err(5);array ca=max(1,abs(l.v)).as(s32);I c=sum<I>(ca);
A if(!cnt(l))c=0;A t(ra+1,r.s,scl(0));t.s[ra]=c;
A if(!cnt(t)){z=t;R;}t.v=array(t.s,r.v.type());t.v=0;
A array pw=0<l.v;array pa=pw*l.v;I pc=sum<I>(pa);if(!pc){z=t;R;}
A pw=where(pw);pa=scan(pa(pw),0,AF_BINARY_ADD,false);
A array si(pc,s32);si=0;si(pa)=1;si=accum(si)-1;sx[ra]=si;
A array ti(pc,s32);ti=1;ti(pa)=scan(ca,0,AF_BINARY_ADD,false)(pw);
A ti=scanByKey(si,ti);tx[ra]=ti;
A t.v(tx[0],tx[1],tx[2],tx[3])=r.v(sx[0],sx[1],sx[2],sx[3]);z=t;}
A MF(sqd_f){z=r;}
A DF(sqd_f){if(l.r>1)err(4);B s=!l.r?1:l.s[l.r-1];
A if(s>r.r)err(5);if(!cnt(l)){z=r;R;}
A I sv[4];af::index x[4];l.v.as(s32).host(sv);
A DO((I)s,if(sv[i]<0||sv[i]>=r.s[i])err(3));
A DO((I)s,x[r.r-(i+1)]=sv[i]);z.r=r.r-(U)s;z.s=dim4(z.r,r.s.get());
A z.v=r.v(x[0],x[1],x[2],x[3]);}
A MF(sub_f){z.r=r.r;z.s=r.s;z.v=-r.v;}
A SF(sub_f,z.v=lv-rv)
A MF(tke_f){z=r;}
A

```

```

R DF(tke_f){I lv[4];seq it[4];seq ix[4];B c=cnt(l);
R if(l.r>1||(c>r.r&&r.r))err(4);if(!c){z=r;R;}
R U rk=r.r?r.r:(U)l.s[0];z.r=rk;z.s=r.s;l.v.as(s32).host(lv);
R DO((I)c,{U j=rk-(i+1);I a=std::abs(lv[i]);z.s[j]=a;
R if(a>r.s[j])ix[j]=seq((D)r.s[j]);
R else if(lv[i]<0)ix[j]=seq((D)r.s[j]-a,(D)r.s[j]-1);
R else ix[j]=seq(a);
R it[j]=ix[j]+(lv[i]<0)*(a-(D)r.s[j]);})
R if(!cnt(z)){z.v=scl(0);R;}z.v=array(z.s,r.v.type());z.v=0;
R z.v(it[0],it[1],it[2],it[3])=r.v(ix[0],ix[1],ix[2],ix[3]);}
R MF(trn_f){z.r=r.r;DO(r.r,z.s[i]=r.s[r.r-(i+1)])
R switch(r.r){CS(0,z.v=r.v)CS(1,z.v=r.v)CS(2,z.v=r.v.T())
R CS(3,z.v=reorder(r.v,2,1,0))CS(4,z.v=reorder(r.v,3,2,1,0))}}
R DF(trn_f){I lv[4];if(l.r>1||cnt(l)!=r.r)err(5);
R l.v.as(s32).host(lv);DO(r.r,if(lv[i]<0||lv[i]>=r.r)err(4))
R U8 f[]={0,0,0,0};DO(r.r,f[lv[i]]=1)
R U8 t=1;DO(r.r,if(t&&!f[i])t=0;else if(!t&&f[i])err(5))
R DO(r.r,if(!f[i])err(16))
R z.r=r.r;DO(r.r,z.s[r.r-(lv[i]+1)]=r.s[r.r-(i+1)])
R I s[4];DO(r.r,s[r.r-(lv[i]+1)]=r.r-(i+1))
R switch(r.r){CS(0,z.v=r.v)CS(1,z.v=r.v)
R CS(2,z.v=reorder(r.v,s[0],s[1]))
R CS(3,z.v=reorder(r.v,s[0],s[1],s[2]))
R CS(4,z.v=reorder(r.v,s[0],s[1],s[2],s[3]))}}
R MF(unq_f){if(r.r>1)err(4);z.r=1;if(!cnt(r)){z.s=r.s;z.v=r.v;R;}
R array a,b;sort(a,b,r.v);z.v=a!=shift(a,1);z.v(0)=1;
R z.v=where(z.v);sort(b,z.v,b(z.v),a(z.v));
R z.s=dim4(z.v.elements());}
R DF(unq_f){if(r.r>1||l.r>1)err(4);z.r=1;dtype mt=mxt(l.v,r.v);
R if(!cnt(l)){z.s=r.s;z.v=r.v;R;}if(!cnt(r)){z.s=l.s;z.v=l.v;R;}
R array x=setUnique(l.v);B c=x.elements();
R z.v=!anyTrue(tile(r.v,1,(U)c)==tile(array(x,1,c),(U)r.s[0],1),1);
R z.v=join(0,l.v.as(mt),r.v(where(z.v)).as(mt));
R z.s=dim4(z.v.elements());}
R

```



```

A #define brkop(zz,ll,rr,pp) brk_o zz(ll,rr,pp)
A #define comop(zz,rr,pp) com_o zz(rr,pp)
A #define dotop(zz,ll,rr,pp) dot_o zz(ll,rr,pp)
A #define mapop(zz,rr,pp) map_o zz(rr,pp)
A #define redop(zz,rr,pp) red_o zz(rr,pp)
A #define jotop(zz,ll,rr,pp) jot_o zz(ll,rr,pp)
A #define oupop(zz,rr,pp) oup_o zz(rr,pp)
A #define powop(zz,ll,rr,pp) pow_o zz(ll,rr,pp)
A #define rdfop(zz,rr,pp) rdf_o zz(rr,pp)
A #define rnkop(zz,ll,rr,pp) rnk_o zz(ll,rr,pp)
A #define scnop(zz,rr,pp) scn_o zz(rr,pp)
A #define scfop(zz,rr,pp) scf_o zz(rr,pp)
A
A MF(brk_o){ll(z,r,(r.r?r.r-1:0)-ww.v.as(f64).scalar<D>(),p);}
A DF(brk_o){D ax=l.r;if(r.r>l.r)ax=r.r;if(ax)ax--;
A ll(z,l,r,ax-ww.v.as(f64).scalar<D>(),p);}
A MF(com_o){ll(z,r,r,p);}DF(com_o){ll(z,r,l,p);}
A DF(dot_o){I ra=r.r?r.r-1:0;if(r.r&l.r&l.s[0]!=r.s[ra])err(5);
A I la=l.r?l.r-1:0;A t(la+ra,r.s,r.v(0));if(t.r>4)err(10);
A t.s[ra]=1;DO(la,t.s[i+ra]=l.s[i+1])if(!cnt(t)){t.v=scl(0);z=t;R;}
A if(!l.s[0]||!r.s[ra]){t.v=ll.id(t.s);z=t;R;}
A I c=(I)(l.r?l.s[0]:r.s[ra]);
A I rc=(I)(cnt(r)/r.s[ra]);I lc=(I)(cnt(l)/l.s[0]);
A array x=array(l.v,(I)l.s[0],lc);array y=array(r.v,rc,(I)r.s[ra]);
A if(1==l.s[0]){x=tile(x,c,1);}if(1==r.s[ra]){y=tile(y,1,c);}
A if("add"==ll.nm&&"mul"==rr.nm){
A t.v=array(matmul(y.as(f64),x.as(f64)),t.s);z=t;R;}
A x=tile(array(x,c,1,lc),1,rc,1);y=tile(y.T(),1,1,lc);
A A X(3,dim4(c,rc,lc),x.as(f64));A Y(3,dim4(c,rc,lc),y.as(f64));
A mapop(mfn,rr,p);redop(rfn,ll,p);mfn(X,X,Y,p);rfn(X,X,p);
A t.v=array(X.v,t.s);z=t;}
A MF(jot_o){if(!fl){rr(z,aa,r,p);R;}if(!fr){ll(z,r,ww,p);R;}
A rr(z,r,p);ll(z,z,p);}
A DF(jot_o){if(!fl||!fr){err(2);}rr(z,r,p);ll(z,l,z,p);}
A MF(map_o){if(scm(ll)){ll(z,r,p);R;}
A z.r=r.r;z.s=r.s;I c=(I)cnt(z);if(!c){z.v=scl(0);R;}
A A zs;A rs=scl(r.v(0));ll(zs,rs,p);if(c==1){z.v=zs.v;R;}
A array v=array(z.s,zs.v.type());v(0)=zs.v(0);
A DO(c-1,rs.v=r.v(i+1);ll(zs,rs,p);v(i+1)=zs.v(0))z.v=v;}
A DF(map_o){if(scd(ll)){ll(z,l,r,p);R;}
A if((l.r==r.r&l.s==r.s)||!l.r){z.r=r.r;z.s=r.s;}
A else if(!r.r){z.r=l.r;z.s=l.s;}else if(l.r!=r.r)err(4);
A else if(l.s!=r.s)err(5);else err(99);I c=(I)cnt(z);
A if(!c){z.v=scl(0);R;}A zs;A rs=scl(r.v(0));A ls=scl(l.v(0));
A ll(zs,ls,rs,p);if(c==1){z.v=zs.v;R;}
A array v=array(z.s,zs.v.type());v(0)=zs.v(0);
A if(!r.r){rs.v=r.v;
A DO(c-1,ls.v=l.v(i+1);ll(zs,ls,rs,p);v(i+1)=zs.v(0);)
A z.v=v;R;}
A if(!l.r){ls.v=l.v;
A DO(c-1,rs.v=r.v(i+1);ll(zs,ls,rs,p);v(i+1)=zs.v(0);)
A z.v=v;R;}
A DO(c-1,ls.v=l.v(i+1);rs.v=r.v(i+1);ll(zs,ls,rs,p);
A v(i+1)=zs.v(0))z.v=v;}
A DF(oup_o){A t(l.r+r.r,r.s,r.v(0));if(t.r>4)err(10);
A DO(l.r,t.s[i+r.r]=l.s[i])if(!cnt(t)){t.v=scl(0);z=t;R;}
A array x(flat(l.v),1,cnt(l));array y(flat(r.v),cnt(r),1);
A dim4 ts(cnt(r),cnt(l));x=tile(x,(I)ts[0],1);y=tile(y,1,(I)ts[1]);
A mapop(mfn,ll,p);A xa(2,ts,x);A ya(2,ts,y);mfn(xa,xa,ya,p);
A t.v=array(xa.v,t.s);z=t;}

```

```

R MF(pow_o){if(fr){A t;A v=r;
R do{A u;ll(u,v,p);rr(t,u,v,p);if(t.r)err(5);v=u;}
R while(!t.v.as(s32).scalar<I>());z=v;R;}
R if(wv.r)err(4);I c=wv.v.as(s32).scalar<I>();z=r;DO(c,ll(z,z,p))}
R DF(pow_o){if(fr){A t;A v=r;
R do{A u;ll(u,l,v,p);rr(t,u,v,p);if(t.r)err(5);v=u;}
R while(!t.v.as(s32).scalar<I>());z=v;R;}
R if(wv.r)err(4);I c=wv.v.as(s32).scalar<I>();
R A t=r;DO(c,ll(t,l,t,p))z=t;}
R MF(rdf_o){A t(r.r?r.r-1:0,dim4(1),r.v(0));DO(t.r,t.s[i]=r.s[i])
R I rc=(I)r.s[t.r];I zc=(I)cnt(t);mapop(mfn,ll,p);
R if(!zc){t.v=scl(0);z=t;R;}if(!rc){t.v=ll.id(t.s);z=t;R;}
R if(1==rc){t.v=array(r.v,t.s);z=t;R;}
R if("add"==ll.nm){if(r.v.isbool())t.v=count(r.v,t.r).as(s32);
R else t.v=sum(r.v.as(f64),t.r);z=t;R;}
R if("mul"==ll.nm){t.v=product(r.v.as(f64),t.r);z=t;R;}
R if("min"==ll.nm){t.v=min(r.v,t.r);z=t;R;}
R if("max"==ll.nm){t.v=max(r.v,t.r);z=t;R;}
R if("and"==ll.nm){t.v=allTrue(r.v,t.r);z=t;R;}
R if("lor"==ll.nm){t.v=anyTrue(r.v,t.r);z=t;R;}
R af::index x[4];x[t.r]=rc-1;t.v=r.v(x[0],x[1],x[2],x[3]);
R DO(rc-1,x[t.r]=rc-(i+2);
R mfn(t,A(t.r,t.s,r.v(x[0],x[1],x[2],x[3])),t,p));z=t;}
R DF(rdf_o){if(l.r!=0&&(l.r!=1||l.s[0]!=1))err(5);if(!r.r)err(4);
R I lv=l.v.as(s32).scalar<I>();I ra=r.r-1;
R if((r.s[ra]+1)<lv)err(5);I rc=(I)((1+r.s[ra])-abs(lv));
R mapop(mfn,ll,p);A t(r.r,r.s,scl(0));t.s[ra]=rc;if(!cnt(t)){z=t;R;}
R if(!lv){t.v=ll.id(t.s);z=t;R;}seq rng(rc);af::index x[4];
R if(lv>=0){x[ra]=rng+((D)lv-1);t.v=r.v(x[0],x[1],x[2],x[3]);
R DO(lv-1,x[ra]=rng+((D)lv-(i+2));
R mfn(t,A(t.r,t.s,r.v(x[0],x[1],x[2],x[3])),t,p))
R }else{x[ra]=rng;t.v=r.v(x[0],x[1],x[2],x[3]);
R DO(abs(lv)-1,x[ra]=rng+(D)(i+1);
R mfn(t,A(t.r,t.s,r.v(x[0],x[1],x[2],x[3])),t,p))}
R z=t;}
R MF(red_o){A t(r.r?r.r-1:0,dim4(1),z.v);DO(t.r,t.s[i]=r.s[i+1])
R I rc=(I)r.s[0];I zc=(I)cnt(t);if(!zc){t.v=scl(0);z=t;R;}
R if(!rc){t.v=ll.id(t.s);z=t;R;}
R if(1==rc){t.v=array(r.v,t.s);z=t;R;}
R if("add"==ll.nm){if(r.v.isbool())t.v=count(r.v,0).as(s32);
R else t.v=sum(r.v.as(f64),0);z=t;R;}
R if("mul"==ll.nm){t.v=product(r.v.as(f64),0);z=t;R;}
R if("min"==ll.nm){t.v=min(r.v,0);z=t;R;}
R if("max"==ll.nm){t.v=max(r.v,0);z=t;R;}
R if("and"==ll.nm){t.v=allTrue(r.v,0);z=t;R;}
R if("lor"==ll.nm){t.v=anyTrue(r.v,0);z=t;R;}
R t.v=r.v(rc-1,span);mapop(mfn,ll,p);
R DO(rc-1,mfn(t,A(t.r,t.s,r.v(rc-(i+2),span)),t,p))z=t;}
R DF(red_o){if(l.r!=0&&(l.r!=1||l.s[0]!=1))err(5);if(!r.r)err(4);
R I lv=l.v.as(s32).scalar<I>();if((r.s[0]+1)<lv)err(5);
R I rc=(I)((1+r.s[0])-abs(lv));mapop(mfn,ll,p);
R A t(r.r,r.s,scl(0));t.s[0]=rc;if(!cnt(t)){z=t;R;}
R if(!lv){t.v=ll.id(t.s);z=t;R;}seq rng(rc);
R if(lv>=0){t.v=r.v(rng+((D)lv-1),span);
R DO(lv-1,mfn(t,A(t.r,t.s,r.v(rng+((D)lv-(i+2)),span)),t,p))
R }else{t.v=r.v(rng,span);
R DO(abs(lv)-1,mfn(t,A(t.r,t.s,r.v(rng+(D)(i+1),span)),t,p))}
R z=t;}
R

```

```

A MF(rnk_o){if(cnt(wv)!=1)err(4);I cr=ww.v.as(s32).scalar<I>();
A if(scm(ll)||cr>=r.r){ll(z,r,p);R;}
A if(cr<=-r.r||!cr){mapop(f,ll,p);f(z,r,p);R;}
A if(cr<0)cr=r.r+cr;if(cr>3)err(10);I dr=r.r-cr;
A dim4 sp(1);DO(dr,sp[cr]*=r.s[i+cr])DO(cr,sp[i]=r.s[i])
A std::vector<A> tv(sp[cr]);A b(cr+1,sp,array(r.v,sp));
A DO((I)sp[cr],sqdfn(tv[i],scl(scl(i)),b,p);ll(tv[i],tv[i],p))
A I mr=0;dim4 ms(1);dtype mt=b8;if(mr>3)err(10);
A DO((I)sp[cr],if(mr<tv[i].r)mr=tv[i].r;mt=mxt(mt,tv[i]);I si=i;
A DO(4,if(ms[3-i]<tv[si].s[3-i]){ms=tv[si].s;break;}))
A I mc=(I)cnt(ms);array v(mc*sp[cr],mt);v=0;
A DO((I)sp[cr],seq ix((D)cnt(tv[i]));v(ix+(D)(i*mc))=flat(tv[i].v))
A z.r=mr+dr;z.s=ms;z.s[mr]=sp[cr];z.v=array(v,z.s);}
A DF(rnk_o){I cl,cr,dl,dr;dim4 sl(1),sr(1);array wwv=ww.v.as(s32);
A if(cnt(ww)==1)cl=cr=wwv.scalar<I>();
A else if(cnt(ww)==2){cl=wwv.scalar<I>();cr=wwv(1).scalar<I>();}
A else err(4);
A if(cl>l.r)cl=l.r;if(cr>r.r)cr=r.r;if(cl<-l.r)cl=0;if(cr<-r.r)cr=0;
A if(cl<0)cl=l.r+cl;if(cr<0)cr=r.r+cr;if(cr>3||cl>3)err(10);
A dl=l.r-cl;dr=r.r-cr;if(dl!=dr&&dl&&dr)err(4);
A if(dl==dr)DO(dr,if(l.s[i+cl]!=r.s[i+cr])err(5))
A DO(dl,sl[cl]*=l.s[i+cl])DO(cl,sl[i]=l.s[i])
A DO(dr,sr[cr]*=r.s[i+cr])DO(cr,sr[i]=r.s[i])
A B sz=dl>dr?sl[cl]:sr[cr];std::vector<A> tv(sz);
A A a(cl+1,sl,array(l.v,sl));A b(cr+1,sr,array(r.v,sr));
A I mr=0;dim4 ms(1);dtype mt=b8;
A DO((I)sz,A ta;A tb;A ai=scl(scl(i%sl[cl]));A bi=scl(scl(i%sr[cr]));
A sqdfn(ta,ai,a,p);sqdfn(tb,bi,b,p);ll(tv[i],ta,tb,p);
A if(mr<tv[i].r)mr=tv[i].r;mt=mxt(mt,tv[i]);A t=tv[i];
A DO(4,if(ms[i]<t.s[i])ms[i]=t.s[i]))
A B mc=cnt(ms);array v(mc*sz,mt);v=0;
A DOB(sz,seq ix((D)cnt(tv[i]));v(ix+(D)(i*mc))=flat(tv[i].v))
A z.r=mr+(dr>dl?dr:dl);z.s=ms;z.s[mr]=sz;z.v=array(v,z.s);}
A MF(scn_o){z.r=r.r;z.s=r.s;I rc=(I)r.s[0];
A if(1==rc){z.v=r.v;R;}if(!cnt(z)){z.v=scl(0);R;}
A if("add"==ll.nm){z.v=scan(r.v.as(f64),0,AF_BINARY_ADD);R;}
A if("mul"==ll.nm){z.v=scan(r.v.as(f64),0,AF_BINARY_MUL);R;}
A if("min"==ll.nm){z.v=scan(r.v.as(f64),0,AF_BINARY_MIN);R;}
A if("max"==ll.nm){z.v=scan(r.v.as(f64),0,AF_BINARY_MAX);R;}
A mapop(mfn,ll,p);z.v=array(z.s,f64);A t(z.r?z.r-1:0,z.s,r.v(0));
A DO(t.r,t.s[i]=t.s[i+1]);t.s[t.r]=1;I tc=(I)cnt(t);
A DO(rc,t.v=r.v(i,span).as(f64);I c=i;
A DO(c,mfn(t,A(t.r,t.s,r.v(c-(i+1),span)),t,p))
A z.v(i,span)=t.v)}
A MF(scf_o){z.r=r.r;z.s=r.s;I ra=r.r?r.r-1:0;I rc=(I)r.s[ra];
A if(1==rc){z.v=r.v;R;}if(!cnt(z)){z.v=scl(0);R;}
A if("add"==ll.nm){z.v=scan(r.v.as(f64),ra,AF_BINARY_ADD);R;}
A if("mul"==ll.nm){z.v=scan(r.v.as(f64),ra,AF_BINARY_MUL);R;}
A if("min"==ll.nm){z.v=scan(r.v.as(f64),ra,AF_BINARY_MIN);R;}
A if("max"==ll.nm){z.v=scan(r.v.as(f64),ra,AF_BINARY_MAX);R;}
A z.v=array(z.s,f64);A t(z.r?z.r-1:0,z.s,r.v(0));t.s[ra]=1;
A I tc=(I)cnt(t);af::index x[4];mapop(mfn,ll,p);
A DO(rc,x[ra]=i;t.v=r.v(x[0],x[1],x[2],x[3]).as(f64);I c=i;
A DO(c,x[ra]=c-(i+1);
A mfn(t,A(t.r,t.s,r.v(x[0],x[1],x[2],x[3])),t,p))
A x[ra]=i;z.v(x[0],x[1],x[2],x[3])=t.v)}
A

```

```

R EXPORT A*mkarray(lp*d){A*z=new A;cpda(*z,d);R z;}
R EXPORT V freea(A*a){delete a;}
R EXPORT V exarray(lp*d,A*a){cpad(d,*a);}
R EXPORT V afsync(){sync();}
R EXPORT Window *w_new(char *k){R new Window(k);}
R EXPORT I w_close(Window*w){R w->close();}
R EXPORT V w_del(Window*w){delete w;}
R EXPORT V w_img(lp*d,Window*w){A a;cpda(a,d);
R   w->image(a.v.as(a.r==2?f32:u8));}
R EXPORT V w_plot(lp*d,Window*w){A a;cpda(a,d);w->plot(a.v.as(f32));}
R EXPORT V w_hist(lp*d,D l,D h,Window*w){A a;cpda(a,d);
R   w->hist(a.v.as(u32),l,h);}
R EXPORT V loading(lp*z,char*p,I c){array a=loadImage(p,c);
R   A b(a.numdims(),a.dims(),a.as(s16));cpad(z,b);}
R EXPORT V saveimg(lp*im,char*p){A a;cpda(a,im);
R   saveImageNative(p,a.v.as(a.v.type()==s32?u16:u8));}

```

:EndNamespace