```
root@62226b904dda:/work# cafeobj lecture6/submit.cafe

-- loading standard prelude

            -- CafeOBJ system Version 1.6.2(PigNose0.99) --
                   built: 2024 Dec 26 Thu 0:09:00 GMT
                         prelude file: std.bin
                                  ***
                      2024 Dec 27 Fri 0:05:17 GMT
                            Type ? for help
                                  ***
                  -- Containing PigNose Extensions --
                                  ---
                             built on SBCL
                             2.0.1.debian
processing input : /work/lecture6/submit.cafe

-- defining module! NAT-TRIPLE-IMLM
-- reading in file  : nat

processing input : /usr/local/share/cafeobj-1.6/lib/nat.cafe

-- defining module! NAT
-- reading in file  : nznat

processing input : /usr/local/share/cafeobj-1.6/lib/nznat.cafe

-- defining module! NZNAT
-- reading in file  : bool

processing input : /usr/local/share/cafeobj-1.6/lib/bool.cafe

processing input : /usr/local/share/cafeobj-1.6/lib/base_bool.cafe

processing input : /usr/local/share/cafeobj-1.6/lib/truth.cafe

-- defining module! TRUTH
-- reading in file  : truth

-- done reading in file: truth

-- defining module* BASE-BOOL
-- reading in file  : eql

processing input : /usr/local/share/cafeobj-1.6/lib/eql.cafe

-- defining module! EQL
-- done reading in file: eql

processing input : /usr/local/share/cafeobj-1.6/lib/sys_bool.cafe

-- defining module! BOOL
-- done reading in file: bool

-- done reading in file: nznat

-- done reading in file: nat

-- defining module! NAT-TRIPLE-OMLM
-- opening module NAT-TRIPLE-OMLM
-- reduce in %NAT-TRIPLE-OMLM : (1st(((1 + 1) , (2 + 2) , (3 + 3)))):Nat
[1]: (1st(((1 + 1) , (2 + 2) , (3 + 3)))):Nat
---> (1 + 1):NzNat
[2]: (1 + 1):NzNat
---> (2):NzNat
(2):NzNat
(0.0000 sec for parse, 0.0000 sec for 2 rewrites + 2 matches)
-- defining module! NAT-IF
module! NAT-IF
{
  imports {
    protecting (NAT)
  }
  signature {
    op if _ then { _ } else { _ } : Bool Nat Nat -> Nat { strat: (1 0 2 3) prec: 0 }
  }
  axioms {
    var N1 : Nat
    var N2 : Nat
    eq (if true then { N1 } else { N2 })
        = N1 .
    eq (if false then { N1 } else { N2 })
        = N2 .
  }
}

-- defining module! GLIST
-- defining module! INF-LIST
-- defining module! NAT-INF-LIST
-- opening module NAT-INF-LIST
-- reduce in %NAT-INF-LIST : (mkNILFrom(0)):InfList
(0 | mkNILFrom(1)):InfList
(0.0000 sec for parse, 0.0000 sec for 1 rewrites + 1 matches)
-- reduce in %NAT-INF-LIST : (take(mkNILFrom(0),10)):List
(0 | (1 | (2 | (3 | (4 | (5 | (6 | (7 | (8 | (9 | nil)))))))))):List
(0.0000 sec for parse, 0.0000 sec for 41 rewrites + 51 matches)
-- reduce in %NAT-INF-LIST : (drop(mkNILFrom(0),10)):InfList
(10 | mkNILFrom((1 + 10))):InfList
(0.0000 sec for parse, 0.0000 sec for 41 rewrites + 51 matches)
-- reduce in %NAT-INF-LIST : (take(drop(mkNILFrom(0),997),10)):List
(997 | (998 | (999 | (1000 | (1001 | (1002 | (1003 | (1004 | (1005 | (1006 | nil)))))))))):List
(0.0000 sec for parse, 0.0070 sec for 4030 rewrites + 5037 matches)
-- reduce in %NAT-INF-LIST : (take((take(mkNILFrom(0),10) @ drop(mkNILFrom(0),10)),20)):List
(0 | (1 | (2 | (3 | (4 | (5 | (6 | (7 | (8 | (9 | (10 | (11 | (12 | (13 | (14 | (15 | (16 | (17 | (18 |
(19 | nil)))))))))))))))))))):List
(0.0000 sec for parse, 0.0010 sec for 154 rewrites + 204 matches)
-- reduce in %NAT-INF-LIST : (take(mkNILFrom(0),20)):List
(0 | (1 | (2 | (3 | (4 | (5 | (6 | (7 | (8 | (9 | (10 | (11 | (12 | (13 | (14 | (15 | (16 | (17 | (18 |
(19 | nil)))))))))))))))))))):List
(0.0000 sec for parse, 0.0000 sec for 81 rewrites + 101 matches)
-- reduce in %NAT-INF-LIST : (zip(mkNILFrom(0),mkNILFrom(0))):InfList
(0 | (0 | zip(mkNILFrom(1),mkNILFrom(1)))):InfList
(0.0000 sec for parse, 0.0000 sec for 3 rewrites + 3 matches)
-- reduce in %NAT-INF-LIST : (take(drop(zip(mkNILFrom(0),mkNILFrom(0)),997),10)):List
(498 | (499 | (499 | (500 | (500 | (501 | (501 | (502 | (502 | (503 | nil)))))))))):List
(0.0000 sec for parse, 0.0110 sec for 4532 rewrites + 5539 matches)
-- defining module! ERATOSTHENES-SIEVE
-- opening module ERATOSTHENES-SIEVE
-- reduce in %ERATOSTHENES-SIEVE : (primes):InfList
(2 | sieve(check(2,mkNILFrom((1 + 2))))):InfList
(0.0000 sec for parse, 0.0000 sec for 3 rewrites + 3 matches)
-- reduce in %ERATOSTHENES-SIEVE : (take(primes,10)):List
(2 | (3 | (5 | (7 | (11 | (13 | (17 | (19 | (23 | (29 | nil)))))))))):List
(0.0000 sec for parse, 0.0010 sec for 332 rewrites + 483 matches)
-- reduce in %ERATOSTHENES-SIEVE : (take(primes,20)):List
(2 | (3 | (5 | (7 | (11 | (13 | (17 | (19 | (23 | (29 | (31 | (37 | (41 | (43 | (47 | (53 | (59 | (61 |
(67 | (71 | nil)))))))))))))))))))):List
(0.0000 sec for parse, 0.0060 sec for 1055 rewrites + 1590 matches)
-- reduce in %ERATOSTHENES-SIEVE : (take(primes,50)):List
(2 | (3 | (5 | (7 | (11 | (13 | (17 | (19 | (23 | (29 | (31 | (37 | (41 | (43 | (47 | (53 | (59 | (61 |
(67 | (71 | (73 | (79 | (83 | (89 | (97 | (101 | (103 | (107 | (109 | (113 | (127 | (131 | (137 | (139
| (149 | (151 | (157 | (163 | (167 | (173 | (179 | (181 | (191 | (193 | (197 | (199 | (211 | (223 |
(227 | (229 | nil)))))))))))))))))))))))))))))))))))))))))))))))))):List
(0.0000 sec for parse, 0.0540 sec for 5332 rewrites + 8345 matches)
-- reduce in %ERATOSTHENES-SIEVE : (take(primes,100)):List
(2 | (3 | (5 | (7 | (11 | (13 | (17 | (19 | (23 | (29 | (31 | (37 | (41 | (43 | (47 | (53 | (59 | (61 |
(67 | (71 | (73 | (79 | (83 | (89 | (97 | (101 | (103 | (107 | (109 | (113 | (127 | (131 | (137 | (139
| (149 | (151 | (157 | (163 | (167 | (173 | (179 | (181 | (191 | (193 | (197 | (199 | (211 | (223 |
(227 | (229 | (233 | (239 | (241 | (251 | (257 | (263 | (269 | (271 | (277 | (281 | (283 | (293 | (307
| (311 | (313 | (317 | (331 | (337 | (347 | (349 | (353 | (359 | (367 | (373 | (379 | (383 | (389 |
(397 | (401 | (409 | (419 | (421 | (431 | (433 | (439 | (443 | (449 | (457 | (461 | (463 | (467 | (479
| (487 | (491 | (499 | (503 | (509 | (521 | (523 | (541 |
nil))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))
:List
(0.0000 sec for parse, 0.3100 sec for 18953 rewrites + 30314 matches)
-- defining module! HAMMING
-- opening module HAMMING
-- reduce in %HAMMING : (ham):InfList
(1 | merge(merge(2*(ham),3*(ham)),5*(ham))):InfList
(0.0000 sec for parse, 0.0000 sec for 1 rewrites + 1 matches)
-- reduce in %HAMMING : (take(ham,10)):List
(1 | (2 | (3 | (4 | (5 | (6 | (8 | (9 | (10 | (12 | nil)))))))))):List
(0.0000 sec for parse, 0.0030 sec for 404 rewrites + 438 matches)
-- reduce in %HAMMING : (take(ham,20)):List
(1 | (2 | (3 | (4 | (5 | (6 | (8 | (9 | (10 | (12 | (15 | (16 | (18 | (20 | (24 | (25 | (27 | (30 | (32
| (36 | nil)))))))))))))))))))):List
(0.0000 sec for parse, 0.0220 sec for 1370 rewrites + 1484 matches)
-- reduce in %HAMMING : (take(ham,50)):List
(1 | (2 | (3 | (4 | (5 | (6 | (8 | (9 | (10 | (12 | (15 | (16 | (18 | (20 | (24 | (25 | (27 | (30 | (32
| (36 | (40 | (45 | (48 | (50 | (54 | (60 | (64 | (72 | (75 | (80 | (81 | (90 | (96 | (100 | (108 |
(120 | (125 | (128 | (135 | (144 | (150 | (160 | (162 | (180 | (192 | (200 | (216 | (225 | (240 | (243
| nil)))))))))))))))))))))))))))))))))))))))))))))))))):List
(0.0000 sec for parse, 0.3360 sec for 10826 rewrites + 11788 matches)
-- reduce in %HAMMING : (take(ham,100)):List
(1 | (2 | (3 | (4 | (5 | (6 | (8 | (9 | (10 | (12 | (15 | (16 | (18 | (20 | (24 | (25 | (27 | (30 | (32
| (36 | (40 | (45 | (48 | (50 | (54 | (60 | (64 | (72 | (75 | (80 | (81 | (90 | (96 | (100 | (108 |
(120 | (125 | (128 | (135 | (144 | (150 | (160 | (162 | (180 | (192 | (200 | (216 | (225 | (240 | (243
| (250 | (256 | (270 | (288 | (300 | (320 | (324 | (360 | (375 | (384 | (400 | (405 | (432 | (450 |
(480 | (486 | (500 | (512 | (540 | (576 | (600 | (625 | (640 | (648 | (675 | (720 | (729 | (750 | (768
| (800 | (810 | (864 | (900 | (960 | (972 | (1000 | (1024 | (1080 | (1125 | (1152 | (1200 | (1215 |
(1250 | (1280 | (1296 | (1350 | (1440 | (1458 | (1500 | (1536 |
nil))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))
:List
(0.0000 sec for parse, 5.8130 sec for 72940 rewrites + 79462 matches)
-- defining module! LOC
-- defining module! STATE
-- defining module! TID
-- defining module! FMUTEX
-- defining module! COMP
-- defining module! SCHED
-- opening module SCHED
-- reduce in %SCHED : (take(sched(123),10)):List
(t2 | (t4 | (t1 | (t1 | (t1 | (t3 | (t4 | (t1 | (t1 | (t1 | nil)))))))))):List
(0.0000 sec for parse, 0.0000 sec for 78 rewrites + 97 matches)
-- reduce in %SCHED : (take(sched(1234),10)):List
(t1 | (t4 | (t1 | (t1 | (t1 | (t4 | (t1 | (t1 | (t4 | (t1 | nil)))))))))):List
(0.0000 sec for parse, 0.0000 sec for 78 rewrites + 98 matches)
-- reduce in %SCHED : (take(sched(12345),10)):List
(t2 | (t3 | (t4 | (t1 | (t2 | (t4 | (t1 | (t1 | (t1 | (t4 | nil)))))))))):List
(0.0000 sec for parse, 0.0000 sec for 84 rewrites + 107 matches)
-- defining module! SIM
-- opening module SIM
-- reduce in %SIM : (take(sim((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs),123),10)):FComp
((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs) | ((locked: false , pc1: rs , pc2: ms , pc3:
rs , pc4: rs) | ((locked: false , pc1: rs , pc2: ms , pc3: rs , pc4: ms) | ((locked: false , pc1: ms ,
pc2: ms , pc3: rs , pc4: ms) | ((locked: true , pc1: cs , pc2: ms , pc3: rs , pc4: ms) | ((locked:
false , pc1: rs , pc2: ms , pc3: rs , pc4: ms) | ((locked: false , pc1: rs , pc2: ms , pc3: ms , pc4:
ms) | ((locked: true , pc1: rs , pc2: ms , pc3: ms , pc4: cs) | ((locked: true , pc1: rs , pc2: ms ,
pc3: ms , pc4: cs) | ((locked: true , pc1: rs , pc2: ms , pc3: ms , pc4: cs) | nil)))))))))):FComp
(0.0000 sec for parse, 0.0010 sec for 100 rewrites + 166 matches)
-- reduce in %SIM : (take(sim((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs),1234),10)):FComp
((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs) | ((locked: false , pc1: ms , pc2: rs , pc3:
rs , pc4: rs) | ((locked: false , pc1: ms , pc2: rs , pc3: rs , pc4: ms) | ((locked: true , pc1: cs ,
pc2: rs , pc3: rs , pc4: ms) | ((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: ms) | ((locked:
false , pc1: ms , pc2: rs , pc3: rs , pc4: ms) | ((locked: true , pc1: ms , pc2: rs , pc3: rs , pc4:
cs) | ((locked: true , pc1: cs , pc2: rs , pc3: rs , pc4: cs) | ((locked: false , pc1: rs , pc2: rs ,
pc3: rs , pc4: cs) | ((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs) | nil)))))))))):FComp
(0.0000 sec for parse, 0.0000 sec for 100 rewrites + 175 matches)
-- reduce in %SIM : (take(sim((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs),12345),10)):FComp
((locked: false , pc1: rs , pc2: rs , pc3: rs , pc4: rs) | ((locked: false , pc1: rs , pc2: ms , pc3:
rs , pc4: rs) | ((locked: false , pc1: rs , pc2: ms , pc3: ms , pc4: rs) | ((locked: false , pc1: rs ,
pc2: ms , pc3: ms , pc4: ms) | ((locked: false , pc1: ms , pc2: ms , pc3: ms , pc4: ms) | ((locked:
true , pc1: ms , pc2: cs , pc3: ms , pc4: ms) | ((locked: true , pc1: ms , pc2: cs , pc3: ms , pc4: cs)
| ((locked: true , pc1: cs , pc2: cs , pc3: ms , pc4: cs) | ((locked: false , pc1: rs , pc2: cs , pc3:
ms , pc4: cs) | ((locked: false , pc1: ms , pc2: cs , pc3: ms , pc4: cs) | nil)))))))))):FComp
(0.0000 sec for parse, 0.0010 sec for 106 rewrites + 198 matches)
CafeOBJ> ^C
** Caught an exception: Interactive interrupt at #xFFFFA724D040.
[Returning to Toplevel]

CafeOBJ> [Leaving CafeOBJ]
```