\*version\* ad2d string "0.0.2"

;;; core types

%lambda

%package

%exception

%vector

%closure

;;; fixum

1+ 47bad func #<:function :lambda [req:1, form:b9a29]>

1- b9ccd func #<:function :lambda [req:1, form:baf49]>

ceiling c236d func #<:function :lambda [req:2, form:c50e9]>

floor befed func #<:function :lambda [req:2, form:c1b29]>

mod c54ad func #<:function :lambda [req:2, form:c7869]>

rem bed2d func #<:function :lambda [req:2, form:c9f69]>

round ca5cd func #<:function :lambda [req:2, form:cd1c9]>

truncate bb82d func #<:function :lambda [req:2, form:be549]>

;;; lists/maps

append f390d func #<:function :lambda [req:1, form:152b89]>

assq 1927ed func #<:function :lambda [req:2, form:193d49]>

count 15ea4d func #<:function :lambda [req:2, form:160ce9]>

dropl ee6ed func #<:function :lambda [req:2, form:156789]>

dropr ee48d func #<:function :lambda [req:2, form:15aca9]>

findl 7dfcd func #<:function :lambda [req:2, form:143489]>

findr 14404d func #<:function :lambda [req:2, form:143ec9]>

foldl 15e54d func #<:function :lambda [req:3, form:1772c9]>

foldr eba6d func #<:function :lambda [req:3, form:179b49]>

last 15b56d func #<:function :lambda [req:1, form:15e289]>

mapc 17bf8d func #<:function :lambda [req:2, form:17e4c9]>

mapcar f862d func #<:function :lambda [req:2, form:182369]>

mapl 182d2d func #<:function :lambda [req:2, form:186289]>

maplist 13322d func #<:function :lambda [req:2, form:18b3e9]>

positionl 1394cd func #<:function :lambda [req:2, form:14a189]>

positionr 14b06d func #<:function :lambda [req:2, form:150309]>

reverse 14366d func #<:function :lambda [req:1, form:1541a9]>

rassq 19404d func #<:function :lambda [req:2, form:195509]>

;;;

apply a8e4d func #<:function :lambda [req:2, form:10ed09]>

format a0aad func #<:function :lambda [req:3, form:eb7e9]>

load-file 24dc8d func #<:function :lambda [req:1, form:24db69]>

type-of 19ead func #<:function :lambda [req:1, form:89169]>

;;; compiler

compile 2e28d func #<:function :lambda [req:1, form:39be9]>

compile-funcall 306ad func #<:function :lambda [req:3, form:106b29]>

;;; types

def-type 6f84d func #<:function :lambda [req:2, form:72b09]>

typep 8df2d func #<:function :lambda [req:2, form:a0189]>

;;; packages

intern 1a17ed func #<:function :lambda [req:2, form:1a1669]>

make-package 19ae0d func #<:function :lambda [req:2, form:19d209]>

;;; predicates

atom cc0d func #<:function :lambda [req:1, form:ca89]>

bit-vector-p 897ad func #<:function :lambda [req:1, form:29be29]>

boundp fd72d func #<:function :mu [req:1, form:boundp]>

charp 1748d func #<:function :lambda [req:1, form:17309]>

consp c4cd func #<:function :lambda [req:1, form:17b69]>

constantp f1ad func #<:function :lambda [req:1, form:f089]>

dotted-pair-p 13d7ed func #<:function :lambda [req:1, form:13d6c9]>

exceptionp b5b4d func #<:function :lambda [req:1, form:b5a29]>

fboundp 25f26d func #<:function :lambda [req:1, form:25f0e9]>

fixnump f30d func #<:function :lambda [req:1, form:18349]>

floatp f3ed func #<:function :lambda [req:1, form:18b29]>

functionp 739ed func #<:function :lambda [req:1, form:113aa9]>

keywordp ceed func #<:function :lambda [req:1, form:2600e9]>

listp 19d2d func #<:function :lambda [req:1, form:19ba9]>

minusp 1298d func #<:function :lambda [req:1, form:14809]>

namespacep 1a82d func #<:function :lambda [req:1, form:1a709]>

not c3ed func #<:function :lambda [req:1, form:c269]>

null bd6d func #<:function :lambda [req:1, form:1ae49]>

numberp 1042d func #<:function :lambda [req:1, form:102a9]>

packagep 1a82d func #<:function :lambda [req:1, form:1a709]>

plusp 14c8d func #<:function :lambda [req:1, form:16aa9]>

streamp 1b7ad func #<:function :lambda [req:1, form:1b629]>

stringp 3d14d func #<:function :lambda [req:1, form:3cfc9]>

structp 1c00d func #<:function :lambda [req:1, form:1be89]>

symbolp d20d func #<:function :lambda [req:1, form:2611c9]>

uninternedp 117d0d func #<:function :lambda [req:1, form:25f829]>

vectorp 7598d func #<:function :lambda [req:1, form:299ea9]>

zerop 107cd func #<:function :lambda [req:1, form:124a9]>

;;; streams

close 24440d func #<:function :lambda [req:1, form:245789]>

get-output-stream-string 3d8ad func #<:function :lambda [req:1, form:3d789]>

make-string-stream 23b0ad func #<:function :lambda [req:2, form:23fc89]>

open 24030d func #<:function :lambda [req:2, form:244169]>

peek-char 1f23ad func #<:function :lambda [req:3, form:251ee9]>

read-byte 250ded func #<:function :lambda [req:3, form:250cc9]>

read-char 1b4e0d func #<:function :lambda [req:3, form:24fe29]>

read-stream-designator 6afcd func #<:function :lambda [req:1, form:23a809]>

unread-char 1b43cd func #<:function :lambda [req:2, form:253629]>

with-open-file 24696d func #<:function :lambda [req:4, form:24bcc9]>

write 4782d func #<:function :lambda [req:3, form:25dd09]>

write-byte 24ef0d func #<:function :lambda [req:2, form:24ede9]>

write-char 6cc6d func #<:function :lambda [req:2, form:24e4e9]>

write-line 6dd2d func #<:function :lambda [req:2, form:6dc09]>

write-stream-designator 6cf8d func #<:function :lambda [req:1, form:237fe9]>

write-string 6b10d func #<:function :lambda [req:2, form:6c929]>

;;; exceptions

debug b21cd func #<:function :lambda [req:2, form:b168900b204960]>

eprint b348d func #<:function :lambda [req:2, form:b3309]>

error af08d func #<:function :lambda [req:3, form:aef09]>

make-exception a920d func #<:function :lambda [req:5, form:ae069]>

raise 1074d func #<:function :lambda [req:3, form:afc49]>

raise-env b0b2d func #<:function :lambda [req:3, form:b0a09]>

warn b4d2d func #<:function :lambda [req:2, form:b4ba9]>

with-exception abb8d func #<:function :lambda [req:2, form:aba69]>

;;; symbols

genkeyword 160fad func #<:function :lambda [req:1, form:269c09]>

gensym 265f0d func #<:function :lambda [req:0, form:265d89]>

make-keyword 261f8d func #<:function :lambda [req:1, form:264269]>

symbol-name 2f6cd func #<:function :mu [req:1, form:symbol-name]>

symbol-namespace 2f5ed func #<:function :lambda [req:1, form:261949]>

symbol-value fe9ad func #<:function :mu [req:1, form:symbol-value]>

;;; macros

define-symbol-macro 27d78d func #<:function :lambda [req:2, form:27ed69]>

get-macro-character 1f964d func #<:function :lambda [req:1, form:232fe9]>

macro-function 309cd func #<:function :lambda [req:2, form:171cc9]>

macroexpand 16c8cd func #<:function :lambda [req:2, form:16e3e9]>

macroexpand-1 16c52d func #<:function :lambda [req:2, form:16c409]>

set-macro-character 233b2d func #<:function :lambda [req:3, form:2362a9]>

;;; vectors

make-vector 29e14d func #<:function :lambda [req:1, form:29f609]>

vector-displaced-p 29ddad func #<:function :lambda [req:1, form:29dc89]>

vector-length 29faad func #<:function :lambda [req:1, form:2a9269]>

vector-ref 4794d func #<:function :lambda [req:2, form:2b4969]>

vector-slice 29fd0d func #<:function :lambda [req:3, form:2a6049]>

vector-type 2a9a0d func #<:function :lambda [req:1, form:2ac1e9]>

;;; reader

parse-float 1b046d func #<:function :lambda [req:1, form:1b0349]>

parse-integer 1a420d func #<:function :lambda [req:2, form:1ae669]>

read 1b3ecd func #<:function :lambda [req:3, form:257009]>

read-line 25b8cd func #<:function :lambda [req:3, form:25b7a9]>

read-string 6928d func #<:function :lambda [req:3, form:6aa49]>

;;; strings

string 5998d func #<:function :lambda [req:1, form:5c709]>

string-append 5cd0d func #<:function :lambda [req:1, form:61009]>

string-find 53ded func #<:function :lambda [req:2, form:59209]>

string-position 4d9ed func #<:function :lambda [req:2, form:52ea9]>

string= 4108d func #<:function :lambda [req:2, form:46f89]>

substr 6238d func #<:function :lambda [req:3, form:684c9]>

schar 3dc0d func #<:function :lambda [req:2, form:3ffa9]>