

033 GJS 16:32:57

MARCH 12, 1968

DS3 GJS 16:32:59 MARCH 12, 1968

DDDDDDDDDDDDDD	EEE EEEEE EEEEE	CCCCCCCC	222222222
DDDDDDDDDDDDDD	EEE EEEEE EEEEE	CCCCCCCC	222222222
DDDDDDDDDDDDDD	EEE EEEEE EEEEE	CCCCCCCC	222222222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDD	DDD	CCC	222
DDDDDDDDDDDDDD	EEE EEEEE EEEEE	CCCCCCCC	222222222222
DDDDDDDDDDDDDD	EEE EEEEE EEEEE	CCCCCCCC	222222222222
DDDDDDDDDDDDDD	EEE EEEEE EEEEE	CCCCCCCC	222222222222

TO HACK

OT YO: \$Q (10)
OT YI: \$Q (10)

ψ atom finds \$
E \$-to-and E mode
n S string \$
n L
n R
C - doesn't work

20

I (sep) &
B (sep) - enters E mode

P
(\$RB n) - flushes break

$T_{\text{atom}\langle\text{ind}\rangle}$

```

001 (COND ((GET (QUOTE CMP1) (QUOTE SUBR))
002   (SETQ SCOUNT 0)
003   (DEFFPROP GENSYM
004     (LAMBDA NIL
005       (MACRO
006         (APPEND
007           (EXPLODE (W S))
008           (EXPLODE (SETQ SCOUNT
009             (ADD1 SCOUNT)))))))
010   EXPR)
011   (SPECIAL SCEN $SKEL IBASE
012   CHRCT SAA SAL SCURSOR SX SCI SMARK SIND
013   SOB SCOML SAM SOP SCI SVT SFF SSL SSE1
014   SLNS SSE2 SSE3 SPRIN LINEL STOP SBOT
015   SLP SRP SSP SPT SPAN SE SH SCURSOR SCR
016   SLF SCH STB SSEXPR SRD SCU $EXPR)
017   (CMP1)))
018
019   (DEFFPROP MCONS
020     (LAMBDA (L)
021       (COND ((NULL (CDDR L)) (CADR L))
022         (T (LIST (Q CONS)
023           (CADR L)
024           (CONS (CAR L)
025             (CDDR L)))))))
026
027   MACRO)
028
029   (DEFFPROP Q
030     (LAMBDA (L) (CONS (QUOTE QUOTE)
031       (CDR L))))
032
033   MACRO)
034
035   (MAPC
036     (FUNCTION
037       (LAMBDA (X)
038         (SET
039           (CAR X)
040           (INTERN
041             (LIST (CAR NIL)
042               (Q PNAME)
043               (LIST (CDDR (LSH (CADR X)
044                 35)))))))
045
046     (Q ((STB 11) (SLF 12)
047       (SVT 13)
048       (SFF 14)
049       (SCR 15)
050       (SRD 177)
051       (SSP 40)
052       (SCO 54)
053       (SLP 50)
054       (SRP 51)
055       (SP1 56)
056       (SAM 33)
057       (SSL 57))))
```

```

001 (DEFFPROP SSKEL
002   (LAMBDA (S)
003     (COND
004       ((ATOM S) (FLATSIZE S))
005       (((LAMBDA (SX Y)
006         (MCONS (PLUS (COND ((ATOM SX) SX)
007           ((CAR SX)))
008           (COND ((ATOM Y)
009             (PLUS Y 5))
010             ((ADD1 (CAR Y)))))))
011           SX
012           (COND ((ATOM Y) (NCONS Y))
013             ((CDR Y))))))
014           ($SKEL (CAR S))
015           (COND ((CDR S) ($SKEL (CDR S)))
016             ((NCONS 1)))))))
017
018   EXPR)
019
020   (DEFFPROP SWIDTH
021     (LAMBDA (S)
022       (COND ((ATOM S) S)
023         ((ATOM (CADR S))
024           (PLUS 2
025             (CADR S)
026             (SLWIDTH (CDDR S)))))
027           ((ADD1 (SLWIDTH (CDR S)))))))
028
029   (DEFFPROP SLWIDTH
030     (LAMBDA (L)
031       (COND ((NULL L) 0)
032         ((NULL (CDR L))
033           (ADD1 (SWIDTH (CAR L)))))
034           ((MAX (SWIDTH (CAR L))
035             (SLWIDTH (CDR L)))))))
036
037   EXPR)
038
039   (DEFFPROP SCEN1
040     (LAMBDA (SE)
041       (PROG (EE)
042         (COND ((ATOM SE) (RETURN NIL)))
043           (CAR (AND (EQ SE SSE1) (ERR SLNS)))
044           (AND (NULL (CAR SE))
045             (NULL SSE1)
046             (NULL SSE2)
047             (EQ SE SSE3)
048             (ERR SLNS)))
049           (SCEN1 (CAR SE)))
050           (COND ((ATOM (SETQ SE
051             (CDR (SETQ EE
052               SSE1)))
053             (RETURN (AND (ATOM SSE1)
054               (EQ EE SSE2)
055               (ERR SLNS)))))))
056           (GO CAR1)))
057
058   EXPR)
059
060   (DEFFPROP SCEN
061     (LAMBDA (SE SK CC DPTH PN)
062       (PROG
```

```

061 (PG IN EE)
062 (COND (IOR (ATOM SE)
063 (LESSP (CAR SK)
064 (DIFFERENCE LINE1
065 CC
066 DPTH)))
067 (RETURN (SCEN1 SE)))
068 ((ATOM (CAR SE)) (GO ATOM)))
069 (SETQ IN 1)
070 (SETQ SK (CDR SK))
071 (CAR(AND (EQ SE SSE1) (ERR SLNS))
072 (AND (NULL (CAR SE))
073 (NULL SSE1)
074 (NULL SSE2)
075 (EQ SE SSE3)
076 (ERR SLNS))
077 (SCEN (CAR SE)
078 (CAR SK)
079 (PLUS CC IN)
080 (ADD1 DPTH)
081 PN)
082 CDR(SETQ SK (CDR SK))
083 (COND ((ATOM (SETQ SE (CDR (SETQ EE SE))))))
084 (GO DOT)))
085 (SETQ SLNS (ADD1 SLNS))
086 (GO CAR)
087 DOT(AND SE (SETQ SLNS (ADD1 SLNS)))
088 (RETURN (AND (ATOM SSE1)
089 (EQ EE SSE2)
090 (ERR SLNS)))
091 ATOM
092 (AND (EQ SE SSE1) (ERR SLNS))
093 (COND ((EQ (CAR SE) (Q PRUG)) (SETQ IN 6))
094 (*GREAT PN 0)
095 (SETQ IN 1)
096 (SETQ PN (*DIF PN (CADR SK)))
097 (SETQ SLNS (ADD1 SLNS))
098 (SETQ IN (PLUS 2 (CADR SK))))
099 (SETQ SK (CDDR SK))
100 (COND ((NULL (SETQ SE (CDR (SETQ EE SE))))))
101 (GO DOT)))
102 (GO CAR)))
103 EXPR)

104 (DEFFPROP SLINE
105 (LAMBDA NIL
106 (PROG NIL
107 (SETQ SLNS (ADD1 SLNS))
108 (COND (SPRIN)
109 ((SETQ SPRIN
110 (*GREAT SLNS
111 STOP))
112 (RETURN NIL))
113 ((RETURN NIL)))
114 (AND (*GREAT SLNS $BOT) (ERR))
115 (TERPRI)))
116 EXPR)

117 (DEFFPROP SPRINCN
118 (LAMBDA (C N)
119
120

```

```

121 (PROG NIL
122 LOOP (COND ((LESSP (SETQ N
123 (SUB1 N))
124 0)
125 (RETURN NIL)))
126 (PRINC C)
127 (GO LOOP)))
128 EXPR)

129 (DEFFPROP SPRINL
130 (LAMBDA (S)
131 (PROG NIL
132 (SETQ S (EXPLODE S))
133 LOOP (COND ((OR (LESSP CHRCT 10)
134 (NULL S))
135 (RETURN NIL)))
136 (PRINC (CAR S))
137 (SETQ S (CDR S))
138 (GO LOOP)))
139 EXPR)

140 (DEFFPROP SPRINEXPR2
141 (LAMBDA (SE)
142 (PROG (EE)
143 (COND ((ATOM SE) (RETURN (PRINI SE))))
144 (PRINC SLP)
145 CAR (AND (EQ SE SSE1) (PRINC SMARK))
146 (COND ((AND (NULL (CAR SE))
147 (NULL SSE1)
148 (NULL SSE2)
149 (EQ SE SSE3))
150 (PRINC SLP)
151 (PRINC SMARK)
152 (PRINC SRP))
153 ((SPRINEXPR2 (CAR SE))))
154 (COND ((ATOM (SETQ SE
155 (CDR (SETQ EE
156 SSE1)))
157 (GO DOT)))
158 (PRINC SSP)
159 (GO CAR)
160 (PRINC SPT)
161 (GO CAR)
162 DOT (COND (SE (PRINC SSP)
163 (PRINC SPT)
164 (PRINC SSP)
165 (AND (ATOM SSE1)
166 (EQ EE SSE2)
167 (SETQ CHRCT
168 (PLUS 5 CHRCT))
169 (PRINC SMARK))
170 (PRINI SE))
171 (SSE1)
172 (EQ EE SSE2)
173 (SETQ CHRCT (PLUS 5 CHRCT))
174 (PRINC SMARK)))
175 (RETURN (PRINC SRP))))
176 EXPR)

177 (DEFFPROP SPRINEXPR1
178 (LAMBDA (SE SK CC DPTH PN)
179 (PROG

```

```

1 PG IN EE)
2 (COND
3   ((OR (ATOM SK)
4     (LESSP (CAR SK)
5       (DIFFERENCE LINE1 CC DPTH)))
6     (RETURN (AND SPRIN
7       (COND ((EQ SLNS SCEN)
8         (SPRINEXPR2 SE))
9         ((PRINI SE)))))))
10   (AND SPRIN (PRINC SLP)))
11   (COND ((ATOM (CAR SE)) (GO ATOM)))
12   (SETQ IN 1)
13   (SETQ SK (CDR SK))
14 CDR(AND (EQ SE SSE1) (PRINC SMARK))
15   (COND ((AND (NULL (CAR SE))
16     (NULL SSE1)
17     (NULL SSE2)
18     (EQ SE SSE3))
19       (PRINC SLP)
20       (PRINC SMARK)
21       (PRINC SRP))
22     ((SPRINEXPR1 (CAR SE)
23       (CAR SK)
24       (PLUS CC IN)
25       (ADD1 DPTH)
26       PN)))
27   CDR(SETQ SK (CDR SK))
28   (COND ((ATOM (SETQ SE (CDR (SETQ EE SE)))))
29     (GO DOT)))
30   (SLINE)
31   (AND SPRIN
32     (COND ((AND PG (ATOM (CAR SE))
33       (SPRINON SSP (ADD1 CC)))
34       ((SPRINON SSP (PLUS CC IN)))))
35     (GO CAR))
36   DOT(COND (SE (SLINE)
37     (AND SPRIN
38       (SPRINON SSP (PLUS CC IN))
39       (PRINC SPT)
40       (PRINC SSP)))
41     (AND (ATOM SSE1)
42       (EQ EE SSE2)
43       (PRINC SMARK))
44     (AND SPRIN (PRINI SE)))
45   SSE1)
46   ((EQ EE SSE2) (PRINC SMARK)))
47   (RETURN (AND SPRIN (PRINC SRP)))
48 ATOM
49   (AND (EQ SE SSE1) (PRINC SMARK))
50   (AND SPRIN
51     (PROG2 (PRINI (CAR SE)) (PRINC SSP)))
52     (COND ((EQ (CAR SE) (Q PROG))
53       (SETQ PG (SETQ IN 0)))
54       ((GREAT PN 0)
55         (SETQ IN 1)
56         (SETQ PN (*DIF PN (CADR SK))))
57         (SLINE)
58         (AND SPRIN (SPRINON SSP (ADD1 CC))))
59         ((SETQ IN (PLUS 2 (CADR SK))))).
60   (SETQ SK (CDDR SK)))

```

```

241 (COND ((ATOM (SETQ SE (CDR (SETQ EE SE)))))
242 (GO DOT)))
243 (GO CAR)))
244 EXPR)
245
246 (DEFPROP SPRINEXPR
247 (LAMBDA (SE SSE1 SSE2 SSE3)
248 (PROG
249 (SLNS SCEN SPRIN STOP SBOT)
250 (COND (SSKEL)
251 (SETQ SSKEL (SSKEL SE))
252 (SETQ SPAN (*DIF (SKIDTH SSKEL)
253 LINEL))))
254 (SETQ SCEN (ERRSET (SCEN SE
255 SSKEL
256 (SETQ SLNS
257 0)
258 SPAN)))
259
260 (SETQ
261 SPRIN
262 (ZEROP (SETQ STOP
263 (MAX 0
264 (*DIF SCEN
265 (*QUO SH 2)))))))
266 (SETQ SBOT (+PLUS STOP SH))
267 (ERRSET (SPRINEXPRI SE
268 SSKEL
269 (SETQ SLNS 0)
270 0
271 SPAN)))))

272 EXPR)
273
274 (DEFPROP SSHOW
275 (LAMBDA (C SH)
276 (PROG
277 (ISMARK)
278 (SETQ SMARK (Q '+)[/])
279 (COND ((NULL C) (PRINC SLP)
280 (PRINC SMARK)
281 (RETURN (SPRINEXP (CAAR (LAST C))
282 (PRINC (PRINC SRP)))))
283 (ICAR C)
284 (LAST (CAADR C))
285 (CAR C)
286 (LAST (CAADR C))
287 (CAADR C))))))

```

```

001 (ARRAY SLOCAR T 10)
002 (ARRAY SCNDAR T 10)
003 (ARRAY SCOUNTAR T 10)
004
005 (DEFFPROP SCURLOC
006   (LAMBDA (L) (CONS (= CAR) (CDR L)))
007   MACRO)
008
009 (DEFFPROP SCUREXPR
010   (LAMBDA (L)
011     (CONS (= CAAR) (CDR L)))
012   MACRO)
013
014 (DEFFPROP SCURCON
015   (LAMBDA (L) (CONS (= CDR) (CDR L)))
016   MACRO)
017
018 (DEFFPROP SRIGHT
019   (LAMBDA (C)
020     (COND ((AND C (SCURLOC C))
021       (CONS (SCURCON (SCURLOC C))
022         (SCURCON C))))))
023
024 EXPR)
025
026 (DEFFPROP SLEFT
027   (LAMBDA (C)
028     (PROG
029       (R S)
030       (COND ((EQ (SETQ S
031         (SCUREXPR (SCURCON C)))
032           (SETQ R (SCURLOC C)))
033             (RETURN NIL)))
034
035 LOOP
036   (COND ((EQ (SCURCON S) R)
037     (RETURN (CONS S (SCURCON C)))))
038   (SETQ S (CDR S))
039   (GO LOOP)))
040
041 EXPR)
042
043 (DEFFPROP SDOWN
044   (LAMBDA (C)
045     (COND
046       ((AND C
047         (SCURLOC C)
048           (NOT (ATOM (SCUREXPR C))))))
049           (CONS (SCUREXPR C) C)))
050
051 EXPR)
052
053 (DEFFPROP SUP
054   (LAMBDA (C)
055     (COND ((NULL C) NIL)
056       (((LAMBDA (SX)
057         (COND ((AND SX (SCURCON SX)
058           SX)))
059           (SCURCON C))))))
060
061 EXPR)
062
063 (DEFFPROP SKILL

```

```

064 (LAMBDA (C)
065   (COND
066     ((SCURLOC C)
067       (PROG
068         (CUR)
069         (COND
070           ((SCURLOC (SETQ CUR (SRIGHT C)))
071             (RPLACD (RPLACA (SCURLOC C)
072               (CAR (SCURLOC CUR)))
073                 (CUR (SCURLOC CUR))))
074
075             (RETURN C))
076             (SETQ CUR (SLEFT C))
077             (RPLACD (SCURLOC CUR) NIL)
078             (RETURN CUR)))
079             (RETURN NIL))))))
080
081 EXPR)
082
083 (DEFFPROP SOPER
084   (LAMBDA (OP N)
085     (PROG (CUR)
086       (SETQ CUR SCURSOR)
087       LOOP (COND ((AND (*GREAT (SETQ N
088         -1)
089           (SETQ CUR (OP CUR)))
090             (SETQ SCURSOR CUR)
091               (GO LOOP))))))
092
093 (DEFFPROP R
094   (LAMBDA (N)
095     (SUPER (= SRIGHT) N))
096
097 EXPR)
098
099 (DEFFPROP L
100   (LAMBDA (N)
101     (SUPER (= SLEFT) N))
102
103 EXPR)
104
105 (DEFFPROP D
106   (LAMBDA (N)
107     (SUPER (= SDOWN) N)))
108
109 (DEFFPROP U
110   (LAMBDA (N)
111     (SUPER (= SUP) N)))
112
113 (DEFFPROP K
114   (LAMBDA (N)
115     (SUPER (= SKILL) N)))
116
117 EXPR)
118
119 (DEFFPROP SINSERT1
120   (LAMBDA (S C)
121     (COND

```

```

((NULL C)
 (SRIGHT (SDOWN (LIST (LIST (LIST S))))))
 (SCURCON C)
 (COND
  ((SCURLOC C)
   (RPLACA (SCURLOC C)
            (CONS (CAR (SCURLOC C))
                  (CDR (SCURLOC C)))))

  S)
  ((SRIGHT C))
  (SETQ C (SLEFT C))
  (RPLACD (SCURLOC C) (LIST S))
  (SRIGHT (SRIGHT C))))))
(EXPR)

(DEFPROP $INSERT
  (LAMBDA (L)
    (PROG
      (PC $SEXPR)
      (SETQ $SKEL NIL)
      (SETQ PC 0))

  READ
  (COND
    (L)
    ($SEXPR
     (COND
       ((ZEROP PC)
        (SETQ
          SCURSOR
          ($INSERT) (READLIST (REVERSE $SEXPR))
          SCURSOR))
       (RETURN NIL)))
    (T
     ($ILP SCURSOR)
     (COND ((SETQ L
                  (CDR (REVERSE $SEXPR)))).
           ((RETURN NIL)))))

    ((RETURN NIL)))
  (COND
    ((MEMQ (CAR L) (LIST SCR SLF))
     (SETQ L (CDR L)))
    ((AND
      (NOT (NUMBERP SCH))
      (LESSP (ASCI SCH) 40))
     (SETQ L (CDR L)))
    ((MEMQ (CAR L) (LIST SSP STB SVT SFF)))
    (COND
      ($SEXPR
       (COND
         ((ZEROP PC)
          (SETQ L (CDR L)))
         (SETQ
           SCURSOR
           ($INSERT)
           (READLIST (REVERSE $SEXPR))
           SCURSOR))
         (SETQ $SEXPR NIL))
       ((SETQ $SEXPR (CONS (CAR L) $SEXPR))
        (SETQ L (CDR L)))))))

```

```

181 ((SETQ L (CDR L))))))
182 (EQ (CAR L) SLP)
183 (COND
184   ((AND $SEXPR (ZEROP PC))
185    (SETQ
186      SCURSOR
187      (SINSERT1 (READLIST (REVERSE $SEXPR)
188                  SCURSOR)))
189    ((SETQ PC (ADD1 PC))
190     (SETQ $SEXPR (CONS (CAR L) $SEXPR))
191     (SETQ L (CDR L))))))
192   ((EQ (CAR L) SRP)
193    (COND
194      ((ZEROP PC)
195       (SETQ L (CDR L))
196       (COND
197         ($SEXPR
198          (SETQ
199            SCURSOR
200            (SINSERT
201              (READLIST (REVERSE $SEXPR)
202                  SCURSOR)))))
203        (SRP)))
204      ((SETQ PC (SUB1 PC))
205       (SETQ $SEXPR (CONS (CAR L) $SEXPR))
206       (SETQ L (CDR L)))
207       (COND
208         ((ZEROP PC)
209          (SETQ
210            SCURSOR
211            (SINSERT
212              (READLIST (REVERSE $SEXPR)
213                  SCURSOR)))
214          (SETQ $SEXPR NIL))))))
215   ((EQ (CAR L) SSL)
216    (SETQ $SEXPR (MCONS (CAR L) SSL $SEXPR))
217    (SETQ L (CDR L)))
218   ((SETQ $SEXPR (CONS (CAR L) $SEXPR))
219    (SETQ L (CDR L))))
220   (GO READ)))
221 )
222 EXPR)

223 (DEFFPROP SSEG
224 (LAMBDA NIL
225
226   (PROG (S)
227     LOOP (COND ((EQ (CAR SCOML) SAM)
228                 (SETQ SCOML (CDR SCOML))
229                 (RETURN (REVERSE S)))
230                 ((SETQ S
231                   (CONS (CAR SCOML)
232                         S))
233                   (SETQ SCOML (CDR SCOML))
234                   (GO LOOP))))
235
236   (DEFFPROP SS1
237     (LAMBDA (CMS C)
238       (COND ((OR (NULL CMS)
239               (NULL C))

```

```

241 (( (CAR CMS) (CDR CMS)
242 C)
243 )
244
245 (DEFPROP SATM
246 (LAMBDA (CMS C)
247 (COND ((AND (SCURLOC C)
248 (EQUAL (CAR CMS)
249 (SCUREXPRESS C))
250 (SS1 (CDR CMS)
251 (SRIGHT C)))))))
252 )
253
254 (DEFPROP SLPR
255 (LAMBDA (CMS C)
256 (SS1 CMS (SDOWN C))))
257 )
258
259 (DEFPROP SRPR
260 (LAMBDA (CMS C)
261 (COND ((NOT (SRIGHT C))
262 (SS1 CMS (SRIGHT (SUP C))))))
263 )
264
265 (DEFPROP SSEARCH2
266 (LAMBDA (CMS C)
267 (PROG (SC1)
268 A (COND ((SETQ C
269 (SS1 CMS (SETQ SC1 C)))
270 ((SETQ C (SDOWN SC1)) (GO A))
271 ((SETQ C (SRIGHT SC1)) (GO A))
272 ((SETQ C (SRIGHT (SUP SC1)))
273 (GO A)))
274 (RETURN C)))
275 )
276
277 (DEFPROP SSEARCH1
278 (LAMBDA (L)
279 (PROG
280 (CH CMS OP NAME CUR)
281 S1 (SETQ NAME (SETQ OP NIL))
282 S2 (COND (L) ((GO SAM)))
283 (SETQ CH (CAR L))
284 (COND ((EQ CH SLP) (GO SLP))
285 ((EQ CH SRP) (GO SRP))
286 ((EQ CH SSP) (GO SSP))
287 ((MEMO CH (LIST SCR SLP)) (GO S3)))
288 (SETQ NAME (CONS (CAR L) NAME))
289 S3 (SETQ L (CDR L))
290 (GO S2)
291 SLP (SETQ OP (Q SLP))
292 (GO SSP)
293 SRP (SETQ OP (Q SRP))
294 SSP (AND
295 NAME
296 (SETQ CMS
297 (MCONS (READLIST (REVERSE NAME))
298 (Q SATM)
299 CMS)))
300 (AND OP (SETQ CMS (CONS OP CMS))))
```

PAGE 3.

```

301 (SETQ L (CDR L))
302 (GO S1)
303 SAM(AND
304 NAME
305 (SETQ CMS
306 (MCONS (READLIST (REVERSE NAME))
307 (Q SATM)
308 CMS)))
309 (AND (SETQ CUR (SSEARCH2 (REVERSE CMS)
310 SCURSOR))
311 (SETQ SCURSOR CUR))
312 (RETURN NIL)))
313
314 EXPR)

315 (DEFFPROP SSEARCH
316 (LAMBDA (S N)
317 (PROG (C)
318 LOOP (COND ((LESSP N 1)
319 (RETURN C))
320 (T (SETQ C
321 (SSEARCH1 S)
322 (SETQ N (SUB1 N))
323 (GO LOOP)))
324
325 EXPR))

326 (DEFFPROP SGL
327 (LAMBDA (X)
328 (COND ((EQ (CAR X) (Q PNAME)
329 (Q (((SNOT FOUNDS))))
330 ((ATOM (CADR X))
331 (SGL (CDR X)))
332 (T (SETQ SIND (CAR X))
333 (LIST (CADR X)
334 (LIST (CADR X)))))))
335
336 EXPR))

337 (DEFFPROP SYANK
338 (LAMBDA (O I)
339 (PROG2
340 (SETQ
341 SCURSOR
342 (COND
343 ((ATOM (SETQ S08 O))
344 (COND
345 ((SETQ SIND I)
346 ((LAMBDA (X)
347 (COND (X (LIST X (LIST X)))
348 ((Q (((SNOT FOUNDS))))))
349 (GET O I)))
350 (T (SETQ SIND NIL) (SGL (CDR O)))))
351 ((LAMBDA (X)
352 (COND ((ATOM X)
353 (Q (((SNOT FOUNDS)))
354 (EVAL O))))))
355 (SETQ SSKEL NIL)))
356 EXPR))

357 (DEFFPROP SFILE
358 (LAMBDA (O I)
359
360

```

PAGE 3.5

```

361      ((LAMBDA (SX Y)
362        (COND ((ATOM SX)
363          (PUTPROP SX Y (COND (()) ($IND)))
364          ((PROG2 (RPLACA SX (CAR Y))
365            Y
366            (RPLACD SX (CDR Y))))))
367        (COND (0) ($OB))
368        (COND (SCURSOR (CAAR (LAST SCURSOR))))))
369      EXPR)
370
371      (DEFFPROP SCURSEG
372        (LAMBDA (SX Y Z)
373          (COND ((EQ SX Y) Z)
374            (T (SCURSEG SX
375              (CDR Y)
376              (CDR Z)))))))
377      EXPR)
378
379      (DEFFPROP SNEWCUR
380        (LAMBDA (X Y)
381          (COND ((NULL X) X)
382            ((NULL (CDR X)) Y)
383            (((LAMBDA (Z)
384              (CONS (SCURSEG (CAR X)
385                (CAADR X)
386                (CAAR Z)))
387                Z))
388              (SNEWCUR (CDR X) Y))))))
389      EXPR)
390
391      (DEFFPROP SREMBREAK
392        (LAMBDA (N)
393          ((LAMBDA (L)
394            (COND (L (RPLACA (CAR (CADDR L))
395              (CDDAAR (CADDR L)))
396              (STORE (SLOCAR N) NIL)
397              (STORE (SCONDAR N) NIL)
398              (STORE (SCOUNTAR N) NIL)
399              (SETQ SSKEL NIL))))
400              (SLOCAR N)))
401            EXPR))
402
403      (DEFFPROP SRB
404        (LAMBDA (X)
405          (COND
406            ((NULL X)
407              (PROG (N)
408                (SETW N 7)
409                LOOP (SREMBREAK N)
410                (COND ((MINUSP (SETQ N (SUB1 N)))
411                  (RETURN NIL))
412                  ((GO LOOP))))))
413              ((MAPC (FUNCTION SREMBREAK) X))))))
414      FEXPR)
415
416      (DEFFPROP SSETBREAK
417        (LAMBDA (I C N CND)
418          (PROG (PROGL)
419            (SETQ PROGL (COND (C (CAR C))))
420            (COND ((ATOM PROGL) (RETURN NIL))))
```

```

421      (SREMBREAK N)
422      (RPLACA (RPLACA PROGL
423        (CONS N
424          (CAR PROGL)))
425        (CONS (Q $) (CAR PROGL)))
426        (STORE (SLOCAR N) (LIST I O C))
427        (STORE (SCONDAR N)
428        (COND ((NULL CND) T) (CND)))
429        (STORE (SCOUNTAR N) 0)
430        (RETURN NIL)))
431      EXPR)
432
433      (DEFFPROP SBREAK
434        (LAMBDA (SE SAA)
435          ((LAMBDA (SC)
436            ((LAMBDA (SL SCN SCT)
437              (COND ((AND (NUMBERP SCN)
438                (*GREAT SCT SCN))
439                  (TECO SL)
440                  (CDR SE))
441                  (EVAL SCN SAA)
442                  (TECO SL)
443                  (CDR SE))))
444              (SLOCAR SC)
445              (SCONDAR SC)
446              (SCOUNTAR SC)))
447              (CAR SE)))
448            EXPR))
449
450      (DEFFPROP S
451        (LAMBDA (SX SA)
452          (EVAL (SBREAK SX SA) SA)))
453        FEXPR)
454
455      (DEFFPROP SLSP
456        (LAMBDA (L)
457          (AND (EQ (CAR L) SSL)
458            (NOT (SLSP (CDR L))))))
459        EXPR)
460
461      (DEFFPROP SASCI
462        (LAMBDA (X)
463          (LSH (IMAKNUM (CAAR (GET X (G PNAME)))
464            (QUOTE FIXNUM))
465            -13)))
466        EXPR)
467
468      (DEFFPROP SREAD
469        (LAMBDA NIL
470          (PROG
471            (PC S$EXPR IGLIST)
472            (SETQ PC 0)
473            (SETQ IGLIST (LIST SCR SLF)))
474          READ
475          (OR SCH (SETQ SCH (READCH)))
476          (COND
477            ((EQ SCH SAM)
478              (SETQ SCH NIL)
479              (RETURN
480                (COND
```



```

601      (RETURN
602          (READLIST (REVERSE $SEXPR))))))
603      (SETQ PG (SUB1 PC))
604      (SETQ $SEXPR (CONS SCH $SEXPR))
605      (SETQ SCOML (CDR SCOML))
606      (COND
607          ((ZEROP PG)
608              (RETURN
609                  (READLIST (REVERSE $SEXPR))))))
610          ((EQ SCH SSL)
611              (SETQ $SEXPR
612                  (MCOMS (CAR SCOML) SSL $SEXPR))
613              (SETQ SCOML (CDR SCOML)))
614          ((SETQ $SEXPR (CONS SCH $SEXPR))
615              (SETQ SCOML (CDR SCOML))))
616          (GO READ)))
617      (GO READ)))
618  EXPR)

(DEFPROP I
  (LAMBDA ($X) (INSERT (CAR $X)))
  FEXPR)

(DEFPROP S
  (LAMBDA (L) (COND ((NULL L) NIL)
    ((NULL (CDR L))
     (ISSEARCH (CAR L) 1))
    ((ISSEARCH (CAR L)
    (CAR L)
    (CADR L))))))

(DEFPROP Y
  (LAMBDA (L)
    (COND ((NULL L) NIL)
      ((NULL (CDR L))
       (SYANK (CAR L) NIL))
      ((SYANK (CAR L)
      (CAR L)
      (CADR L))))))

(DEFPROP F
  (LAMBDA (L)
    (COND ((NULL L)
    ($FILE NIL NIL))
      ((NULL (CDR L))
       ($FILE (CAR L) NIL))
      ((($FILE (CAR L)
      (CAR L)
      (CADR L)))))))

(DEFPROP T
  (LAMBDA NIL
    (SETQ SCURSOR
      (CONS (CAAR (LAST SCURSOR))
        (LAST SCURSOR)))))

(DEFPROP C
  (LAMBDA NIL
    (SETQ SCURSOR
      ($NEWCUR (SUBST 0

```

```

661      0
662      (LAST SCURSOR)))))

(DEFPROP TECO
  (LAMBDA ($X $AL)
    (PROG
      (SCURSOR SCH
        IGLIST
        NUM
        $SKEL
        SPAN
        SCOML
        S0B
        $IND
        $EXPR)
      (SETQ IGLIST
        (LIST SCR SLF SAM SSP STB SCO SAM SRP))
      (PRINC $FF)
      (COND ($X (SETQ $X (EVAL (CAR $X)))
        (SETQ $UB (CAR $X))
        (SETQ $IND (CADR $X))
        (SETQ SCURSOR (CADDR $X))
        (GO EVALMODE)))
        (TERPRI)))
      LOOP
        (PRINC $FF)
        (SETQ SCOML NIL)
        (COND ($X (PRINC (Q BREAK))
          (PRINC SSP)
          (PRINC (CADAAR (CADDR $X)))
          (PRINC SSP)
          ($PRINT (CDDAAR (CADDR $X)))
          ($PRINT (SPRINTN SPT 3))
          (TERPRI)
          (TERPRI)))
        ($SHOW SCURSOR (COND ($X 23) (26)))
        (TERPRI)
        (TERPRI))
      READ
        (SETQ SCOML NIL)
      READ1
        (COND
          ((EQ (SETQ SCH (READCH)) SAM)
            (COND ((EQ (CAR SCOML) SAM)
              (SETQ SCOML (REVERSE SCOML))
              (GO LOOP1))
              ((SETQ SCOML (CONS SAM SCOML))))))
          ((EQ SCH S0B)
            (COND
              (SCOML
                (COND ((SLSP (CDR SCOML))
                  (PRINC SSL)
                  (PRINC (CAR SCOML))
                  (SETQ SCOML (CDR SCOML)))
                  ((PRINC (COND ((EQ (CAR SCOML)
                    SAM)
                    (L S))
                    ((CAR SCOML)))))))
                (SETQ SCOML (CDR SCOML)))))))

```

```

721     (INTERPRI)
722     ((EQ SCH SBL)
723     (SETQ SCOML (COND (READCH) SBL SCOML)))
724     ((BETW SCOML (COND SCH SCOML)))
725     (GO READ))
726   LOOP1
727     (SETQ NUM NIL)
728   LOOP2
729     (COND ((NULL SCOML) (GO LOOP)))
730     ((EQ (SETQ SCH (CAR SCOML)) SLP)
731     (GO EVAL))
732     ((BETW SCOML (CDR SCOML)))
733     (COND ((NUMBERP SCH)
734     (SETQ NUM
735       (+ SCH
736         (* TIMES IBASE
737           (COND (NUM)
738             (0)))))))
739     (GO LOOP2))
740     ((OR (LESSP (SASC1 SCH) 40)
741     (MEMQ SCH IGLIST)))
742     (MEMQ SCH (U (R L D U K)))
743     (SCH (COND (NUM) (1))))
744     ((EQ SCH (Q I)) ($INSERT (SSEG)))
745     ((EQ SCH (Q Y)) (SYANK (SEOREAD)
746       (SEOREAD)))
747     ((EQ SCH (Q F)) (SFILE (SEOREAD)
748       (SEOREAD)))
749     ((EQ SCH (Q C)) (C))
750     ((EQ SCH (Q S)) (SSearch (SSEG)
751       (COND (NUM)
752         (1)))))
753     ((EQ SCH (Q P)) (GO PROCEED))
754     ((EQ SCH (Q E)) (GO BREAK))
755     ((EQ SCH (Q E1)) (GO EVALMODE))
756     ((EQ SCH (Q T1)) (T1))
757     (T (TERPRI) (PRINC SCH) (PRINC (Q / NOT/ OP)) (TERPRI) (GO READ)))
758     (GO LOOP1))
759   BREAK
760     (SSETBREAK SOB
761       SIND
762       SCURSOR
763       (COND (NUM) (0)))
764     (SEOREAD))
765     (SETQ SSKEL NIL)
766     (GO LOOP1))
767   EVAL
768     (SETQ SEXPR (ERRSET (SEOREAD)))
769     (COND (SEXPR) ((GO LOOP)))
770     (COND ((ERRSET (EVAL (CAR SEXPR))) (GO LOOP)))
771     (GO LOOP1))
772   EVALMODE
773     (PRINC SFF)
774     (SETQ SCH NIL)
775     (COND (EX (PRINC (Q BREAK))
776       (PRINC SSP)
777       (PRINC (CADDR (CADDR SX)))
778       (PRINC SSP)
779       (SPRINT (CDDAAR (CADDR SX)))))


```

```

781     (SPRINTN SSP 3)
782     (TERPRI)
783     (TERPRI))
784     (SSHOW SCURSOR 10)
785     (TERPRI)
786     (PRINC (Q LISTENING))
787     (TERPRI)
788   LISTEN
789     (TERPRI)
790     (SETQ SEXPR (ERRSET (SBREAD)))
791     (COND (SEXPR) ((GO LISTEN)))
792     (SETQ SEXPR (CAR SEXPR))
793     (PRINC SSP)
794     (COND
795       ((EQ SEXPR (R CONTINUE)) (TERPRI)
796        (GO LOOP)))
797       ((EQ SEXPR (Q PROCEED)) (TERPRI)
798        (GO PROCEED)))
799     (ISETQ SEXPR
800       (ERRSET (EVAL SEXPR
801         (COND (EX SAA)
802           (SAL))))))
803     (PRINI (CAR SEXPR)))
804     (GO LISTEN)
805   PROCEED
806     (PRINC SFF)
807     (RETURN NIL)))
808   FEXPR
809
810


```