

1 DEFINE(#NEXTITEM(NULL)#, #NEXTITEM#)
2 DEFINE(#NEWSTATE(OLD,TKN)#, #NEWSTATE#)
3 DEFINE(#NEXTTKN(NULL)#, #NEXTTKN#)
4 DEFINE(#NEXTFNC(NULL)#, #NEXTFNC#)
5 DEFINE(#EXTEND(KEY)#, #EXTEND#)
6 DEFINE(#EXTENDSS(NULL,FLAG,STATE,TKN,NSTATE,XSTATE))
7 DEFINE(#ASSEM(LOC,OP,RAND,COMM,A,B))
8 DEFINE(#UNSAVE(NULL)#)
9 DEFINE(#GET(NULL)#)

*

10 DEFINE(#DUMPFNC(NULL)#, #DUMPFNC#)
11 DEFINE(#DUMPTKN(NULL)#, #DUMPTKN#)
12 DEFINE(#DUMPSTATET(NULL)#, #DUMPSTATET#)
13 DEFINE(#DUMPRULES(NULL)#, #DUMPRULES#)
14 DEFINE(#DUMPNONTERM(NULL)#, #DUMPNONTERM#)
15 DEFINE(#PRINTSS(SS,FLAG),NM,S#)
16 DEFINE(#PRINTRULES(NULL),LINE#)
17 DEFINE(#PRINTS(S),NM,LN,T#)
18 DEFINE(#PRINTTKNS(NULL)#)

*

*

19 DATA(#LISTEL(VALUE,NEXT)#)

*

20 F1 = #F1.#
21 F2 = #F2.#
22 F3 = #F3.#
23 T1 = #T1.#
24 T2 = #T2.#
25 T3 = #T3.#
26 T4 = #T4.#
27 T5 = #T5.#
28 T6 = #T6.#
29 T7 = #T7.#
30 T8 = #T8.#
31 T9 = #T9.#
32 T10 = #T10.#
33 T11 = #T11.#
34 S2 = #S2.#
35 S4 = #S4.#
36 S6 = #S6.#
37 R4 = #R4.#
38 R5 = #R5.#
39 R6 = #R6.#
40 R7 = #R7.#
41 R8 = #R8.#
42 S51 = #S51.#
43 S52 = #S52.#
44 S54 = #S54.#
45 SSS = #SSS.#

*

46 DETACH(#INPUT#)
47 INPUT(#INPUT#, #GRAMMAR#, 80)
48 OUTPUT(#ASSEMLINE#, #SCMOD#, ##)
49 INPUT(#UNSAVELINE#, #SAVFF#, 80)

```

50          RF-IND( *SAVEF# )
*
*
51          SYNERR = **** SYNT/X ERROR: #
*
*
52          : (50)
*
53          NEXTTKN ITEM = NEXTITEM()
54          IDENT( ITEM, #,* ) :S(FRETURN)
55          NEXTTKN = $( T1 ITEM )
56          IDENT( NEXTTKN, ## ) :F( RETURN ) S( FAIL1 )
*
57          NEXTFNC ITEM = NEXTITEM()
58          IDENT( ITEM, #,* ) :S(FRETURN)
59          NEXTFNC = $( F1 ITEM )
60          IDENT( NEXTFNC, ## ) :F( RETURN ) S( FAIL2 )
*
61          NEXTITEM LINE SPAN(* *) BREAK(* *) . NEXTITEM = ## :S( RETURN )
62          NEXTITEM1 LINE = INPUT
63          OUTPUT = LINE
64          LINE POS(0) ## :F( NEXTITEM ) S( NEXTITEM1 )
*
65          NEWSTATE NEWSTATE = $( S4 OLD #,* TKN )
66          IDENT( $( S4 OLD #,* TKN ), ## ) :F( RETURN )
67          STATECNT = STATECNT + 1
68          $( S2 STATECNT ) = $( S2 OLD ) ## TKN
69          $( S4 OLD #,* TKN ) = STATECNT
70          NEWSTATE = STATECNT :F( RETURN )
*
*
*
*
*
    EXTEND A LIST BY TRANSITIVITY THROUGH NON TERMINALS
*
71          EXTEND = TKN = TERMTKNCNT + 1
72          EXTEND1 GT( TKN, TKNCNT ) :S( EXTEND2 )
73          $( T6 TKN ) = $X2
74          TKN = TKN + 1 :F( EXTEND1 )
*
75          EXTEND2 FLAG = %OFF%
76          TKNH0 = TERMTKNCNT + 1
*
77          EXTEND3 GT( TKNHD, TKNONT ) :S( EXTEND9 )
78          CHKTKN = $( T6 TKNHD )
79          $( T6 TKNHD ) = $( KEY TKNHD )
80          TKN = $( KEY TKNHD )
*
81          EXTEND4 IDENT( TKN, CHKTKN ) :S( EXTEND8 )
82          LE( TKN, TERMTKNCNT ) :S( EXTEND7 )
83          SUBTKN = $( KEY TKN )
*
84          EXTEND5 IDENT( SUBTKN, #X# ) :S( EXTEND7 )
85          IDENT( $( KEY TKNHD #,* SUBTKN ), #X# ) :F( EXTEND6 )

```

CAL S N 0 8 0 L 22 FEB 71

12:41:55 PAGE 3

```

86      $(KEY TKNHD #.≠ SUBTKN) = $(KEY TKNHD)
87      $(KEY TKNHD) = SUBTKN
88      FLAG = #ON#
89      EXTEND6 SUBTKN = $(KEY TKN #.≠ SUBTKN)           ;(EXTENDS5)
90      EXTEND7 TKN = $(KEY TKNHD #.≠ TKN )           ;(EXTENDA4)
91      EXTEND8 TKNHD = TKNHD + 1           ;(EXTEND3)
92      EXTEND9 IDENT( FLAG, #OFF# )           ;S(RETURN)F(EXTEND2)
93      EXTENDSS STATE SET REPRFSENTED IN ARRA SET
94      EXTENDSS1 SETX SET AND SETX EMPTY AT END
95      EXTENDSS2 GT( STATE, STATECNT) :S(EXTENDSS5)
96          IDENT( SET[STATE], #2# ) :S(EXTENDSS4)
97          IDENT( SETX[STATE], #1# ) :S(EXTENDSS4)
98          SETX[STATE] = 1
99          TKN = TERMTKNCNT + 1
100     EXTENDSS3 GT( TKN, TKNCNT) :S(EXTENDSS6)
101         STATE = $(S4 STATE #.≠ TKN )
102         IDENT( STATE, #2# ) :S(EXTENDSS3)
103         XSTATE = $(S4 0 #.≠ TKN )
104         IDENT( SET[XSTATE], 1 ) :S(EXTENDSS3)
105         SETX[XSTATE] = 1
106         FLAG = #ON#
107     EXTENDSS4 TKN = TKN + 1           ;(EXTENDSS2)
108     EXTENDSS5 STATE = STATE + 1           ;(EXTENDSS1)
109     EXTENDSS6 IDENT( FLAG, #OFF# )           ;P(EXTENDSS)
110         STATE = 1
111         SSTATE = #2#
112     EXTENDSS7 GT( STATE + STATECNT) :S(EXTENDSS8)
113         IDENT( SET[STATE], #2# ) :S(EXTENDSS7)
114         SSTATE = SSTATE #.≠ STATE
115     EXTENDSS8 SET[STATE] = #2#
116         SETX[STATE] = #2#
117         STATE = STATE + 1           ;(EXTENDSS6)

```

```

118  EXTENDSS EXTENDSS = $(SSI SSTATE)
119      INENT( EXTENDSS, ## ) IF( RETURN )
120      SSCNT = SSCNT + 1
121      EXTENDSS = SSCNT
122      $(SSI SSTATE) = SSCNT
123      $(SS2 SSCNT) = SSTATE ;(RETURN)
#
#
#

```

OUTPUT AN ASSEMBLY LINE

```

124  ASSEM A = LOC * *
125      A = TAR(1) . B = ##
126      A = R OP * *
127      A = TAR(2) . B = ##
128      INENT( COMM, ## ) IS(ASSEM1)
129      A = R RAND *
130      A = TAR(3) . B = ##
131      ASSEMLINE = B COMM ;(RETURN)
#
ASSEM1 ASSEMLINE = B RAND ;(RETURN)
#

```

READ IN INFO FROM PASS 1

```

133  UNSAVE TERMTCNT = GET()
134          TKNCNT = GET()
135          RULECNT = GET()
136          FNCCNT = GET()
137          STATECNT = GET()
138  UNSAVE1 ITEM1 = GET()
139          IDENT(ITEM1,#####*) IS(UNSAVE2)
140          $(ITEM1) = GET() IS(UNSAVE1)
141  UNSAVE2 OUT( *ITEMS IN = *ITEMCNT ) ;(RETURN)
#
GET      GET = TRIM( UNSAVELINE )
ITEMCNT = ITEMCNT + 1 ;(RETURN)
#
#
#
#
#

```

DUMP FUNCTION DATA

```

144  DUMPFNC I = 1
145          OUTPUT = ##
146          OUTPUT = * FUNCTION TABLE*
147          OUTPUT = ##
148  DUMPFNC1 GT( I, FNCCNT ) ;(RETURN)
149          OUTPUT = I ## $(F2 I) ## $(F3 I) ## $(F1 $(F2 I))

```

```

150           I = I + 1 : (DUMPFNC1)
*
*
*
*          DUMP TOKEN TABLE
*
151 DUMPTKN  OUTPUT = ***
152             OUTPUT = **
153             OUTPUT = *      TOKENS TERMINAL *
154             OUTPUT = **
155             I=1
156 DUMPTKN1 GT( I, TERMTKNCNT ) :S(DUMPTKN2)
157             PART = T ** $ (T2 I) ** $ (T3 I) ** $ (T4 I)
158             OUTPUT = PART ** $ (T1 $ (T2 I) )
159             I = I + 1 : (DUMPTKN1)
*
160 DUMPTKN2 OUTPUT = **
161             OUTPUT = *      TOKENS NON TERMINAL*
162             OUTPUT = **
*
163 DUMPTKN3 GT( I, TKNCNT ) :S(RETURN)
164             OUTPUT = I ** S(T2 I) ** S(T3 I) ** S(T1 $ (T2 I) )
165             I = I + 1 : (DUMPTKN3)
*
*
166 DUMPSTATET   OUTPUT = **
167             OUTPUT = **
168             OUTPUT = *      STATE TABLE*
169             I=0
*
170 DUMPSTATET1  GT( -I, STATECNT ) :S(RRETURN)
171             OUTPUT = I ** S(S2 I) ** S(SS I)
172             J = 1
173 DUMPSTATET2  GT(J, TKNCNT ) :S(DUMPSTATET4)
174             INENT( $ (S4 I ** J) ) ** :S(DUMPSTATET3)
175             OUTPUT = *      & J ** S(qq I ** J )
176 DUMPSTATET3  J = J + 1 : (DUMPSTATET2)
177 DUMPSTATET4  I = I + 1 : (nUMPSTATET1)
*
*
178 DUMPRULES  OUTPUT = **
179             OUTPUT = **
180             OUTPUT = *      RULE TABLE*
181             OUTPUT = **
182             I = 1
183 DUMPRULES1  GT( I, RULECNT ) :S(RRETURN)
184             OUTPUT = I ** S(R4 I) ** S(R5 I) ** S(R6 I)
185             OUTPUT = *      & S(R7 I)
186             E = $ (R8 I)
187 DUMPRULES2  INENT(E, **) :S(DUMPRULES3)
188             OUTPUT = *      & VALUE(E)
189             E = NEXT(E) : (DUMPRULES2)
190 DUMPRULES3  I = I + 1 : (nUMPRULES1)
*

```

*
** DUMP NON TERMINAL EXTRA INFO
*
191 DMPNONTERM OUTPUT = **
192 OUTPUT = **
193 OUTPUT = # NON TERMINAL EXTRA INFO #
194 OUTPUT = **
195 I = TERMTKNCNT + 1
*
196 DMPNONTERM1 GT(I, TKNANT) IS (RETURN)
197 OUTPUT = I * \$ (T2 I)
198 OUTPUT = # INITIALS#
199 J = \$ (TS I)
200 DMPNONTERM1A IDENT(J, **) IS (DMPNONTERM2)
201 OUTPUT = * * J
202 J = \$ (TS I * J) IS (DMPNONTERM1A)
*
203 DMPNONTERM2 OUTPUT = **
204 OUTPUT = # INITIAL TERMINALS#
205 J = \$ (T7 I)
206 DMPNONTERM3 IDENT(J, **) IS (DMPNONTERM4)
207 OUTPUT = # * J
208 J = \$ (T7 I * J) IS (DMPNONTERM3)
209 DMPNONTERM4 OUTPUT = **
210 OUTPUT = # FINALS#
211 J = \$ (T8 I)
212 DMPNONTERMS IDENT(J, **) IS (DMPNONTERMS)
213 OUTPUT = # * J
214 J = \$ (T8 I * J) IS (DMPNONTERMS)
*
215 DMPNONTERM6 OUTPUT = **
216 OUTPUT = # FINAL NON TERMINALS#
217 J = \$ (T9 I)
218 DMPNONTERM7 IDENT(J, **) IS (DMPNONTERM8)
219 OUTPUT = # * J
220 J = \$ (T9 I * J) IS (DMPNONTERM7)
*
221 DMPNONTERM8 OUTPUT = **
222 OUTPUT = **
223 OUTPUT = # FOLLOWING TERMINALS#
224 OUTPUT = **
225 J = \$ (T10 I)
226 DMPNONTERM9 IDENT(J, **) IS (DMPNONTERM10)
227 OUTPUT = # * J
228 J = \$ (T10 I * J) IS (DMPNONTERM9)
*
229 DMPNONTERM10 I = I + 1 IS (DMPNONTERM1)
*
*
* PRINT OUT A STATE SET
*
*
*
230 PRINTSS NM = \$(SS2 SS) *, #

231 PRINTSS1 NM = # BREAK(##) . S = ## :F(RETURN)
232 PRINTS(S)
233 INENT(FLAG, ##) :S(PRINTSS1),
234 OUT(OUTPUT) : (PRINTSS1)

*

*

*

PRINT OUT A STATE

235 PRINTS NM = \$(\$2 S) ##
236 LN = * ##
237 PRINTS1 NM = # BREAK(##) . T = ## :F(PRINTS2)
238 LN = LN ## \$(\$2 T) : (PRINTS1)
239 PRINTS2 OUTPUT = LN : (RETURN)

*

*

PRINT OUT RULE TABLE

240 PRINTRULES OUTPUT = ##
241 OUTPUT = ##
242 OUTPUT = * RULE INFO*
243 OUTPUT = ##
244 I = 1
245 PRINTRULES1 GT(I, RULECNT) :S(RETURN)
246 LINE = * ## I ## ## \$(\$2 \$(\$4 I)) ## ## \$(\$2 \$(\$5 I))
247 LINE = LINE ## ## \$(\$2 I)
248 OUTPUT = LINE
249 I = I + 1 : (PRINTRULES1)

*

*

PRINT OUT TOKEN INFO
AND PLACE IN ASSEMBLY ALSO

250 PRINTTNS OUTPUT = ##
251 OUTPUT = ##
252 OUTPUT = * TOKEN DEFINITIONS*
253 OUTPUT = ##
254 I = 1
255 PRINTTKNS1 GT(I, TKNCNT) :S(RETURN)
256 TEMP = \$(\$2 I) ## ##
257 TEMP = TAB(24) . TEMP1 = ##
258 OUTPUT = TEMP1 \$(\$3 I)
259 ASSEM(#XXX#; #SET#; \$(\$2 I); \$(\$2 I))
260 I = I + 1 : (PRINTTKNS1)

*

*

261 60 UNSAVE()

*

*

*

262 MMESS = ARRAY(#1:# -TKNCNT =)

```

263      MVER = ARRAY( #1:# TKNCNT )
264      SET = ARRAY( #1:# STATECNT )
265      SETX = ARRAY( #1:# STATECNT )
266      SETZ = ARRAY( #1:# STATECNT )

```

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

*

CAL SNOBOL

22 FEB 71

12:41:55

PAGE 9

301 S = SETZ01
302 K = \$(S4 S #. # TKN)
303 IDENT(K, ##) IS(S56)
304 SETK1 = 1
305 TFLAG = #ON#
306 *
306 S56 I = I + 1 : (SS5)
307 *
307 SS7 IDENT(TFLAG, #OFF#) IS(SS8)
308 MVESS(TKN) = EXTENDSS()
309 *
309 SS8 TKN = TKN + 1 IS(SS9)
310 *
310 *
310 SS9 I = 1
311 *
311 SS10 GT(I, SCNT) IS(Ss13)
312 . SE = SETZ11
313 R = \$(S5 S)
314 IDENT(R, ##) IS(Ss12)
315 NTKN = \$(R4 R)
316 T = \$(T1) NTKN
317 *
317 SS11 IDENT(T, #X#) IS(Ss12)
318 IDENT(MVESS(T), ##) IS(SS11A)
318 *
319 ASSEM(##, #ERR#, ##, #RULE AND STATE MOVE CONFLICT#)
320 OUTPUT = ##
321 OUTPUT = ##
322 OUTPUT = #*****# ERROR: RULE AND STATE MOVE CONFLICT#
323 OUT(OUTPUT)
324 OUTPUT = #STATE SET = * SS * # \$ (SS2 SS)
325 OUT(OUTPUT)
326 PRINTSS(SS, #YES#)
327 OUTPUT = #TOKEN = # \$ (T2 T)
328 OUT(OUTPUT)
329 OUTPUT = ##
330 OUTPUT = ##
330 *
331 *
331 SS11A IDENT(MVERCT1, ##) IS(SS11B)
332 ASSEM(##, #ERR#, ##, #2 RULES WITH SAME TOKEN#)
333 OUTPUT = ##
334 OUTPUT = ##
335 OUTPUT = #*****# ERROR: 2 RULES WITH SAME TOKEN#
336 OUT(OUTPUT)
337 OUTPUT = #STATE SET = * SS * # \$ (SS2 SS)
338 OUT(OUTPUT)
339 PRINTSS(SS, #YES#)
340 OUTPUT = #TOKEN = # \$ (T2 T)
341 OUT(OUTPUT)
342 OUTPUT = ##
343 OUTPUT = ##
343 *
344 SS11B MVERCT1 = R

345 T = \$(T1) NTKN #.# T) :(SS11)
*
*
346 SS12 I = I + 1 :(SS10)
*
347 SS13 T = I
348 SS13A GT(T, TERMTKNCNT) :S(SS14)
349 IDENT(MMESS[T], ##) :F(SS13B)
350 IDENT(MVER[T], ##) :S(SS13D)
351 SS13B IDENT(TYPESTKN, ##) :S(SS13C)
352 IDENT(TYPESTKN, \$ (T4 T)) :S(SS13D)
353 ASSEM(##, #ERR#, ##, #TOKEN TYPES CONFLICT#)
354 OUTPUT = ##
355 OUTPUT = ##
356 OUTPUT = ##### ERROR: TOKEN TYPES CONFLICT#
357 OUT(OUTPUT)
358 OUTPUT = #STATE SET = * SS # * \$ (SS2 SS)
359 OUT(OUTPUT)
360 PRINTSS(SS, #YES#)
361 OUTPUT = #KINDS OF GETTKN ARE * TYPESTKN * AND * \$ (T4 T)
362 OUT(OUTPUT)
363 OUTPUT = ##
364 OUTPUT = ##
365 :(SS13C1)
366 SS13C TYPESTKN = \$ (T4 T)
*
367 SS13C1 . IDENT(SKEY, ##) :S(SS13C2)
368 IDENT(SKEY, \$ (T11 T)) :S(SS13D)
369 ASSEM(##, #ERR#, ##, #TOKEN SUB KEYS CONFLICT#)
370 OUTPUT = ##
371 OUTPUT = ##
372 OUTPUT = ##### ERROR: TOKEN SUBKEYS CONFLICT#
373 OUT(OUTPUT)
374 OUTPUT = #STATE SET = * SS # * \$ (SS2 SS)
375 OUT(OUTPUT)
376 PRINTSS(SS, #YES#)
377 OUTPUT = #KIND OF GETTKN IS * TYPESTKN
378 OUT(OUTPUT)
379 OUTPUT = #SUBKEY TYPES ARE * SKEY * AND * \$ (T11 T)
380 OUT(OUTPUT)
381 OUT(OUTPUT)
382 OUTPUT = ##
383 OUTPUT = ##
384 :(SS13D)
*
385 SS13C2 . SKEY = \$ (T11 T)
386 SS13D T = T + 1 :(SS13A)
*
387 SS14 OUTLINE = SS # * \$ (SS2 SS) # * TYPESTKN # * \$ (SS5 SS)
388 OUTPUT = OUTLINE # * SKEY
389 OUTPUT = ##
390 PRINTSS(SS)
391 ASSEM(#S, # SS, #VFD#, #30/# SKEY #, #30/# TYPESTKN)
392 OUTPUT = ##

```

393          T = 1
394  SS14A   GT( T, TKNCNT ) :S(SS15)
395          IDENT( MVESS[T], ** ) :S(SS14B)
396          OUTPUT = * STATE # $(T2 T) # * MVESS[T]
397          RAND = #12/0,18/S. # MVESS[T] #,18/* $(T3 T) #,12/0#
398          ASSEM( **, #VFD#, RAND ) :T(SS14C)
399  SS14B   IDENT( MVER[T], ** ) :S(SS14C)
400          OUTPUT = * RULE # $(T2 T) # * MVER[T]
401          RAND = #12/0,18/R. # MVER[T] #,18/* $(T3 T) #,12/1#
402          ASSEM( **, #VFD#, RAND )
#
403  SS14C   T = T + 1 :T(SS14A)
*
404  SS15    OUTPUT = * FAIL#
405          ASSEM( **, #DATA#, 5 )
406          IDENT( S(SS5 SS), ** ) :S(SS15B)
407          ASSEM( S(SS5 SS), #EoU#, **, SS )
#
408  SS15B   SS = SS + 1 :T(SS1)
*
409  SSEND
*
410          ASSEM( #F#, #BSS#, #0# )
411          ASSEM( #Z#, #END#, ** )
*
412          :END
*
*
413  FAIL1   OUT( SYNERR ITEM # NOT A TOKEN# ) :T(FAIL)
414  FAIL2   OUT( SYNERR ITEM # NOT A FUNCTION# ) :T(FAIL)
415  FAIL4   OUT( SYNERR # PERIOD EXPECTED# ) :T(FAIL)
416  FAIL5   OUT( SYNERR # # EXPECTED ** ) :T(FAIL)
417  FAIL6   OUT( SYNERR # # # EXPECTED ** ) :T(FAIL)
418  FAIL7   OUT( SYNERR # PERIOD NOT EXPECTED# ) :T(FAIL)
419  FAIL8   OUTPUT = *ccc NOT EXPECTED ON <AVEFF# :T(FAIL)
#
420  FAIL    OUTPUT = SYNERR
421          ASSEM( **, #ERR#, ##, SYNERR )
*
*
*
*
422  END

```

SUCCESSFUL COMPIILATION

STATE SETS AND TRANSITION TABLE

I .1.4 GETKYRD S.LINP KEYWDL

LINE

SENTERCE

STATE PCAP ?

STATE PDATA ?

| | | |
|-------|------------|----|
| STATE | MDATA | 6 |
| STATE | MCAP | 7 |
| STATE | VIEW | 8 |
| STATE | NEWOF | 9 |
| STATE | NEWOR | 10 |
| STATE | KILLDF | 11 |
| STATE | KILLDR | 12 |
| STATE | NEWRLK | 13 |
| STATE | KILLBLK | 14 |
| STATE | COPYBDFILE | 15 |
| STATE | KILLDRJ | 16 |
| STATE | DELOWN | 17 |
| STATE | DELLINK | 18 |
| STATE | NEWKEY | 19 |
| STATE | ADDKEY | 20 |
| STATE | DELKEY | 21 |
| STATE | SENTENCE | 22 |
| FAIL | | |

2 .5 GETTKN 0

SENTENCE PCAP

STATE BLANK 53
FATL

3-8 GETTKN

SENTENCE DATA

STATE BLANK 24
FATE

4 -12 GETTKN 0

SENTENCE NEW

STATE - BLANK - 35
FAIL

5 , 15 GETTKN 0

SENTENCE KILLE

STATE PLANK 56
FATE

6-18 GETTKN 0

SENTENCE-META

STATE BLANK 27

7-22 GETKN 0

SENTENCE MCAR

STATE BLANK 28
FAIL

8 .26 GETTKN 0

SENTENCE VIEW

STATE BLANK 29
FAIL

9 .29 GETTKN 0

SENTENCE NEDF

STATE BLANK 30
FAIL

10 .32 GETTKN 0

SENTENCE NEWDR

STATE BLANK 31
FAIL

11 .37 GETTKN 0

SENTENCE KILLDF

STATE BLANK 32
FAIL

12 .40 GETTKN 0

SENTENCE KILLDR

STATE BLANK 33
FAIL

13 .43 GETTKN 0

SENTENCE NEWRLK

STATE BLANK 34
FAIL

14 .46 GETTKN 0

SENTENCE KILLRLK

STATE BLANK 35
FAIL

15 .49 GETTKN 0

SENTENCE COPYBDFILE

STATE BLANK 36

16 .56 GETTKN 6

SENTENCE KTLLOBJ

STATE BLANK 67

FAIL

17 .59 GETTKN 6

SENTENCE DELOWAI

STATE BLANK 68

FAIL

18 .62 GETTKN 6

SENTENCE DELLINK

STATE BLANK 69

FAIL

19 .65 GETTKN 6

SENTENCE NEKEY

STATE BLANK 69

FAIL

20 .68 GETTKN 6

SENTENCE ADDKEY

STATE BLANK 70

FAIL

21 .73 GETTKN 6

SENTENCE DELKEY

STATE BLANK 70

FAIL

22 .8 SUBGRAM S-PARAN

LINE SENTENCE

STATE CR 43

FAIL

23 .6 SUBGRAM S-PARAN

SENTENCE PCAP BLANK

STATE PARAN 44

FAIL

24 .4 SURGRAM S.PARAM

STATE PARAM 45

FAIL

25 .13 SURGRAM S.PARAM

SENTECE NEWV BLANK

STATE PARAM 46

FAIL

26 .16 SURGRAM S.PARAM

SENTECE KILLV BLANK

STATE PARAM 47

FAIL

27 .19 SURGRAM S.PARAM

SENTECE MOUNTA BLANK

STATE PARAM 48

FAIL

28 .23 SURGRAM S.PARAM

SENTECE MCAR BLANK

STATE PARAM 49

FAIL

29 .27 SURGRAM S.PARAM

SENTECE VIVEN BLANK

STATE PARAM 50

FAIL

30 .30 SURGRAM S.PARAM

SENTECE NEADE BLANK

STATE PARAM 51

FAIL

31 .33 SURGRAM S.PARAM X

SENTECE NEMDR BLANK

STATE PARAM 52

FAIL

32 .38 SURGRAM S.PARAM

SENTENCE KILLED BLANK

STATE PARAM 53

FAIL

33 .41 SURGRAM S.PARAM

SENTENCE KILLED BLANK

STATE PARAM 54

FAIL

34 .44 SURGRAM S.PARAM

SENTENCE NEWBLK BLANK

STATE PARAM 55

FAIL

35 .47 SURGRAM S.PARAM

SENTENCE KILLBLK BLANK

STATE PARAM 56

FAIL

36 .50 GETTKH 0

SENTENCE COPYBFILE BLANK

STATE TIDENT 57

FAIL

37 .57 SURGRAM S.PARAM

SENTENCE KILLEDJ BLANK

STATE PARAM 58

FAIL

38 .60 SURGRAM S.PARAM

SENTENCE DELOWN BLANK

STATE PARAM 59

FAIL

39 .63 SURGRAM S.PARAM

SENTENCE DELLINK BLANK

STATE PARAM 60

FAIL

40 .66 SURGRAM S.PARAM

SENTENCE NEWKEY BLANK

STATE PARAM 41
FAIL

41 .69 SUBGRAM S.PARAM

SENTENCE ADKEY BLANK

STATE PARAM 42
FAIL

42 .74 SUBGRAM S.PARAM

SENTENCE DELKEY BLANK

STATE PARAM 43
FAIL

43 .3

LINE SENTENCE CR

FAIL

44 .7 SUBGRAM S.PARAM

SENTENCE PCAP-BLANK-PARAM

RULE CR 2
FAIL

45 .10 SUBGRAM S.PARAM

SENTENCE PDATA BLANK PARAM

RULE CR 3
STATE PARAM 44
FAIL

46 .14 SUBGRAM S.PARAM

SENTENCE NEWV BLANK PARAM

RULE CR 5
FAIL

47 .17 SUBGRAM S.PARAM

SENTENCE KILLV BLANK PARAM

RULE CR 6
FAIL

48 .20 SUBGRAM S.PARAM

SENTENCE MDATA BLANK PARAM

STATE PARAM 45

49 .24 SURGRAM S.PARAM

SENTENCE MCAP BLANK PARAM

STATE PARAM 64
FAIL

50 .28 SURGRAM S.PARAM

SENTENCE VIEW BLANK PARAM

RULE CR 9
FAIL

51 .31 SURGRAM S.PARAM

SENTENCE NENDF BLANK PARAM

RULE CR 10
FAIL

52 .34 SURGRAM S.PARAM

SENTENCE NENDR BLANK PARAM

STATE PARAM 67
FAIL

53 .39 SURGRAM S.PARAM

SENTENCE KILLDF BLANK PARAM

RULE CR 12
FAIL

54 .42 SURGRAM S.PARAM

SENTENCE KILLDR BLANK PARAM

RULE CR 13
FAIL

55 .45 SURGRAM S.PARAM

SENTENCE NEWBLK BLANK PARAM

RULE CR 14
FAIL

56 .48 SURGRAM S.PARAM

SENTENCE KILLBLK BLANK PARAM

RULE CR 15
FAIL

57 .51 GETTKI 0

SENTENCE COPYBFILE BLANK IDENT

STATE BLANK 68

FAIL

58 .58 SURGRAM S.PARAM

SENTENCE KILLORJ BLANK PARAM

RULE CR 17

FAIL

59 .61 SURGRAM S.PARAM

SENTENCE DFLOWN BLANK PARAM

RULE CR 18

FAIL

60 .64 SURGRAM S.PARAM

SENTENCE DELLINK BLANK PARAM

RULE CR 19

FAIL

61 .67 SURGRAM S.PARAM

SENTENCE NEWKEY BLANK PARAM

RULE CR 20

FAIL

62 .70 SURGRAM S.PARAM

SENTENCE ADJKEY BLANK PARAM

STATE PARAM 49

FAIL

63 .75 SURGRAM S.PARAM

SENTENCE DELKEY BLANK PARAM

STATE PARAM 70

FAIL

64 .11 SURGRAM S.PARAM

SENTENCE PDATA BLANK PARAM PARAM

RULE CR 4

FAIL

65 .21 SURGRAM S.PARAM

SENTENCE MCATA BLANK PARAM PARAM

RULE CR 7
FAIL

66 .25 SURGRAM S.PARAM

SENTENCE MCAP BLANK PARAM PARAM

RULE CR 8
FAIL

67 .35 SURGRAM S.PARAM

SENTENCE NEWDR BLANK PARAM PARAM

STATE PARAM 71
FAIL

68 .52 GETTKN 0

SENTENCE COPYBDFILE BLANK IDENT BLANK

STATE IDENT 72
FAIL

69 .71 SURGRAM S.PARAM

SENTENCE ANDKEY BLANK PARAM PARAM

STATE PARAM 73
FAIL

70 .76 SURGRAM S.PARAM

SENTENCE DELKEY BLANK PARAM PARAM

RULE CR 22
FAIL

71 .36 SURGRAM S.PARAM

SENTENCE NEWDR BLANK PARAM PARAM PARAM

RULE CR 11
FAIL

72 .53 GETTKN 0

SENTENCE COPYBDFILE BLANK IDENT BLANK IDENT

STATE BLANK 74
FAIL

73 .72 SURGRAM S.PARAM

SENTENCE ANDKEY BLANK PARAM PARAM PARAM

FATL

74 .54 SUBGRAM S.PARAM

SENTEICE COPYBDFILE BLANK IDENT BLANK IDENT BLANK

STATE PARAM 75

FATL

75 .55 SUBGRAM S.PARAM

SENTEICE COPYBDFILE BLANK IDENT BLANK IDENT BLANK PARAM

RULE CR 16

FAIL