```
 1: Terminals unused in grammar
 2:
 3:     ROOT
 4:
 5:
 6: Grammar
 7:
 8:      0 $accept: program $end
 9:
10:      1 program: stmtseq
11:
12:      2 stmtseq: stmtseq expr ';'
13:      3          | stmtseq error ';'
14:      4          | stmtseq ';'
15:      5          | %empty
16:
17:      6 expr: expr '=' expr
18:      7       | expr '+' expr
19:      8       | expr '-' expr
20:      9       | expr '*' expr
21:     10       | expr '/' expr
22:     11       | expr '^' expr
23:     12       | '+' expr
24:     13       | '-' expr
25:     14       | '(' expr ')'
26:     15       | IDENT
27:     16       | NUMBER
28:
29:
30: Terminals, with rules where they appear
31:
32: $end (0) 0
33: '(' (40) 14
34: ')' (41) 14
35: '*' (42) 9
36: '+' (43) 7 12
37: '-' (45) 8 13
38: '/' (47) 10
39: ';' (59) 2 3 4
40: '=' (61) 6
41: '^' (94) 11
42: error (256) 3
43: ROOT (258)
44: IDENT (259) 15
45: NUMBER (260) 16
46: POS (261)
47: NEG (262)
48:
49:
50: Nonterminals, with rules where they appear
51:
52: $accept (17)
53:     on left: 0
54: program (18)
55:     on left: 1, on right: 0
56: stmtseq (19)
57:     on left: 2 3 4 5, on right: 1 2 3 4
58: expr (20)
```

```
 59:      on left: 6 7 8 9 10 11 12 13 14 15 16, on right: 2 6 7 8 9 10 11
 60:      12 13 14
 61:
 62:
 63: State 0
 64:
 65:      0 $accept: . program $end
 66:
 67:      $default  reduce using rule 5 (stmtseq)
 68:
 69:      program  go to state 1
 70:      stmtseq  go to state 2
 71:
 72:
 73: State 1
 74:
 75:      0 $accept: program . $end
 76:
 77:      $end  shift, and go to state 3
 78:
 79:
 80: State 2
 81:
 82:      1 program: stmtseq .
 83:      2 stmtseq: stmtseq . expr ';'
 84:      3        | stmtseq . error ';'
 85:      4        | stmtseq . ';'
 86:
 87:      error   shift, and go to state 4
 88:      IDENT   shift, and go to state 5
 89:      NUMBER  shift, and go to state 6
 90:      '+'     shift, and go to state 7
 91:      '-'     shift, and go to state 8
 92:      ';'     shift, and go to state 9
 93:      '('     shift, and go to state 10
 94:
 95:      $end  reduce using rule 1 (program)
 96:
 97:      expr  go to state 11
 98:
 99:
100: State 3
101:
102:      0 $accept: program $end .
103:
104:      $default  accept
105:
106:
107: State 4
108:
109:      3 stmtseq: stmtseq error . ';'
110:
111:      ';'  shift, and go to state 12
112:
113:
114: State 5
115:
116:      15 expr: IDENT .
```

```
117:
118:        $default  reduce using rule 15 (expr)
119:
120:
121: State 6
122:
123:     16 expr: NUMBER .
124:
125:      $default  reduce using rule 16 (expr)
126:
127:
128: State 7
129:
130:     12 expr: '+' . expr
131:
132:      IDENT   shift, and go to state 5
133:      NUMBER  shift, and go to state 6
134:      '+'     shift, and go to state 7
135:      '-'     shift, and go to state 8
136:      '('     shift, and go to state 10
137:
138:      expr  go to state 13
139:
140:
141: State 8
142:
143:     13 expr: '-' . expr
144:
145:      IDENT   shift, and go to state 5
146:      NUMBER  shift, and go to state 6
147:      '+'     shift, and go to state 7
148:      '-'     shift, and go to state 8
149:      '('     shift, and go to state 10
150:
151:      expr  go to state 14
152:
153:
154: State 9
155:
156:      4 stmtseq: stmtseq ';' .
157:
158:      $default  reduce using rule 4 (stmtseq)
159:
160:
161: State 10
162:
163:     14 expr: '(' . expr ')'
164:
165:      IDENT   shift, and go to state 5
166:      NUMBER  shift, and go to state 6
167:      '+'     shift, and go to state 7
168:      '-'     shift, and go to state 8
169:      '('     shift, and go to state 10
170:
171:      expr  go to state 15
172:
173:
174: State 11
```

```
175:
176:       2 stmtseq: stmtseq expr . ';'
177:       6 expr: expr . '=' expr
178:       7     | expr . '+' expr
179:       8     | expr . '−' expr
180:       9     | expr . '*' expr
181:      10     | expr . '/' expr
182:      11     | expr . '^' expr
183:
184:       '='  shift, and go to state 16
185:       '+'  shift, and go to state 17
186:       '−'  shift, and go to state 18
187:       '*'  shift, and go to state 19
188:       '/'  shift, and go to state 20
189:       '^'  shift, and go to state 21
190:       ';'  shift, and go to state 22
191:
192:
193: State 12
194:
195:       3 stmtseq: stmtseq error ';' .
196:
197:       $default  reduce using rule 3 (stmtseq)
198:
199:
200: State 13
201:
202:       6 expr: expr . '=' expr
203:       7     | expr . '+' expr
204:       8     | expr . '−' expr
205:       9     | expr . '*' expr
206:      10     | expr . '/' expr
207:      11     | expr . '^' expr
208:      12     | '+' expr .
209:
210:       $default  reduce using rule 12 (expr)
211:
212:
213: State 14
214:
215:       6 expr: expr . '=' expr
216:       7     | expr . '+' expr
217:       8     | expr . '−' expr
218:       9     | expr . '*' expr
219:      10     | expr . '/' expr
220:      11     | expr . '^' expr
221:      13     | '−' expr .
222:
223:       $default  reduce using rule 13 (expr)
224:
225:
226: State 15
227:
228:       6 expr: expr . '=' expr
229:       7     | expr . '+' expr
230:       8     | expr . '−' expr
231:       9     | expr . '*' expr
232:      10     | expr . '/' expr
```

```
233:    11      | expr . '^' expr
234:    14      | '(' expr . ')'
235:
236:       '='  shift, and go to state 16
237:       '+'  shift, and go to state 17
238:       '-'  shift, and go to state 18
239:       '*'  shift, and go to state 19
240:       '/'  shift, and go to state 20
241:       '^'  shift, and go to state 21
242:       ')'  shift, and go to state 23
243:
244:
245: State 16
246:
247:       6 expr: expr '=' . expr
248:
249:       IDENT   shift, and go to state 5
250:       NUMBER  shift, and go to state 6
251:       '+'     shift, and go to state 7
252:       '-'     shift, and go to state 8
253:       '('     shift, and go to state 10
254:
255:       expr  go to state 24
256:
257:
258: State 17
259:
260:       7 expr: expr '+' . expr
261:
262:       IDENT   shift, and go to state 5
263:       NUMBER  shift, and go to state 6
264:       '+'     shift, and go to state 7
265:       '-'     shift, and go to state 8
266:       '('     shift, and go to state 10
267:
268:       expr  go to state 25
269:
270:
271: State 18
272:
273:       8 expr: expr '-' . expr
274:
275:       IDENT   shift, and go to state 5
276:       NUMBER  shift, and go to state 6
277:       '+'     shift, and go to state 7
278:       '-'     shift, and go to state 8
279:       '('     shift, and go to state 10
280:
281:       expr  go to state 26
282:
283:
284: State 19
285:
286:       9 expr: expr '*' . expr
287:
288:       IDENT   shift, and go to state 5
289:       NUMBER  shift, and go to state 6
290:       '+'     shift, and go to state 7
```

```
291:        '-'     shift, and go to state 8
292:        '('     shift, and go to state 10
293:
294:       expr  go to state 27
295:
296:
297: State 20
298:
299:    10 expr: expr '/' . expr
300:
301:      IDENT   shift, and go to state 5
302:      NUMBER  shift, and go to state 6
303:      '+'     shift, and go to state 7
304:      '-'     shift, and go to state 8
305:      '('     shift, and go to state 10
306:
307:       expr  go to state 28
308:
309:
310: State 21
311:
312:    11 expr: expr '^' . expr
313:
314:      IDENT   shift, and go to state 5
315:      NUMBER  shift, and go to state 6
316:      '+'     shift, and go to state 7
317:      '-'     shift, and go to state 8
318:      '('     shift, and go to state 10
319:
320:       expr  go to state 29
321:
322:
323: State 22
324:
325:     2 stmtseq: stmtseq expr ';' .
326:
327:      $default  reduce using rule 2 (stmtseq)
328:
329:
330: State 23
331:
332:    14 expr: '(' expr ')' .
333:
334:      $default  reduce using rule 14 (expr)
335:
336:
337: State 24
338:
339:     6 expr: expr . '=' expr
340:     6      | expr '=' expr .
341:     7      | expr . '+' expr
342:     8      | expr . '-' expr
343:     9      | expr . '*' expr
344:    10      | expr . '/' expr
345:    11      | expr . '^' expr
346:
347:      '='  shift, and go to state 16
348:      '+'  shift, and go to state 17
```

```
349:        '-'   shift, and go to state 18
350:        '*'   shift, and go to state 19
351:        '/'   shift, and go to state 20
352:        '^'   shift, and go to state 21
353:
354:        $default   reduce using rule 6 (expr)
355:
356:
357: State 25
358:
359:       6 expr: expr . '=' expr
360:       7     | expr . '+' expr
361:       7     | expr '+' expr .
362:       8     | expr . '-' expr
363:       9     | expr . '*' expr
364:      10     | expr . '/' expr
365:      11     | expr . '^' expr
366:
367:        '*'   shift, and go to state 19
368:        '/'   shift, and go to state 20
369:        '^'   shift, and go to state 21
370:
371:        $default   reduce using rule 7 (expr)
372:
373:
374: State 26
375:
376:       6 expr: expr . '=' expr
377:       7     | expr . '+' expr
378:       8     | expr . '-' expr
379:       8     | expr '-' expr .
380:       9     | expr . '*' expr
381:      10     | expr . '/' expr
382:      11     | expr . '^' expr
383:
384:        '*'   shift, and go to state 19
385:        '/'   shift, and go to state 20
386:        '^'   shift, and go to state 21
387:
388:        $default   reduce using rule 8 (expr)
389:
390:
391: State 27
392:
393:       6 expr: expr . '=' expr
394:       7     | expr . '+' expr
395:       8     | expr . '-' expr
396:       9     | expr . '*' expr
397:       9     | expr '*' expr .
398:      10     | expr . '/' expr
399:      11     | expr . '^' expr
400:
401:        '^'   shift, and go to state 21
402:
403:        $default   reduce using rule 9 (expr)
404:
405:
406: State 28
```

```
407:
408:      6 expr: expr . '=' expr
409:      7     | expr . '+' expr
410:      8     | expr . '−' expr
411:      9     | expr . '*' expr
412:     10     | expr . '/' expr
413:     10     | expr '/' expr .
414:     11     | expr . '^' expr
415:
416:      '^'  shift, and go to state 21
417:
418:      $default  reduce using rule 10 (expr)
419:
420:
421: State 29
422:
423:      6 expr: expr . '=' expr
424:      7     | expr . '+' expr
425:      8     | expr . '−' expr
426:      9     | expr . '*' expr
427:     10     | expr . '/' expr
428:     11     | expr . '^' expr
429:     11     | expr '^' expr .
430:
431:      '^'  shift, and go to state 21
432:
433:      $default  reduce using rule 11 (expr)
```

```
1: // $Id: test1.in,v 1.1 2015-07-08 13:29:32-07 - - $
2: a=b*c+d*e;
```

```
 1: ; 0.  1: # 1 "test1.in"; 1.  0:
 2: ; 1.  1: # 1 "<built-in>"
 3: ; 2.  1: # 1 "<command-line>"
 4: ; 3.  1: # 1 "/usr/include/stdc-predef.h" 1 3 4
 5: ; 4.  1: # 1 "<command-line>" 2
 6: ; 5.  1: # 1 "test1.in"
 7: ; 6.  1:
 8: ; 6.  2: a=b*c+d*e;
 9: ; ROOT "<<ROOT>>" (0.0.0)
10: ;     '=' "=" (6.2.2)
11: ;        IDENT "a" (6.2.1)
12: ;        '+' "+" (6.2.6)
13: ;           '*' "*" (6.2.4)
14: ;              IDENT "b" (6.2.3)
15: ;              IDENT "c" (6.2.5)
16: ;           '*' "*" (6.2.8)
17: ;              IDENT "d" (6.2.7)
18: ;              IDENT "e" (6.2.9)
19:
20:          pushvar   b                    ; test1.in 2.3
21:          pushvar   c                    ; test1.in 2.5
22:          mul                            ; test1.in 2.4
23:          pushvar   d                    ; test1.in 2.7
24:          pushvar   e                    ; test1.in 2.9
25:          mul                            ; test1.in 2.8
26:          add                            ; test1.in 2.6
27:          popvar    a                    ; test1.in 2.1
```

```
 1: Command: zexprsm -ly test1.in
 2: -- popen (/usr/bin/cpp test1.in), fileno(yyin) = 4
 3: Starting parse
 4: Entering state 0
 5: Reducing stack by rule 5 (line 46):
 6: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>":)
 7: Stack now 0
 8: Entering state 2
 9: Reading a token: --(end of buffer or a NUL)
10: --accepting rule at line 35 ("# 1 "test1.in"")
11: --included # 1 "test1.in"
12: --accepting rule at line 37 ("
13: ")
14: --accepting rule at line 35 ("# 1 "<built-in>"")
15: --included # 1 "<built-in>"
16: --accepting rule at line 37 ("
17: ")
18: --accepting rule at line 35 ("# 1 "<command-line>"")
19: --included # 1 "<command-line>"
20: --accepting rule at line 37 ("
21: ")
22: --accepting rule at line 35 ("# 1 "/usr/include/stdc-predef.h" 1 3 4")
23: --included # 1 "/usr/include/stdc-predef.h"
24: --accepting rule at line 37 ("
25: ")
26: --accepting rule at line 35 ("# 1 "<command-line>" 2")
27: --included # 1 "<command-line>"
28: --accepting rule at line 37 ("
29: ")
30: --accepting rule at line 35 ("# 1 "test1.in"")
31: --included # 1 "test1.in"
32: --accepting rule at line 37 ("
33: ")
34: --accepting rule at line 37 ("
35: ")
36: --accepting rule at line 40 ("a")
37: Next token is token IDENT (0x5a1c9d0->{IDENT 6.2.1 "a":)
38: Shifting token IDENT (0x5a1c9d0->{IDENT 6.2.1 "a":)
39: Entering state 5
40: Reducing stack by rule 15 (line 58):
41:    $1 = token IDENT (0x5a1c9d0->{IDENT 6.2.1 "a":)
42: -> $$ = nterm expr (0x5a1c9d0->{IDENT 6.2.1 "a":)
43: Stack now 0 2
44: Entering state 11
45: Reading a token: --accepting rule at line 41 ("=")
46: Next token is token '=' (0x5a1cb80->{'=' 6.2.2 "=":)
47: Shifting token '=' (0x5a1cb80->{'=' 6.2.2 "=":)
48: Entering state 16
49: Reading a token: --accepting rule at line 40 ("b")
50: Next token is token IDENT (0x5a1ccc0->{IDENT 6.2.3 "b":)
51: Shifting token IDENT (0x5a1ccc0->{IDENT 6.2.3 "b":)
52: Entering state 5
53: Reducing stack by rule 15 (line 58):
54:    $1 = token IDENT (0x5a1ccc0->{IDENT 6.2.3 "b":)
55: -> $$ = nterm expr (0x5a1ccc0->{IDENT 6.2.3 "b":)
56: Stack now 0 2 11 16
57: Entering state 24
58: Reading a token: --accepting rule at line 44 ("*")
```

```
 59: Next token is token '*' (0x5a1ce00->{'*' 6.2.4 "*":)
 60: Shifting token '*' (0x5a1ce00->{'*' 6.2.4 "*":)
 61: Entering state 19
 62: Reading a token: --accepting rule at line 40 ("c")
 63: Next token is token IDENT (0x5a1cfe0->{IDENT 6.2.5 "c":)
 64: Shifting token IDENT (0x5a1cfe0->{IDENT 6.2.5 "c":)
 65: Entering state 5
 66: Reducing stack by rule 15 (line 58):
 67:    $1 = token IDENT (0x5a1cfe0->{IDENT 6.2.5 "c":)
 68: -> $$ = nterm expr (0x5a1cfe0->{IDENT 6.2.5 "c":)
 69: Stack now 0 2 11 16 24 19
 70: Entering state 27
 71: Reading a token: --accepting rule at line 42 ("+")
 72: Next token is token '+' (0x5a1d120->{'+' 6.2.6 "+":)
 73: Reducing stack by rule 9 (line 52):
 74:    $1 = nterm expr (0x5a1ccc0->{IDENT 6.2.3 "b":)
 75:    $2 = token '*' (0x5a1ce00->{'*' 6.2.4 "*":)
 76:    $3 = nterm expr (0x5a1cfe0->{IDENT 6.2.5 "c":)
 77: -> $$ = nterm expr (0x5a1ce00->{'*' 6.2.4 "*": 0x5a1ccc0 0x5a1cfe0)
 78: Stack now 0 2 11 16
 79: Entering state 24
 80: Next token is token '+' (0x5a1d120->{'+' 6.2.6 "+":)
 81: Shifting token '+' (0x5a1d120->{'+' 6.2.6 "+":)
 82: Entering state 17
 83: Reading a token: --accepting rule at line 40 ("d")
 84: Next token is token IDENT (0x5a1d300->{IDENT 6.2.7 "d":)
 85: Shifting token IDENT (0x5a1d300->{IDENT 6.2.7 "d":)
 86: Entering state 5
 87: Reducing stack by rule 15 (line 58):
 88:    $1 = token IDENT (0x5a1d300->{IDENT 6.2.7 "d":)
 89: -> $$ = nterm expr (0x5a1d300->{IDENT 6.2.7 "d":)
 90: Stack now 0 2 11 16 24 17
 91: Entering state 25
 92: Reading a token: --accepting rule at line 44 ("*")
 93: Next token is token '*' (0x5a1d440->{'*' 6.2.8 "*":)
 94: Shifting token '*' (0x5a1d440->{'*' 6.2.8 "*":)
 95: Entering state 19
 96: Reading a token: --accepting rule at line 40 ("e")
 97: Next token is token IDENT (0x5a1d520->{IDENT 6.2.9 "e":)
 98: Shifting token IDENT (0x5a1d520->{IDENT 6.2.9 "e":)
 99: Entering state 5
100: Reducing stack by rule 15 (line 58):
101:    $1 = token IDENT (0x5a1d520->{IDENT 6.2.9 "e":)
102: -> $$ = nterm expr (0x5a1d520->{IDENT 6.2.9 "e":)
103: Stack now 0 2 11 16 24 17 25 19
104: Entering state 27
105: Reading a token: --accepting rule at line 49 (";")
106: Next token is token ';' (0x5a1d660->{';' 6.2.10 ";":)
107: Reducing stack by rule 9 (line 52):
108:    $1 = nterm expr (0x5a1d300->{IDENT 6.2.7 "d":)
109:    $2 = token '*' (0x5a1d440->{'*' 6.2.8 "*":)
110:    $3 = nterm expr (0x5a1d520->{IDENT 6.2.9 "e":)
111: -> $$ = nterm expr (0x5a1d440->{'*' 6.2.8 "*": 0x5a1d300 0x5a1d520)
112: Stack now 0 2 11 16 24 17
113: Entering state 25
114: Next token is token ';' (0x5a1d660->{';' 6.2.10 ";":)
115: Reducing stack by rule 7 (line 50):
116:    $1 = nterm expr (0x5a1ce00->{'*' 6.2.4 "*": 0x5a1ccc0 0x5a1cfe0)
```

```
117:     $2 = token '+' (0x5a1d120->{'+' 6.2.6 "+":)
118:     $3 = nterm expr (0x5a1d440->{'*' 6.2.8 "*": 0x5a1d300 0x5a1d520)
119: -> $$ = nterm expr (0x5a1d120->{'+' 6.2.6 "+": 0x5a1ce00 0x5a1d440)
120: Stack now 0 2 11 16
121: Entering state 24
122: Next token is token ';' (0x5a1d660->{';' 6.2.10 ";":)
123: Reducing stack by rule 6 (line 49):
124:     $1 = nterm expr (0x5a1c9d0->{IDENT 6.2.1 "a":)
125:     $2 = token '=' (0x5a1cb80->{'=' 6.2.2 "=":)
126:     $3 = nterm expr (0x5a1d120->{'+' 6.2.6 "+": 0x5a1ce00 0x5a1d440)
127: -> $$ = nterm expr (0x5a1cb80->{'=' 6.2.2 "=": 0x5a1c9d0 0x5a1d120)
128: Stack now 0 2
129: Entering state 11
130: Next token is token ';' (0x5a1d660->{';' 6.2.10 ";":)
131: Shifting token ';' (0x5a1d660->{';' 6.2.10 ";":)
132: Entering state 22
133: Reducing stack by rule 2 (line 43):
134:     $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>":)
135:     $2 = nterm expr (0x5a1cb80->{'=' 6.2.2 "=": 0x5a1c9d0 0x5a1d120)
136:     $3 = token ';' (0x5a1d660->{';' 6.2.10 ";":)
137: Deleting astree (0x5a1d660->{';' 6.2.10 ";":)
138: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cb80)
139: Stack now 0
140: Entering state 2
141: Reading a token: --accepting rule at line 37 ("
142: ")
143: --(end of buffer or a NUL)
144: --EOF (start condition 0)
145: Now at end of input.
146: Reducing stack by rule 1 (line 40):
147:     $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cb80)
148: -> $$ = nterm program (nullptr)
149: Stack now 0
150: Entering state 1
151: Now at end of input.
152: Shifting token $end ()
153: Entering state 3
154: Stack now 0 1 3
155: Cleanup: popping token $end ()
156: Cleanup: popping nterm program (nullptr)
157: Dumping parser::root:
158: 0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cb80
159:    0x5a1cb80->{'=' 6.2.2 "=": 0x5a1c9d0 0x5a1d120
160:       0x5a1c9d0->{IDENT 6.2.1 "a":
161:       0x5a1d120->{'+' 6.2.6 "+": 0x5a1ce00 0x5a1d440
162:          0x5a1ce00->{'*' 6.2.4 "*": 0x5a1ccc0 0x5a1cfe0
163:             0x5a1ccc0->{IDENT 6.2.3 "b":
164:             0x5a1cfe0->{IDENT 6.2.5 "c":
165:          0x5a1d440->{'*' 6.2.8 "*": 0x5a1d300 0x5a1d520
166:             0x5a1d300->{IDENT 6.2.7 "d":
167:             0x5a1d520->{IDENT 6.2.9 "e":
168: Dumping string_set:
169: string_set[   0]:   10959529184379665549 0x5a1d0c8->"c"
170:                      3729804957429652673 0x5a18418->"<<ROOT>>"
171: string_set[   1]:    2540012008095083820 0x5a1d608->"e"
172:                     11597697714117577063 0x5a1cc68->"="
173:                     10838281452030117757 0x5a1cda8->"b"
174: string_set[   3]:    13707308581590336685 0x5a1d208->"+"
```

```
175:                         6637313742931709005 0x5a1cee8->"*"
176: string_set[   4]:      5344662657899890615 0x5a1d748->";"
177: string_set[   5]:     14494284460613645429 0x5a1d3e8->"d"
178: string_set[   7]:      4993892634952068459 0x5a1cab8->"a"
179: load_factor = 0.909
180: bucket_count = 11
181: max_bucket_size = 3
182: Deleting astree (0x5a1d520->{IDENT 6.2.9 "e":)
183: Deleting astree (0x5a1d300->{IDENT 6.2.7 "d":)
184: Deleting astree (0x5a1d440->{'*' 6.2.8 "*":)
185: Deleting astree (0x5a1cfe0->{IDENT 6.2.5 "c":)
186: Deleting astree (0x5a1ccc0->{IDENT 6.2.3 "b":)
187: Deleting astree (0x5a1ce00->{'*' 6.2.4 "*":)
188: Deleting astree (0x5a1d120->{'+' 6.2.6 "+":)
189: Deleting astree (0x5a1c9d0->{IDENT 6.2.1 "a":)
190: Deleting astree (0x5a1cb80->{'=' 6.2.2 "=":)
191: Deleting astree (0x5a18320->{ROOT 0.0.0 "<<ROOT>>":)
```

```
   1: ==29489== Memcheck, a memory error detector
   2: ==29489== Copyright (C) 2002-2015, and GNU GPL'd, by Julian Seward et al
.
   3: ==29489== Using Valgrind-3.12.0 and LibVEX; rerun with -h for copyright
info
   4: ==29489== Command: zexprsm -ly test1.in
   5: ==29489== Parent PID: 29488
   6: ==29489==
   7: ==29489==
   8: ==29489== HEAP SUMMARY:
   9: ==29489==     in use at exit: 0 bytes in 0 blocks
  10: ==29489==   total heap usage: 61 allocs, 61 frees, 18,675 bytes allocate
d
  11: ==29489==
  12: ==29489== All heap blocks were freed -- no leaks are possible
  13: ==29489==
  14: ==29489== For counts of detected and suppressed errors, rerun with: -v
  15: ==29489== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
  16: EXIT STATUS = 0
```

```
1: // $Id: test4.in,v 1.2 2015-07-08 13:43:56-07 - - $
2: #include "test4a.inh"
3: #include "test4b.inh"
4: 3*4;
```

```
  1: ; 0.   1: # 1 "test4.in"; 1.   0:
  2: ; 1.   1: # 1 "<built-in>"
  3: ; 2.   1: # 1 "<command-line>"
  4: ; 3.   1: # 1 "/usr/include/stdc-predef.h" 1 3 4
  5: ; 4.   1: # 1 "<command-line>" 2
  6: ; 5.   1: # 1 "test4.in"
  7: ; 6.   1:
  8: ; 6.   2: # 1 "test4a.inh" 1
  9: ; 7.   1:
 10: ; 7.   2:
 11: ; 7.   3: pi=3.14159265358979323846264338380;
 12: ; 7.   4: pi;
 13: ; 7.   5: # 3 "test4.in" 2
 14: ; 8.   3: # 1 "test4b.inh" 1
 15: ; 9.   1:
 16: ; 9.   2: a=pi*r^2;
 17: ; 9.   3: 3.14159265358979323846264338380;
 18: ; 9.   4: # 4 "test4.in" 2
 19: ;10.   4: 3*4;
 20: ; ROOT "<<ROOT>>" (0.0.0)
 21: ;     '=' "=" (7.3.3)
 22: ;         IDENT "pi" (7.3.1)
 23: ;         NUMBER "3.14159265358979323846264338380" (7.3.4)
 24: ;     IDENT "pi" (7.4.1)
 25: ;     '=' "=" (9.2.2)
 26: ;         IDENT "a" (9.2.1)
 27: ;         '*' "*" (9.2.5)
 28: ;             IDENT "pi" (9.2.3)
 29: ;             '^' "^" (9.2.7)
 30: ;                 IDENT "r" (9.2.6)
 31: ;                 NUMBER "2" (9.2.8)
 32: ;     NUMBER "3.14159265358979323846264338380" (9.3.1)
 33: ;     '*' "*" (10.4.2)
 34: ;         NUMBER "3" (10.4.1)
 35: ;         NUMBER "4" (10.4.3)
 36:
 37:         pushnum   3.14159265358979323846264338380; test4a.inh 3.4
 38:         popvar    pi                      ; test4a.inh 3.1
 39:         pushvar   pi                      ; test4a.inh 4.1
 40:         pushvar   pi                      ; test4b.inh 2.3
 41:         pushvar   r                       ; test4b.inh 2.6
 42:         pushnum   2                       ; test4b.inh 2.8
 43:         pow                               ; test4b.inh 2.7
 44:         mul                               ; test4b.inh 2.5
 45:         popvar    a                       ; test4b.inh 2.1
 46:         pushnum   3.14159265358979323846264338380; test4b.inh 3.1
 47:         pushnum   3                       ; test4.in 4.1
 48:         pushnum   4                       ; test4.in 4.3
 49:         mul                               ; test4.in 4.2
```

```
 1: Command: zexprsm -ly test4.in
 2: -- popen (/usr/bin/cpp test4.in), fileno(yyin) = 4
 3: Starting parse
 4: Entering state 0
 5: Reducing stack by rule 5 (line 46):
 6: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>":)
 7: Stack now 0
 8: Entering state 2
 9: Reading a token: --(end of buffer or a NUL)
10: --accepting rule at line 35 ("# 1 "test4.in"")
11: --included # 1 "test4.in"
12: --accepting rule at line 37 ("
13: ")
14: --accepting rule at line 35 ("# 1 "<built-in>"")
15: --included # 1 "<built-in>"
16: --accepting rule at line 37 ("
17: ")
18: --accepting rule at line 35 ("# 1 "<command-line>"")
19: --included # 1 "<command-line>"
20: --accepting rule at line 37 ("
21: ")
22: --accepting rule at line 35 ("# 1 "/usr/include/stdc-predef.h" 1 3 4")
23: --included # 1 "/usr/include/stdc-predef.h"
24: --accepting rule at line 37 ("
25: ")
26: --accepting rule at line 35 ("# 1 "<command-line>" 2")
27: --included # 1 "<command-line>"
28: --accepting rule at line 37 ("
29: ")
30: --accepting rule at line 35 ("# 1 "test4.in"")
31: --included # 1 "test4.in"
32: --accepting rule at line 37 ("
33: ")
34: --accepting rule at line 37 ("
35: ")
36: --accepting rule at line 35 ("# 1 "test4a.inh" 1")
37: --included # 1 "test4a.inh"
38: --accepting rule at line 37 ("
39: ")
40: --accepting rule at line 37 ("
41: ")
42: --accepting rule at line 37 ("
43: ")
44: --accepting rule at line 40 ("pi")
45: Next token is token IDENT (0x5a1ca40->{IDENT 7.3.1 "pi":)
46: Shifting token IDENT (0x5a1ca40->{IDENT 7.3.1 "pi":)
47: Entering state 5
48: Reducing stack by rule 15 (line 58):
49:    $1 = token IDENT (0x5a1ca40->{IDENT 7.3.1 "pi":)
50: -> $$ = nterm expr (0x5a1ca40->{IDENT 7.3.1 "pi":)
51: Stack now 0 2
52: Entering state 11
53: Reading a token: --accepting rule at line 41 ("=")
54: Next token is token '=' (0x5a1cbf0->{'=' 7.3.3 "=":)
55: Shifting token '=' (0x5a1cbf0->{'=' 7.3.3 "=":)
56: Entering state 16
57: Reading a token: --accepting rule at line 39 ("3.14159265358979323846264
3383280")
```

```
 58: Next token is token NUMBER (0x5a1cd30->{NUMBER 7.3.4 "3.1415926535897932
38462643383280":)
 59: Shifting token NUMBER (0x5a1cd30->{NUMBER 7.3.4 "3.14159265358979323846
2643383280":)
 60: Entering state 6
 61: Reducing stack by rule 16 (line 59):
 62:    $1 = token NUMBER (0x5a1cd30->{NUMBER 7.3.4 "3.14159265358979323846
43383280":)
 63: -> $$ = nterm expr (0x5a1cd30->{NUMBER 7.3.4 "3.14159265358979323846264
3383280":)
 64: Stack now 0 2 11 16
 65: Entering state 24
 66: Reading a token: --accepting rule at line 49 (";")
 67: Next token is token ';' (0x5a1ce90->{';' 7.3.36 ";":)
 68: Reducing stack by rule 6 (line 49):
 69:    $1 = nterm expr (0x5a1ca40->{IDENT 7.3.1 "pi":)
 70:    $2 = token '=' (0x5a1cbf0->{'=' 7.3.3 "=":)
 71:    $3 = nterm expr (0x5a1cd30->{NUMBER 7.3.4 "3.14159265358979323846264
3383280":)
 72: -> $$ = nterm expr (0x5a1cbf0->{'=' 7.3.3 "=": 0x5a1ca40 0x5a1cd30)
 73: Stack now 0 2
 74: Entering state 11
 75: Next token is token ';' (0x5a1ce90->{';' 7.3.36 ";":)
 76: Shifting token ';' (0x5a1ce90->{';' 7.3.36 ";":)
 77: Entering state 22
 78: Reducing stack by rule 2 (line 43):
 79:    $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>":)
 80:    $2 = nterm expr (0x5a1cbf0->{'=' 7.3.3 "=": 0x5a1ca40 0x5a1cd30)
 81:    $3 = token ';' (0x5a1ce90->{';' 7.3.36 ";":)
 82: Deleting astree (0x5a1ce90->{';' 7.3.36 ";":)
 83: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0)
 84: Stack now 0
 85: Entering state 2
 86: Reading a token: --accepting rule at line 37 ("
 87: ")
 88: --accepting rule at line 40 ("pi")
 89: Next token is token IDENT (0x5a1d160->{IDENT 7.4.1 "pi":)
 90: Shifting token IDENT (0x5a1d160->{IDENT 7.4.1 "pi":)
 91: Entering state 5
 92: Reducing stack by rule 15 (line 58):
 93:    $1 = token IDENT (0x5a1d160->{IDENT 7.4.1 "pi":)
 94: -> $$ = nterm expr (0x5a1d160->{IDENT 7.4.1 "pi":)
 95: Stack now 0 2
 96: Entering state 11
 97: Reading a token: --accepting rule at line 49 (";")
 98: Next token is token ';' (0x5a1d240->{';' 7.4.3 ";":)
 99: Shifting token ';' (0x5a1d240->{';' 7.4.3 ";":)
100: Entering state 22
101: Reducing stack by rule 2 (line 43):
102:    $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0)
103:    $2 = nterm expr (0x5a1d160->{IDENT 7.4.1 "pi":)
104:    $3 = token ';' (0x5a1d240->{';' 7.4.3 ";":)
105: Deleting astree (0x5a1d240->{';' 7.4.3 ";":)
106: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160)
107: Stack now 0
108: Entering state 2
109: Reading a token: --accepting rule at line 37 ("
```

```
110: ")
111: --accepting rule at line 35 ("# 3 "test4.in" 2")
112: --included # 3 "test4.in"
113: --accepting rule at line 37 ("
114: ")
115: --accepting rule at line 35 ("# 1 "test4b.inh" 1")
116: --included # 1 "test4b.inh"
117: --accepting rule at line 37 ("
118: ")
119: --accepting rule at line 37 ("
120: ")
121: --accepting rule at line 40 ("a")
122: Next token is token IDENT (0x5a1d510->{IDENT 9.2.1 "a":)
123: Shifting token IDENT (0x5a1d510->{IDENT 9.2.1 "a":)
124: Entering state 5
125: Reducing stack by rule 15 (line 58):
126:    $1 = token IDENT (0x5a1d510->{IDENT 9.2.1 "a":)
127: -> $$ = nterm expr (0x5a1d510->{IDENT 9.2.1 "a":)
128: Stack now 0 2
129: Entering state 11
130: Reading a token: --accepting rule at line 41 ("=")
131: Next token is token '=' (0x5a1d650->{'=' 9.2.2 "=":)
132: Shifting token '=' (0x5a1d650->{'=' 9.2.2 "=":)
133: Entering state 16
134: Reading a token: --accepting rule at line 40 ("pi")
135: Next token is token IDENT (0x5a1d730->{IDENT 9.2.3 "pi":)
136: Shifting token IDENT (0x5a1d730->{IDENT 9.2.3 "pi":)
137: Entering state 5
138: Reducing stack by rule 15 (line 58):
139:    $1 = token IDENT (0x5a1d730->{IDENT 9.2.3 "pi":)
140: -> $$ = nterm expr (0x5a1d730->{IDENT 9.2.3 "pi":)
141: Stack now 0 2 11 16
142: Entering state 24
143: Reading a token: --accepting rule at line 44 ("*")
144: Next token is token '*' (0x5a1d810->{'*' 9.2.5 "*":)
145: Shifting token '*' (0x5a1d810->{'*' 9.2.5 "*":)
146: Entering state 19
147: Reading a token: --accepting rule at line 40 ("r")
148: Next token is token IDENT (0x5a1d950->{IDENT 9.2.6 "r":)
149: Shifting token IDENT (0x5a1d950->{IDENT 9.2.6 "r":)
150: Entering state 5
151: Reducing stack by rule 15 (line 58):
152:    $1 = token IDENT (0x5a1d950->{IDENT 9.2.6 "r":)
153: -> $$ = nterm expr (0x5a1d950->{IDENT 9.2.6 "r":)
154: Stack now 0 2 11 16 24 19
155: Entering state 27
156: Reading a token: --accepting rule at line 46 ("^")
157: Next token is token '^' (0x5a1da90->{'^' 9.2.7 "^":)
158: Shifting token '^' (0x5a1da90->{'^' 9.2.7 "^":)
159: Entering state 21
160: Reading a token: --accepting rule at line 39 ("2")
161: Next token is token NUMBER (0x5a1dbd0->{NUMBER 9.2.8 "2":)
162: Shifting token NUMBER (0x5a1dbd0->{NUMBER 9.2.8 "2":)
163: Entering state 6
164: Reducing stack by rule 16 (line 59):
165:    $1 = token NUMBER (0x5a1dbd0->{NUMBER 9.2.8 "2":)
166: -> $$ = nterm expr (0x5a1dbd0->{NUMBER 9.2.8 "2":)
167: Stack now 0 2 11 16 24 19 27 21
```

```
168: Entering state 29
169: Reading a token: --accepting rule at line 49 (";")
170: Next token is token ';' (0x5a1dd10->{';' 9.2.9 ";":)
171: Reducing stack by rule 11 (line 54):
172:    $1 = nterm expr (0x5a1d950->{IDENT 9.2.6 "r":)
173:    $2 = token '^' (0x5a1da90->{'^' 9.2.7 "^":)
174:    $3 = nterm expr (0x5a1dbd0->{NUMBER 9.2.8 "2":)
175: -> $$ = nterm expr (0x5a1da90->{'^' 9.2.7 "^": 0x5a1d950 0x5a1dbd0)
176: Stack now 0 2 11 16 24 19
177: Entering state 27
178: Next token is token ';' (0x5a1dd10->{';' 9.2.9 ";":)
179: Reducing stack by rule 9 (line 52):
180:    $1 = nterm expr (0x5a1d730->{IDENT 9.2.3 "pi":)
181:    $2 = token '*' (0x5a1d810->{'*' 9.2.5 "*":)
182:    $3 = nterm expr (0x5a1da90->{'^' 9.2.7 "^": 0x5a1d950 0x5a1dbd0)
183: -> $$ = nterm expr (0x5a1d810->{'*' 9.2.5 "*": 0x5a1d730 0x5a1da90)
184: Stack now 0 2 11 16
185: Entering state 24
186: Next token is token ';' (0x5a1dd10->{';' 9.2.9 ";":)
187: Reducing stack by rule 6 (line 49):
188:    $1 = nterm expr (0x5a1d510->{IDENT 9.2.1 "a":)
189:    $2 = token '=' (0x5a1d650->{'=' 9.2.2 "=":)
190:    $3 = nterm expr (0x5a1d810->{'*' 9.2.5 "*": 0x5a1d730 0x5a1da90)
191: -> $$ = nterm expr (0x5a1d650->{'=' 9.2.2 "=": 0x5a1d510 0x5a1d810)
192: Stack now 0 2
193: Entering state 11
194: Next token is token ';' (0x5a1dd10->{';' 9.2.9 ";":)
195: Shifting token ';' (0x5a1dd10->{';' 9.2.9 ";":)
196: Entering state 22
197: Reducing stack by rule 2 (line 43):
198:    $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160)
199:    $2 = nterm expr (0x5a1d650->{'=' 9.2.2 "=": 0x5a1d510 0x5a1d810)
200:    $3 = token ';' (0x5a1dd10->{';' 9.2.9 ";":)
201: Deleting astree (0x5a1dd10->{';' 9.2.9 ";":)
202: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160 0x5a1d650)
203: Stack now 0
204: Entering state 2
205: Reading a token: --accepting rule at line 37 ("
206: ")
207: --accepting rule at line 39 ("3.14159265358979323846264338 3280")
208: Next token is token NUMBER (0x5a1e030->{NUMBER 9.3.1 "3.1415926535897932
38462643383280":)
209: Shifting token NUMBER (0x5a1e030->{NUMBER 9.3.1 "3.14159265358979323846 2
643383280":)
210: Entering state 6
211: Reducing stack by rule 16 (line 59):
212:    $1 = token NUMBER (0x5a1e030->{NUMBER 9.3.1 "3.1415926535897932384626
43383280":)
213: -> $$ = nterm expr (0x5a1e030->{NUMBER 9.3.1 "3.1415926535897932384626 43
383280":)
214: Stack now 0 2
215: Entering state 11
216: Reading a token: --accepting rule at line 49 (";")
217: Next token is token ';' (0x5a1e130->{';' 9.3.33 ";":)
218: Shifting token ';' (0x5a1e130->{';' 9.3.33 ";":)
219: Entering state 22
```

```
220: Reducing stack by rule 2 (line 43):
221:    $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160 0x5a1d650)
222:    $2 = nterm expr (0x5a1e030->{NUMBER 9.3.1 "3.14159265358979323846264
383280":)
223:    $3 = token ';' (0x5a1e130->{';' 9.3.33 ";":)
224: Deleting astree (0x5a1e130->{';' 9.3.33 ";":)
225: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160 0x5a1d650 0x5a1e030)
226: Stack now 0
227: Entering state 2
228: Reading a token: --accepting rule at line 37 ("
229: ")
230: --accepting rule at line 35 ("# 4 "test4.in" 2")
231: --included # 4 "test4.in"
232: --accepting rule at line 37 ("
233: ")
234: --accepting rule at line 39 ("3")
235: Next token is token NUMBER (0x5a1e280->{NUMBER 10.4.1 "3":)
236: Shifting token NUMBER (0x5a1e280->{NUMBER 10.4.1 "3":)
237: Entering state 6
238: Reducing stack by rule 16 (line 59):
239:    $1 = token NUMBER (0x5a1e280->{NUMBER 10.4.1 "3":)
240: -> $$ = nterm expr (0x5a1e280->{NUMBER 10.4.1 "3":)
241: Stack now 0 2
242: Entering state 11
243: Reading a token: --accepting rule at line 44 ("*")
244: Next token is token '*' (0x5a1e4c0->{'*' 10.4.2 "*":)
245: Shifting token '*' (0x5a1e4c0->{'*' 10.4.2 "*":)
246: Entering state 19
247: Reading a token: --accepting rule at line 39 ("4")
248: Next token is token NUMBER (0x5a1e5a0->{NUMBER 10.4.3 "4":)
249: Shifting token NUMBER (0x5a1e5a0->{NUMBER 10.4.3 "4":)
250: Entering state 6
251: Reducing stack by rule 16 (line 59):
252:    $1 = token NUMBER (0x5a1e5a0->{NUMBER 10.4.3 "4":)
253: -> $$ = nterm expr (0x5a1e5a0->{NUMBER 10.4.3 "4":)
254: Stack now 0 2 11 19
255: Entering state 27
256: Reading a token: --accepting rule at line 49 (";")
257: Next token is token ';' (0x5a1e6e0->{';' 10.4.4 ";":)
258: Reducing stack by rule 9 (line 52):
259:    $1 = nterm expr (0x5a1e280->{NUMBER 10.4.1 "3":)
260:    $2 = token '*' (0x5a1e4c0->{'*' 10.4.2 "*":)
261:    $3 = nterm expr (0x5a1e5a0->{NUMBER 10.4.3 "4":)
262: -> $$ = nterm expr (0x5a1e4c0->{'*' 10.4.2 "*": 0x5a1e280 0x5a1e5a0)
263: Stack now 0 2
264: Entering state 11
265: Next token is token ';' (0x5a1e6e0->{';' 10.4.4 ";":)
266: Shifting token ';' (0x5a1e6e0->{';' 10.4.4 ";":)
267: Entering state 22
268: Reducing stack by rule 2 (line 43):
269:    $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160 0x5a1d650 0x5a1e030)
270:    $2 = nterm expr (0x5a1e4c0->{'*' 10.4.2 "*": 0x5a1e280 0x5a1e5a0)
271:    $3 = token ';' (0x5a1e6e0->{';' 10.4.4 ";":)
272: Deleting astree (0x5a1e6e0->{';' 10.4.4 ";":)
273: -> $$ = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
```

```
1d160 0x5a1d650 0x5a1e030 0x5a1e4c0)
  274: Stack now 0
  275: Entering state 2
  276: Reading a token: --accepting rule at line 37 ("
  277: ")
  278: --(end of buffer or a NUL)
  279: --EOF (start condition 0)
  280: Now at end of input.
  281: Reducing stack by rule 1 (line 40):
  282:    $1 = nterm stmtseq (0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a
1d160 0x5a1d650 0x5a1e030 0x5a1e4c0)
  283: -> $$ = nterm program (nullptr)
  284: Stack now 0
  285: Entering state 1
  286: Now at end of input.
  287: Shifting token $end ()
  288: Entering state 3
  289: Stack now 0 1 3
  290: Cleanup: popping token $end ()
  291: Cleanup: popping nterm program (nullptr)
  292: Dumping parser::root:
  293: 0x5a18320->{ROOT 0.0.0 "<<ROOT>>": 0x5a1cbf0 0x5a1d160 0x5a1d650 0x5a1e0
30 0x5a1e4c0
  294:    0x5a1cbf0->{'=' 7.3.3 "=": 0x5a1ca40 0x5a1cd30
  295:      0x5a1ca40->{IDENT 7.3.1 "pi":
  296:      0x5a1cd30->{NUMBER 7.3.4 "3.14159265358979323846263383280":
  297:    0x5a1d160->{IDENT 7.4.1 "pi":
  298:    0x5a1d650->{'=' 9.2.2 "=": 0x5a1d510 0x5a1d810
  299:      0x5a1d510->{IDENT 9.2.1 "a":
  300:      0x5a1d810->{'*' 9.2.5 "*": 0x5a1d730 0x5a1da90
  301:        0x5a1d730->{IDENT 9.2.3 "pi":
  302:        0x5a1da90->{'^' 9.2.7 "^": 0x5a1d950 0x5a1dbd0
  303:          0x5a1d950->{IDENT 9.2.6 "r":
  304:          0x5a1dbd0->{NUMBER 9.2.8 "2":
  305:    0x5a1e030->{NUMBER 9.3.1 "3.14159265358979323846263383280":
  306:    0x5a1e4c0->{'*' 10.4.2 "*": 0x5a1e280 0x5a1e5a0
  307:      0x5a1e280->{NUMBER 10.4.1 "3":
  308:      0x5a1e5a0->{NUMBER 10.4.3 "4":
  309: Dumping string_set:
  310: string_set[  3]:    8248777770799913213 0x5a1e368->"3"
  311:                     6637313742931709005 0x5a1d8f8->"*"
  312:                    11597697714117577063 0x5a1ccd8->"="
  313:                     5344662657899890615 0x5a1cf78->";"
  314: string_set[  6]:    3729804957429652673 0x5a18418->"<<ROOT>>"
  315: string_set[  9]:    4551451650890805270 0x5a1dcb8->"2"
  316: string_set[ 11]:    1998636459596678225 0x5a1db78->"^"
  317: string_set[ 12]:   15530967369654070964 0x5a1cb28->"pi"
  318:                    14518329498511883088 0x5a1da38->"r"
  319: string_set[ 17]:    4993892634952068459 0x5a1d5f8->"a"
  320: string_set[ 20]:   17027087115628340017 0x5a1ce38->"3.14159265358979323
8462643383280"
  321: string_set[ 21]:   16215888864653804456 0x5a1e688->"4"
  322: load_factor = 0.522
  323: bucket_count = 23
  324: max_bucket_size = 4
  325: Deleting astree (0x5a1e5a0->{NUMBER 10.4.3 "4":)
  326: Deleting astree (0x5a1e280->{NUMBER 10.4.1 "3":)
  327: Deleting astree (0x5a1e4c0->{'*' 10.4.2 "*":)
```

```
  328: Deleting astree (0x5a1e030->{NUMBER 9.3.1 "3.14159265358979323846264383
280":)
  329: Deleting astree (0x5a1dbd0->{NUMBER 9.2.8 "2":)
  330: Deleting astree (0x5a1d950->{IDENT 9.2.6 "r":)
  331: Deleting astree (0x5a1da90->{'^' 9.2.7 "^":)
  332: Deleting astree (0x5a1d730->{IDENT 9.2.3 "pi":)
  333: Deleting astree (0x5a1d810->{'*' 9.2.5 "*":)
  334: Deleting astree (0x5a1d510->{IDENT 9.2.1 "a":)
  335: Deleting astree (0x5a1d650->{'=' 9.2.2 "=":)
  336: Deleting astree (0x5a1d160->{IDENT 7.4.1 "pi":)
  337: Deleting astree (0x5a1cd30->{NUMBER 7.3.4 "3.14159265358979323846264383
280":)
  338: Deleting astree (0x5a1ca40->{IDENT 7.3.1 "pi":)
  339: Deleting astree (0x5a1cbf0->{'=' 7.3.3 "=":)
  340: Deleting astree (0x5a18320->{ROOT 0.0.0 "<<ROOT>>":)
```

```
 1: ==29569== Memcheck, a memory error detector
 2: ==29569== Copyright (C) 2002-2015, and GNU GPL'd, by Julian Seward et al
.
 3: ==29569== Using Valgrind-3.12.0 and LibVEX; rerun with -h for copyright
info
 4: ==29569== Command: zexprsm -ly test4.in
 5: ==29569== Parent PID: 29568
 6: ==29569==
 7: ==29569==
 8: ==29569== HEAP SUMMARY:
 9: ==29569==     in use at exit: 0 bytes in 0 blocks
10: ==29569==   total heap usage: 94 allocs, 94 frees, 20,272 bytes allocate
d
11: ==29569==
12: ==29569== All heap blocks were freed -- no leaks are possible
13: ==29569==
14: ==29569== For counts of detected and suppressed errors, rerun with: -v
15: ==29569== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
16: EXIT STATUS = 0
```