

测试报告

测试一

测试样例一:

```
int a;  
void func(){  
    a=10;  
    return 0;  
}
```

词法分析器输出结果:

```
int KW 1  
a Ident -1  
; SE 24  
void KW 2  
func Ident -1  
( SE 20  
) SE 21  
{ SE 22  
a Ident -1  
= OP 11  
10 INT -1  
; SE 24  
return KW 3  
0 INT -1  
; SE 24  
} SE 23
```

语法分析器输出结果:

```
1  program#int reduction
2  compUnit#int    reduction
3  decl#int        reduction
4  varDecl#int     reduction
5  bType#int       reduction
6  int#int move
7  varDef#Ident    reduction
8  Ident#Ident move
9  argVarDef#; reduction
10 argVarDecl#;    reduction
11 ;#; move
12 compUnit#void    reduction
13 funcDef#void     reduction
14 funcType#void    reduction
15 void#void move
16 Ident#Ident move
17 (#( move
18 funcFParams#)    reduction
19 )#) move
20 block#{ reduction
21 {#{ move
22 blockItem#Ident  reduction
23 stmt#Ident       reduction
24 exp#Ident        reduction
25 assignExp#Ident  reduction
26 eqExp#Ident      reduction
27 relExp#Ident     reduction
28 addExp#Ident     reduction
29 mulExp#Ident     reduction
30 unaryExp#Ident   reduction
31 Ident#Ident move
32 callFunc#=       reduction
```

32 case 0: return 0;

33 mulExpAtom# = reduction

34 addExpAtom# = reduction

```
35  relExpAtom#=      reduction
36  eqExpAtom#= reduction
37  assignExpAtom#= reduction
38  =#= move
39  eqExp#INT    reduction
40  relExp#INT    reduction
41  addExp#INT    reduction
42  mulExp#INT    reduction
43  unaryExp#INT    reduction
44  primaryExp#INT reduction
45  number#INT    reduction
46  INT#INT move
47  mulExpAtom#;    reduction
48  addExpAtom#;    reduction
49  relExpAtom#;    reduction
50  eqExpAtom#; reduction
51  assignExpAtom#; reduction
52  ;#; move
53  blockItem#return    reduction
54  stmt#return reduction
55  return#return    move
56  argExp#INT    reduction
57  exp#INT    reduction
58  assignExp#INT    reduction
59  eqExp#INT    reduction
60  relExp#INT    reduction
61  addExp#INT    reduction
62  mulExp#INT    reduction
63  unaryExp#INT    reduction
64  primaryExp#INT reduction
65  number#INT    reduction
66  INT#INT move
```

```
67 mulExpAtom#;    reduction
68 addExpAtom#;    reduction
69 relExpAtom#;    reduction
70 eqExpAtom#; reduction
71 assignExpAtom#; reduction
72 ;#; move
73 blockItem#} reduction
74 }#} move
75 EOF#EOF accept
```

测试二

测试样例二:

```
int a = 3;
int b = 5;

void func(){
    return a + b;
}
```

词法分析器输出结果:

```
int KW 1
a Ident -1
= OP 11
3 INT -1
; SE 24
int KW 1
b Ident -1
= OP 11
5 INT -1
; SE 24
void KW 2
func Ident -1
( SE 20
) SE 21
{ SE 22
return KW 3
a Ident -1
+ OP 6
b Ident -1
; SE 24
} SE 23
```

语法分析器输出结果:

1	program#int	reduction
2	compUnit#int	reduction
3	decl#int	reduction
4	varDecl#int	reduction
5	bType#int	reduction
6	int#int	move
7	varDef#Ident	reduction
8	Ident#Ident	move
9	argVarDef#	= reduction
10	=#	= move
11	initVal#INT	reduction
12	exp#INT	reduction
13	assignExp#INT	reduction
14	eqExp#INT	reduction
15	relExp#INT	reduction
16	addExp#INT	reduction
17	mulExp#INT	reduction
18	unaryExp#INT	reduction
19	primaryExp#INT	reduction
20	number#INT	reduction
21	INT#INT	move
22	mulExpAtom#	; reduction
23	addExpAtom#	; reduction
24	relExpAtom#	; reduction
25	eqExpAtom#	; reduction
26	assignExpAtom#	; reduction
27	argVarDecl#	; reduction
28	;	#; move
29	compUnit#int	reduction
30	decl#int	reduction
31	varDecl#int	reduction

32	bType#int	reduction
33	int#int	move
34	varDef#Ident	reduction
35	Ident#Ident	move
36	argVarDef#	= reduction
37	=#	= move
38	initVal#INT	reduction
39	exp#INT	reduction
40	assignExp#INT	reduction
41	eqExp#INT	reduction
42	relExp#INT	reduction
43	addExp#INT	reduction
44	mulExp#INT	reduction
45	unaryExp#INT	reduction
46	primaryExp#INT	reduction
47	number#INT	reduction
48	INT#INT	move
49	mulExpAtom#	; reduction
50	addExpAtom#	; reduction
51	relExpAtom#	; reduction
52	eqExpAtom#	; reduction
53	assignExpAtom#	; reduction
54	argVarDecl#	; reduction
55	;	#; move
56	compUnit#void	reduction
57	funcDef#void	reduction
58	funcType#void	reduction
59	void#void	move
60	Ident#Ident	move
61	(#	(move
62	funcFParams#) reduction


```
63  )#) move
64  block#{ reduction
65  #{# { move
66  blockItem#return    reduct
67  stmt#return reduction
68  return#return    move
69  argExp#Ident    reduction
70  exp#Ident    reduction
71  assignExp#Ident reduction
72  eqExp#Ident reduction
73  relExp#Ident    reduction
74  addExp#Ident    reduction
75  mulExp#Ident    reduction
76  unaryExp#Ident  reduction
77  Ident#Ident move
78  callFunc#+    reduction
79  mulExpAtom#+    reduction
80  addExpAtom#+    reduction
81  +#+ move
82  mulExp#Ident    reduction
83  unaryExp#Ident  reduction
84  Ident#Ident move
85  callFunc#;    reduction
86  mulExpAtom#;    reduction
87  addExpAtom#;    reduction
88  relExpAtom#;    reduction
89  eqExpAtom#;    reduction
90  assignExpAtom#; reduction
91  ;#; move
```

```
92  blockItem#} reduction
93  }#} move
94  EOF#EOF accept
```

测试三

测试样例三:

```
int a = 3;
int b = 5;

void func(){
    int a = 5;
    return a + b;
}
```

词法分析器输出:

```
int KW 1
a Ident -1
= OP 11
3 INT -1
; SE 24
int KW 1
b Ident -1
= OP 11
5 INT -1
; SE 24
void KW 2
func Ident -1
( SE 20
) SE 21
{ SE 22
int KW 1
a Ident -1
= OP 11
5 INT -1
; SE 24
return KW 3
a Ident -1
+ OP 6
b Ident -1
; SE 24
} SE 23
```

语法分析器输出：

```
1  program#int reduction
2  compUnit#int    reduction
3  decl#int        reduction
4  varDecl#int     reduction
5  bType#int       reduction
6  int#int move
7  varDef#Ident    reduction
8  Ident#Ident move
9  argVarDef#= reduction
10 =#= move
11 initVal#INT reduction
12 exp#INT reduction
13 assignExp#INT reduction
14 eqExp#INT reduction
15 relExp#INT reduction
16 addExp#INT reduction
17 mulExp#INT reduction
18 unaryExp#INT reduction
19 primaryExp#INT reduction
20 number#INT reduction
21 INT#INT move
22 mulExpAtom#; reduction
23 addExpAtom#; reduction
24 relExpAtom#; reduction
25 eqExpAtom#; reduction
26 assignExpAtom#; reduction
27 argVarDecl#; reduction
28 ;#; move
29 compUnit#int reduction
30 decl#int reduction
31 varDecl#int reduction
```

```
32  bType#int    reduction
33  int#int move
34  varDef#Ident    reduction
35  Ident#Ident move
36  argVarDef#= reduction
37  =#= move
38  initVal#INT reduction
39  exp#INT reduction
40  assignExp#INT    reduction
41  eqExp#INT    reduction
42  relExp#INT    reduction
43  addExp#INT    reduction
44  mulExp#INT    reduction
45  unaryExp#INT    reduction
46  primaryExp#INT    reduction
47  number#INT    reduction
48  INT#INT move
49  mulExpAtom#;    reduction
50  addExpAtom#;    reduction
51  relExpAtom#;    reduction
52  eqExpAtom#; reduction
53  assignExpAtom#; reduction
54  argVarDecl#;    reduction
55  ;#; move
56  compUnit#void    reduction
57  funcDef#void    reduction
58  funcType#void    reduction
59  void#void    move
60  Ident#Ident move
61  (#( move
62  funcFParams#)    reduction
```

```
63  )#) move
64  block#{ reduction
65  #{# move
66  blockItem#int    reduction
67  decl#int    reduction
68  varDecl#int reduction
69  bType#int    reduction
70  int#int move
71  varDef#Ident    reduction
72  Ident#Ident move
73  argVarDef#= reduction
74  =#= move
75  initVal#INT reduction
76  exp#INT reduction
77  assignExp#INT    reduction
78  eqExp#INT    reduction
79  relExp#INT    reduction
80  addExp#INT    reduction
81  mulExp#INT    reduction
82  unaryExp#INT    reduction
83  primaryExp#INT  reduction
84  number#INT    reduction
85  INT#INT move
86  mulExpAtom#;    reduction
87  addExpAtom#;    reduction
88  relExpAtom#;    reduction
89  eqExpAtom#; reduction
90  assignExpAtom#; reduction
91  argVarDecl#;    reduction
92  ;#; move
```

```
93  blockItem#return    reduction
94  stmt#return reduction
95  return#return    move
96  argExp#Ident    reduction
97  exp#Ident    reduction
98  assignExp#Ident reduction
99  eqExp#Ident reduction
100 relExp#Ident    reduction
101 addExp#Ident    reduction
102 mulExp#Ident    reduction
103 unaryExp#Ident  reduction
104 Ident#Ident move
105 callFunc#+    reduction
106 mulExpAtom#+    reduction
107 addExpAtom#+    reduction
108 +#+ move
109 mulExp#Ident    reduction
110 unaryExp#Ident  reduction
111 Ident#Ident move
112 callFunc#;    reduction
113 mulExpAtom#;    reduction
114 addExpAtom#;    reduction
115 relExpAtom#;    reduction
116 eqExpAtom#;    reduction
117 assignExpAtom#; reduction
118 ;#; move
119 blockItem#}    reduction
120 }#} move
121 EOF#EOF accept
```

测试四

测试样例四:

```
int a = 3, b = 1, c = 3;
```

```
void func(){  
    int a = 5;  
    return a + b;  
}
```

词法分析器输出:


```

int KW 1
a Ident -1
= OP 11
3 INT -1
, SE 25
b Ident -1
= OP 11
1 INT -1
, SE 25
c Ident -1
= OP 11
3 INT -1
; SE 24
void KW 2
func Ident -1
( SE 20
) SE 21
{ SE 22
int KW 1
a Ident -1
= OP 11
5 INT -1
; SE 24
return KW 3
a Ident -1
+ OP 6
b Ident -1
; SE 24
} SE 23

```

语法分析器输出:

```
1  program#int reduction
2  compUnit#int    reduction
3  decl#int        reduction
4  varDecl#int     reduction
5  bType#int       reduction
6  int#int move
7  varDef#Ident    reduction
8  Ident#Ident move
9  argVarDef#= reduction
10 =#= move
11 initVal#INT reduction
12 exp#INT reduction
13 assignExp#INT   reduction
14 eqExp#INT       reduction
15 relExp#INT      reduction
16 addExp#INT      reduction
17 mulExp#INT      reduction
18 unaryExp#INT    reduction
19 primaryExp#INT  reduction
20 number#INT      reduction
21 INT#INT move
22 mulExpAtom#,    reduction
23 addExpAtom#,    reduction
24 relExpAtom#,    reduction
25 eqExpAtom#,     reduction
26 assignExpAtom#, reduction
27 argVarDecl#,    reduction
28 ,#, move
29 varDef#Ident    reduction
30 Ident#Ident move
31 argVarDef#= reduction
32 =#= move
```



```
33  initVal#INT reduction
34  exp#INT reduction
35  assignExp#INT  reduction
36  eqExp#INT  reduction
37  relExp#INT  reduction
38  addExp#INT  reduction
39  mulExp#INT  reduction
40  unaryExp#INT  reduction
41  primaryExp#INT reduction
42  number#INT  reduction
43  INT#INT move
44  mulExpAtom#,  reduction
45  addExpAtom#,  reduction
46  relExpAtom#,  reduction
47  eqExpAtom#, reduction
48  assignExpAtom#, reduction
49  argVarDecl#,  reduction
50  ,#, move
51  varDef#Ident  reduction
52  Ident#Ident move
53  argVarDef#= reduction
54  =#= move
55  initVal#INT reduction
56  exp#INT reduction
57  assignExp#INT  reduction
58  eqExp#INT  reduction
59  relExp#INT  reduction
60  addExp#INT  reduction
61  mulExp#INT  reduction
62  unaryExp#INT  reduction
63  primaryExp#INT reduction
64  number#INT  reduction
```

— — — — —

```
65  INT#INT move
```

```
66 mulExpAtom#;    reduction
67 addExpAtom#;    reduction
68 relExpAtom#;    reduction
69 eqExpAtom#; reduction
70 assignExpAtom#; reduction
71 argVarDecl#;    reduction
72 ;#; move
73 compUnit#void   reduction
74 funcDef#void    reduction
75 funcType#void   reduction
76 void#void      move
77 Ident#Ident     move
78 (#( move
79 funcFParams#)   reduction
80 )#) move
81 block#{         reduction
82 {#{ move
83 blockItem#int   reduction
84 decl#int        reduction
85 varDecl#int     reduction
86 bType#int       reduction
87 int#int         move
88 varDef#Ident    reduction
89 Ident#Ident     move
90 argVarDef#      reduction
91 =#              move
92 initVal#INT     reduction
93 exp#INT          reduction
94 assignExp#INT   reduction
95 eqExp#INT       reduction
96 relExp#INT      reduction
97 addExp#INT      reduction
```

```
77 mulExp#INT reduction
98 mulExp#INT reduction
```

```

99  unaryExp#INT      reduction
100 primaryExp#INT    reduction
101 number#INT         reduction
102 INT#INT            move
103 mulExpAtom#;       reduction
104 addExpAtom#;       reduction
105 relExpAtom#;       reduction
106 eqExpAtom#;        reduction
107 assignExpAtom#;    reduction
108 argVarDecl#;       reduction
109 ;#;                move
110 blockItem#return    reduction
111 stmt#return         reduction
112 return#return       move
113 argExp#Ident        reduction
114 exp#Ident           reduction
115 assignExp#Ident     reduction
116 eqExp#Ident         reduction
117 relExp#Ident        reduction
118 addExp#Ident        reduction
119 mulExp#Ident        reduction
120 unaryExp#Ident     reduction
121 Ident#Ident        move
122 callFunc#+         reduction
123 mulExpAtom#+       reduction
124 addExpAtom#+       reduction
125 +#+               move
126 mulExp#Ident       reduction
127 unaryExp#Ident     reduction
128 Ident#Ident        move
129 callFunc#;         reduction
130 mulExpAtom#*       reduction

```



```

130 mulExpAtom#;    reduction
131 addExpAtom#;    reduction

132 relExpAtom#;    reduction
133 eqExpAtom#; reduction
134 assignExpAtom#; reduction
135 ;#; move
136 blockItem#} reduction
137 }#} move
138 EOF#EOF accept

```

测试五

测试样例五:

```

int a = 3;
int b = 1;
int c = 3;
int e = 0, f = 10;

void func(){
    int a = 5;
    int c = 2;
    int temp1 = f + e;
    int temp2 = f + 2;
    int temp3 = c;
    return a + b;
}

```

词法分析器输出:

```
int KW 1
a Ident -1
= OP 11
3 INT -1
; SE 24
int KW 1
b Ident -1
= OP 11
1 INT -1
; SE 24
int KW 1
c Ident -1
= OP 11
3 INT -1
; SE 24
int KW 1
e Ident -1
= OP 11
0 INT -1
, SE 25
f Ident -1
= OP 11
10 INT -1
; SE 24
void KW 2
func Ident -1
( SE 20
) SE 21
{ SE 22
int KW 1
a Ident -1
= OP 11
```

- 01 11
5 INT -1
; SE 24

```
int KW 1
c Ident -1
= OP 11
2 INT -1
; SE 24
int KW 1
temp1 Ident -1
= OP 11
f Ident -1
+ OP 6
e Ident -1
; SE 24
int KW 1
temp2 Ident -1
= OP 11
f Ident -1
+ OP 6
2 INT -1
; SE 24
int KW 1
temp3 Ident -1
= OP 11
c Ident -1
; SE 24
return KW 3
a Ident -1
+ OP 6
b Ident -1
; SE 24
} SE 23
```

语法分析器输出:

```
1  program#int reduction
2  compUnit#int    reduction
3  decl#int        reduction
4  varDecl#int     reduction
5  bType#int       reduction
6  int#int move
7  varDef#Ident    reduction
8  Ident#Ident move
9  argVarDef#= reduction
10 =#= move
11 initVal#INT reduction
12 exp#INT reduction
13 assignExp#INT   reduction
14 eqExp#INT       reduction
15 relExp#INT      reduction
16 addExp#INT      reduction
17 mulExp#INT      reduction
18 unaryExp#INT    reduction
19 primaryExp#INT  reduction
20 number#INT      reduction
21 INT#INT move
22 mulExpAtom#;    reduction
23 addExpAtom#;    reduction
24 relExpAtom#;    reduction
25 eqExpAtom#; reduction
26 assignExpAtom#; reduction
27 argVarDecl#;    reduction
28 ;#; move
29 compUnit#int    reduction
30 decl#int        reduction
31 varDecl#int     reduction
32 hType#int       reduction
```

```
32  type#int      reduction
33  int#int move
34  varDef#Ident   reduction
```

```
35  Ident#Ident move
36  argVarDef#= reduction
37  =#= move
38  initVal#INT reduction
39  exp#INT reduction
40  assignExp#INT  reduction
41  eqExp#INT  reduction
42  relExp#INT  reduction
43  addExp#INT  reduction
44  mulExp#INT  reduction
45  unaryExp#INT  reduction
46  primaryExp#INT reduction
47  number#INT  reduction
48  INT#INT move
49  mulExpAtom#;  reduction
50  addExpAtom#;  reduction
51  relExpAtom#;  reduction
52  eqExpAtom#; reduction
53  assignExpAtom#; reduction
54  argVarDecl#;  reduction
55  ;#; move
56  compUnit#int  reduction
57  decl#int  reduction
58  varDecl#int reduction
59  bType#int  reduction
60  int#int move
61  varDef#Ident  reduction
62  Ident#Ident move
63  argVarDef#= reduction
64  =#= move
65  initVal#INT reduction
66  exp#INT reduction
```

67 assignExp#INT reduction


```
68  eqExp#INT    reduction
69  relExp#INT    reduction
70  addExp#INT    reduction
71  mulExp#INT    reduction
72  unaryExp#INT  reduction
73  primaryExp#INT reduction
74  number#INT    reduction
75  INT#INT move
76  mulExpAtom#;  reduction
77  addExpAtom#;  reduction
78  relExpAtom#;  reduction
79  eqExpAtom#; reduction
80  assignExpAtom#; reduction
81  argVarDecl#;  reduction
82  ;#; move
83  compUnit#int  reduction
84  decl#int      reduction
85  varDecl#int   reduction
86  bType#int     reduction
87  int#int move
88  varDef#Ident  reduction
89  Ident#Ident   move
90  argVarDef#=   reduction
91  =#= move
92  initVal#INT   reduction
93  exp#INT        reduction
94  assignExp#INT  reduction
95  eqExp#INT      reduction
96  relExp#INT      reduction
97  addExp#INT      reduction
98  mulExp#INT      reduction
99  unaryExp#INT    reduction
```

```
100 primaryExp#INT reduction
```

```
101 number#INT    reduction
102 INT#INT move
103 mulExpAtom#,    reduction
104 addExpAtom#,    reduction
105 relExpAtom#,    reduction
106 eqExpAtom#, reduction
107 assignExpAtom#, reduction
108 argVarDecl#,    reduction
109 ,#, move
110 varDef#Ident    reduction
111 Ident#Ident move
112 argVarDef#= reduction
113 =#= move
114 initVal#INT reduction
115 exp#INT reduction
116 assignExp#INT    reduction
117 eqExp#INT    reduction
118 relExp#INT    reduction
119 addExp#INT    reduction
120 mulExp#INT    reduction
121 unaryExp#INT    reduction
122 primaryExp#INT    reduction
123 number#INT    reduction
124 INT#INT move
125 mulExpAtom#;    reduction
126 addExpAtom#;    reduction
127 relExpAtom#;    reduction
128 eqExpAtom#; reduction
129 assignExpAtom#; reduction
130 argVarDecl#;    reduction
131 ;#, move
132 compUnit#void    reduction
```

133 funcDef#void reduction
134 funcType#void reduction

```
135 void#void    move
136 Ident#Ident  move
137 (#( move
138 funcFParams#) reduction
139 )#) move
140 block#{ reduction
141 {#{ move
142 blockItem#int reduction
143 decl#int      reduction
144 varDecl#int   reduction
145 bType#int     reduction
146 int#int       move
147 varDef#Ident  reduction
148 Ident#Ident  move
149 argVarDef#= reduction
150 =#= move
151 initVal#INT reduction
152 exp#INT      reduction
153 assignExp#INT reduction
154 eqExp#INT    reduction
155 relExp#INT    reduction
156 addExp#INT    reduction
157 mulExp#INT    reduction
158 unaryExp#INT  reduction
159 primaryExp#INT reduction
160 number#INT    reduction
161 INT#INT       move
162 mulExpAtom#;  reduction
163 addExpAtom#;  reduction
164 relExpAtom#;  reduction
165 eqExpAtom#; reduction
166 assignExpAtom#; reduction
```

167 argVarDecl#; reduction

```
168 ;#; move
169 blockItem#int    reduction
170 decl#int          reduction
171 varDecl#int       reduction
172 bType#int         reduction
173 int#int move
174 varDef#Ident      reduction
175 Ident#Ident move
176 argVarDef#=       reduction
177 =#= move
178 initVal#INT       reduction
179 exp#INT            reduction
180 assignExp#INT     reduction
181 eqExp#INT          reduction
182 relExp#INT         reduction
183 addExp#INT         reduction
184 mulExp#INT         reduction
185 unaryExp#INT      reduction
186 primaryExp#INT    reduction
187 number#INT        reduction
188 INT#INT move
189 mulExpAtom#;      reduction
190 addExpAtom#;      reduction
191 relExpAtom#;      reduction
192 eqExpAtom#;       reduction
193 assignExpAtom#;   reduction
194 argVarDecl#;      reduction
195 ;#; move
196 blockItem#int     reduction
197 decl#int           reduction
198 varDecl#int        reduction
199 bType#int          reduction
```

```
200 int#int move
201 varDef#Ident reduction
```


202 Ident#Ident move
203 argVarDef#= reduction
204 =#= move
205 initVal#Ident reduction
206 exp#Ident reduction
207 assignExp#Ident reduction
208 eqExp#Ident reduction
209 relExp#Ident reduction
210 addExp#Ident reduction
211 mulExp#Ident reduction
212 unaryExp#Ident reduction
213 Ident#Ident move
214 callFunc#+ reduction
215 mulExpAtom#+ reduction
216 addExpAtom#+ reduction
217 +#+ move
218 mulExp#Ident reduction
219 unaryExp#Ident reduction
220 Ident#Ident move
221 callFunc#; reduction
222 mulExpAtom#; reduction
223 addExpAtom#; reduction
224 relExpAtom#; reduction
225 eqExpAtom#; reduction
226 assignExpAtom#; reduction
227 argVarDecl#; reduction
228 ;#; move
229 blockItem#int reduction
230 decl#int reduction
231 varDecl#int reduction
232 bType#int reduction
233 int#int move

234 varDef#Ident reduction

235 Ident#Ident move
236 argVarDef#= reduction
237 =#= move
238 initVal#Ident reduction
239 exp#Ident reduction
240 assignExp#Ident reduction
241 eqExp#Ident reduction
242 relExp#Ident reduction
243 addExp#Ident reduction
244 mulExp#Ident reduction
245 unaryExp#Ident reduction
246 Ident#Ident move
247 callFunc#+ reduction
248 mulExpAtom#+ reduction
249 addExpAtom#+ reduction
250 +#+ move
251 mulExp#INT reduction
252 unaryExp#INT reduction
253 primaryExp#INT reduction
254 number#INT reduction
255 INT#INT move
256 mulExpAtom#; reduction
257 addExpAtom#; reduction
258 relExpAtom#; reduction
259 eqExpAtom#; reduction
260 assignExpAtom#; reduction
261 argVarDecl#; reduction
262 ;#; move
263 blockItem#int reduction
264 decl#int reduction
265 varDecl#int reduction
266 bType#int reduction

267 int#int move

268 varDef#Ident reduction
269 Ident#Ident move
270 argVarDef#= reduction
271 =#= move
272 initVal#Ident reduction
273 exp#Ident reduction
274 assignExp#Ident reduction
275 eqExp#Ident reduction
276 relExp#Ident reduction
277 addExp#Ident reduction
278 mulExp#Ident reduction
279 unaryExp#Ident reduction
280 Ident#Ident move
281 callFunc#; reduction
282 mulExpAtom#; reduction
283 addExpAtom#; reduction
284 relExpAtom#; reduction
285 eqExpAtom#; reduction
286 assignExpAtom#; reduction
287 argVarDecl#; reduction
288 ;#; move
289 blockItem#return reduction
290 stmt#return reduction
291 return#return move
292 argExp#Ident reduction
293 exp#Ident reduction
294 assignExp#Ident reduction
295 eqExp#Ident reduction
296 relExp#Ident reduction
297 addExp#Ident reduction
298 mulExp#Ident reduction
299 unaryExp#Ident reduction

297 unaryExp#Ident reduction

300 Ident#Ident move

301 callFunc#+ reduction

302 mulExpAtom#+ reduction

303 addExpAtom#+ reduction

304 +#+ move

305 mulExp#Ident reduction

306 unaryExp#Ident reduction

307 Ident#Ident move

308 callFunc#; reduction

309 mulExpAtom#; reduction

310 addExpAtom#; reduction

311 relExpAtom#; reduction

312 eqExpAtom#; reduction

313 assignExpAtom#; reduction

314 ;#; move

315 blockItem#} reduction

316 }#} move

317 EOF#EOF accept