

Universidade Federal de Alagoas
Instituto de Computação
Ciência da Computação

Nova - Parser

Rubens Pessoa

24 de outubro de 2016

Sumário

Sumário	i
1 Outputs	1
1.1 Hello World	1
1.2 Fibonacci	1
1.3 ShellSort	6

1 Outputs

1.1 Hello World

```
1 <program> ::= void <program_aux>
2 <program_aux> ::= main AB_PAR FEC_PAR <scope>
3 <commands> ::= printOut AB_PAR <printout_or_readin> <commands>
4 <msg> ::= CTE_STR
5 <printout_or_readin> ::= <msg> FEC_PAR SP
6 <commands> ::= empty
7 <scope> ::= { <commands> FEC_CH SP
```

1.2 Fibonacci

```
1 <program> ::= int <function_declaration> <program>
2 <parameters>:: int ID
3 <function_declaration> ::= ID AB_PAR <parameters> FEC_PAR <scope>
4 <commands> ::= int ID <declaration> <commands>
5 <declaration> ::= <attribution>
6 <value> ::= <expression>
7 <expression>:: <eq_expression> <expression_aux>
8 <eq_expression>:: <comparative_exp> <eq_expression_aux>
9 <comparative_exp>:: <add_exp> <comparative_exp_aux>
10 <add_exp>:: <mult_exp> <add_exp_aux>
11 <mult_exp>:: <neg_exp> <mult_exp_aux>
12 <neg_exp>:: <exp_aux>
13 <exp_aux>:: <atom_exp>
14 <atom_exp> ::= CTE_INT
15 <mult_exp_aux>:: empty
16 <add_exp_aux>:: empty
17 <comparative_exp_aux>:: empty
18 <eq_expression_aux>:: empty
19 <expression_aux>:: empty
20 <attribution> ::= OP_ATR <value> SP
21 <commands> ::= int ID <declaration> <commands>
22 <declaration> ::= <attribution>
23 <value> ::= <expression>
24 <expression>:: <eq_expression> <expression_aux>
25 <eq_expression>:: <comparative_exp> <eq_expression_aux>
26 <comparative_exp>:: <add_exp> <comparative_exp_aux>
```

```

27 <add_exp>:: <mult_exp> <add_exp_aux>
28 <mult_exp>:: <neg_exp> <mult_exp_aux>
29 <neg_exp>:: <exp_aux>
30 <exp_aux>:: <atom_exp>
31 <atom_exp> ::= CTE_INT
32 <mult_exp_aux>:: empty
33 <add_exp_aux>:: empty
34 <comparative_exp_aux>:: empty
35 <eq_expression_aux>:: empty
36 <expression_aux>:: empty
37 <attribution> ::= OP_ATR <value> SP
38 <commands> ::= int ID <declaration> <commands>
39 <declaration> ::= <attribution>
40 <value> ::= <expression>
41 <expression>:: <eq_expression> <expression_aux>
42 <eq_expression>:: <comparative_exp> <eq_expression_aux>
43 <comparative_exp>:: <add_exp> <comparative_exp_aux>
44 <add_exp>:: <mult_exp> <add_exp_aux>
45 <mult_exp>:: <neg_exp> <mult_exp_aux>
46 <neg_exp>:: <exp_aux>
47 <exp_aux>:: <atom_exp>
48 <atom_exp> ::= CTE_INT
49 <mult_exp_aux>:: empty
50 <add_exp_aux>:: empty
51 <comparative_exp_aux>:: empty
52 <eq_expression_aux>:: empty
53 <expression_aux>:: empty
54 <attribution> ::= OP_ATR <value> SP
55 <commands> ::= PR_IF <ifelse> <commands>
56 <expression>:: <eq_expression> <expression_aux>
57 <eq_expression>:: <comparative_exp> <eq_expression_aux>
58 <comparative_exp>:: <add_exp> <comparative_exp_aux>
59 <add_exp>:: <mult_exp> <add_exp_aux>
60 <mult_exp>:: <neg_exp> <mult_exp_aux>
61 <neg_exp>:: <exp_aux>
62 <exp_aux>:: <atom_exp>
63 <atom_exp> ::= ID
64 <mult_exp_aux>:: empty
65 <add_exp_aux>:: empty
66 <comparative_exp_aux>:: empty
67 <eq_expression_aux>:: OP_REL2 <eq_expression>

```

```

68 <eq_expression>:: <comparative_exp> <eq_expression_aux>
69 <comparative_exp>:: <add_exp> <comparative_exp_aux>
70 <add_exp>:: <mult_exp> <add_exp_aux>
71 <mult_exp>:: <neg_exp> <mult_exp_aux>
72 <neg_exp>:: <exp_aux>
73 <exp_aux>:: <atom_exp>
74 <atom_exp> ::= CTE_INT
75 <mult_exp_aux>:: empty
76 <add_exp_aux>:: empty
77 <comparative_exp_aux>:: empty
78 <eq_expression_aux>:: empty
79 <expression_aux>:: OP_OR <expression>
80 <expression>:: <eq_expression> <expression_aux>
81 <eq_expression>:: <comparative_exp> <eq_expression_aux>
82 <comparative_exp>:: <add_exp> <comparative_exp_aux>
83 <add_exp>:: <mult_exp> <add_exp_aux>
84 <mult_exp>:: <neg_exp> <mult_exp_aux>
85 <neg_exp>:: <exp_aux>
86 <exp_aux>:: <atom_exp>
87 <atom_exp> ::= ID
88 <mult_exp_aux>:: empty
89 <add_exp_aux>:: empty
90 <comparative_exp_aux>:: empty
91 <eq_expression_aux>:: OP_REL2 <eq_expression>
92 <eq_expression>:: <comparative_exp> <eq_expression_aux>
93 <comparative_exp>:: <add_exp> <comparative_exp_aux>
94 <add_exp>:: <mult_exp> <add_exp_aux>
95 <mult_exp>:: <neg_exp> <mult_exp_aux>
96 <neg_exp>:: <exp_aux>
97 <exp_aux>:: <atom_exp>
98 <atom_exp> ::= CTE_INT
99 <mult_exp_aux>:: empty
100 <add_exp_aux>:: empty
101 <comparative_exp_aux>:: empty
102 <eq_expression_aux>:: empty
103 <expression_aux>:: empty
104 <shoot> ::= CTE_INT
105 <commands> ::= PR_SHOOT <shoot> SP
106 <else> ::= empty
107 <ifelse> ::= ABAR <expression> FEC_PAR AB_CH <commands> FEC_CH <else> SP
108 <commands> ::= PR_WHILE <while> <commands>

```

```

109 <expression>:: <eq_expression> <expression_aux>
110 <eq_expression>:: <comparative_exp> <eq_expression_aux>
111 <comparative_exp>:: <add_exp> <comparative_exp_aux>
112 <add_exp>:: <mult_exp> <add_exp_aux>
113 <mult_exp>:: <neg_exp> <mult_exp_aux>
114 <neg_exp>:: <exp_aux>
115 <exp_aux>:: <atom_exp>
116 <atom_exp> ::= ID
117 <mult_exp_aux>:: empty
118 <add_exp_aux>:: empty
119 <comparative_exp_aux>:: OP_REL1 <comparative_exp>
120 <comparative_exp>:: <add_exp> <comparative_exp_aux>
121 <add_exp>:: <mult_exp> <add_exp_aux>
122 <mult_exp>:: <neg_exp> <mult_exp_aux>
123 <neg_exp>:: <exp_aux>
124 <exp_aux>:: <atom_exp>
125 <atom_exp> ::= ID
126 <mult_exp_aux>:: empty
127 <add_exp_aux>:: empty
128 <comparative_exp_aux>:: empty
129 <eq_expression_aux>:: empty
130 <expression_aux>:: empty
131 <while> ::= AB_PAR <expression> FEC_PAR <scope>
132 <commands> ::= fi <attribution_or_function_call> <commands>
133 <attribution_or_function_call> ::= <attribution>
134 <value> ::= <expression>
135 <expression>:: <eq_expression> <expression_aux>
136 <eq_expression>:: <comparative_exp> <eq_expression_aux>
137 <comparative_exp>:: <add_exp> <comparative_exp_aux>
138 <add_exp>:: <mult_exp> <add_exp_aux>
139 <mult_exp>:: <neg_exp> <mult_exp_aux>
140 <neg_exp>:: <exp_aux>
141 <exp_aux>:: <atom_exp>
142 <atom_exp> ::= ID
143 <mult_exp_aux>:: empty
144 <add_exp_aux>:: OP_AD <add_exp>
145 <add_exp>:: <mult_exp> <add_exp_aux>
146 <mult_exp>:: <neg_exp> <mult_exp_aux>
147 <neg_exp>:: <exp_aux>
148 <exp_aux>:: <atom_exp>
149 <atom_exp> ::= ID

```



```

150 <mult_exp_aux>:: empty
151 <add_exp_aux>:: empty
152 <comparative_exp_aux>:: empty
153 <eq_expression_aux>:: empty
154 <expression_aux>:: empty
155 <attribution> ::= OP_ATR <value> SP
156 <commands> ::= f1 <attribution_or_function_call> <commands>
157 <attribution_or_function_call> ::= <attribution>
158 <value> ::= <expression>
159 <expression>:: <eq_expression> <expression_aux>
160 <eq_expression>:: <comparative_exp> <eq_expression_aux>
161 <comparative_exp>:: <add_exp> <comparative_exp_aux>
162 <add_exp>:: <mult_exp> <add_exp_aux>
163 <mult_exp>:: <neg_exp> <mult_exp_aux>
164 <neg_exp>:: <exp_aux>
165 <exp_aux>:: <atom_exp>
166 <atom_exp> ::= ID
167 <mult_exp_aux>:: empty
168 <add_exp_aux>:: empty
169 <comparative_exp_aux>:: empty
170 <eq_expression_aux>:: empty
171 <expression_aux>:: empty
172 <attribution> ::= OP_ATR <value> SP
173 <commands> ::= f2 <attribution_or_function_call> <commands>
174 <attribution_or_function_call> ::= <attribution>
175 <value> ::= <expression>
176 <expression>:: <eq_expression> <expression_aux>
177 <eq_expression>:: <comparative_exp> <eq_expression_aux>
178 <comparative_exp>:: <add_exp> <comparative_exp_aux>
179 <add_exp>:: <mult_exp> <add_exp_aux>
180 <mult_exp>:: <neg_exp> <mult_exp_aux>
181 <neg_exp>:: <exp_aux>
182 <exp_aux>:: <atom_exp>
183 <atom_exp> ::= ID
184 <mult_exp_aux>:: empty
185 <add_exp_aux>:: empty
186 <comparative_exp_aux>:: empty
187 <eq_expression_aux>:: empty
188 <expression_aux>:: empty
189 <attribution> ::= OP_ATR <value> SP
190 <commands> ::= empty

```

```

191 <scope> ::= { <commands> FEC.CH SP
192 <shoot> ::= ID
193 <commands> ::= PR.SHOOT <shoot> SP
194 <scope> ::= { <commands> FEC.CH SP
195 <program> ::= void <program_aux>
196 <program_aux> ::= main AB.PAR FEC.PAR <scope>
197 <commands> ::= int ID <declaration> <commands>
198 <declaration> ::= SP
199 <commands> ::= readIn AB.PAR <printout_or_readin> <commands>
200 <printout_or_readin> ::= ID FEC.PAR SP
201 <commands> ::= int ID <declaration> <commands>
202 <declaration> ::= <attribution>
203 <value> ::= <expression>
204 <expression> ::= <eq_expression> <expression_aux>
205 <eq_expression> ::= <comparative_exp> <eq_expression_aux>
206 <comparative_exp> ::= <add_exp> <comparative_exp_aux>
207 <add_exp> ::= <mult_exp> <add_exp_aux>
208 <mult_exp> ::= <neg_exp> <mult_exp_aux>
209 <neg_exp> ::= <exp_aux>
210 <exp_aux> ::= <atom_exp>
211 <parameters_call> ::= ID <parameters_call>
212 <parameters_call> ::= empty
213 <atom_exp> ::= ID AB.PAR <parameters_call> FEC.PAR
214 <mult_exp_aux> ::= empty
215 <add_exp_aux> ::= empty
216 <comparative_exp_aux> ::= empty
217 <eq_expression_aux> ::= empty
218 <expression_aux> ::= empty
219 <attribution> ::= OP.ATR <value> SP
220 <commands> ::= empty
221 <scope> ::= { <commands> FEC.CH SP

```

1.3 ShellSort

```

1 <program> ::= void <program_aux>
2 <program_aux> ::= main AB.PAR FEC.PAR <scope>
3 <commands> ::= int ID <declaration> <commands>
4 <declaration> ::= SP
5 <commands> ::= readIn AB.PAR <printout_or_readin> <commands>
6 <printout_or_readin> ::= ID FEC.PAR SP

```

```

7 <commands> ::= int ID <declaration> <commands>
8 <declaration> ::= :: CTE_INT <declaration_aux>
9 <declaration_aux> ::= SP
10 <commands> ::= PR_FOR <for> <commands>
11 <for_steps> ::= TYPE_VALUE ID OP_ATR CTE_INT SP ID OP_REL1 ID SP ID OP_AT
12 <for> ::= AB_PAR <for_steps> FEC_PAR <scope>
13 <commands> ::= int ID <declaration> <commands>
14 <declaration> ::= SP
15 <commands> ::= readIn AB_PAR <printout_or_readin> <commands>
16 <printout_or_readin> ::= ID FEC_PAR SP
17 <commands> ::= add <attribution_or_function_call> <commands>
18 <parameters_call> ::= ID <parameters_call>
19 <parameters_call> ::= SP <parameters_call>
20 <parameters_call> ::= ID <parameters_call>
21 <parameters_call> ::= empty
22 <attribution_or_function_call> ::= ( <parameters_call> FEC_PAR SP
23 <commands> ::= empty
24 <scope> ::= { <commands> FEC_CH SP
25 <commands> ::= int ID <declaration> <commands>
26 <declaration> ::= SP
27 <commands> ::= int ID <declaration> <commands>
28 <declaration> ::= <attribution>
29 <value> ::= <expression>
30 <expression> ::= <eq_expression> <expression_aux>
31 <eq_expression> ::= <comparative_exp> <eq_expression_aux>
32 <comparative_exp> ::= <add_exp> <comparative_exp_aux>
33 <add_exp> ::= <mult_exp> <add_exp_aux>
34 <mult_exp> ::= <neg_exp> <mult_exp_aux>
35 <neg_exp> ::= <exp_aux>
36 <exp_aux> ::= <atom_exp>
37 <atom_exp> ::= CTE_INT
38 <mult_exp_aux> ::= empty
39 <add_exp_aux> ::= empty
40 <comparative_exp_aux> ::= empty
41 <eq_expression_aux> ::= empty
42 <expression_aux> ::= empty
43 <attribution> ::= OP_ATR <value> SP
44 <commands> ::= PR_WHILE <while> <commands>
45 <expression> ::= <eq_expression> <expression_aux>
46 <eq_expression> ::= <comparative_exp> <eq_expression_aux>
47 <comparative_exp> ::= <add_exp> <comparative_exp_aux>

```

```

48 <add_exp>:: <mult_exp> <add_exp_aux>
49 <mult_exp>:: <neg_exp> <mult_exp_aux>
50 <neg_exp>:: <exp_aux>
51 <exp_aux>:: <atom_exp>
52 <atom_exp> ::= ID
53 <mult_exp_aux>:: empty
54 <add_exp_aux>:: empty
55 <comparative_exp_aux>:: OP_REL1 <comparative_exp>
56 <comparative_exp>:: <add_exp> <comparative_exp_aux>
57 <add_exp>:: <mult_exp> <add_exp_aux>
58 <mult_exp>:: <neg_exp> <mult_exp_aux>
59 <neg_exp>:: <exp_aux>
60 <exp_aux>:: <atom_exp>
61 <atom_exp> ::= ID
62 <mult_exp_aux>:: empty
63 <add_exp_aux>:: empty
64 <comparative_exp_aux>:: empty
65 <eq_expression_aux>:: empty
66 <expression_aux>:: empty
67 <while> ::= ABPAR <expression> FECPAR <scope>
68 <commands> ::= gap <attribution_or_function_call> <commands>
69 <attribution_or_function_call> ::= <attribution>
70 <value> ::= <expression>
71 <expression>:: <eq_expression> <expression_aux>
72 <eq_expression>:: <comparative_exp> <eq_expression_aux>
73 <comparative_exp>:: <add_exp> <comparative_exp_aux>
74 <add_exp>:: <mult_exp> <add_exp_aux>
75 <mult_exp>:: <neg_exp> <mult_exp_aux>
76 <neg_exp>:: <exp_aux>
77 <exp_aux>:: <atom_exp>
78 <atom_exp> ::= CTE_INT
79 <mult_exp_aux>:: OP_MULT <mult_exp>
80 <mult_exp>:: <neg_exp> <mult_exp_aux>
81 <neg_exp>:: <exp_aux>
82 <exp_aux>:: <atom_exp>
83 <atom_exp> ::= ID
84 <mult_exp_aux>:: empty
85 <add_exp_aux>:: OP_AD <add_exp>
86 <add_exp>:: <mult_exp> <add_exp_aux>
87 <mult_exp>:: <neg_exp> <mult_exp_aux>
88 <neg_exp>:: <exp_aux>

```

```

89 <exp_aux>:: <atom_exp>
90 <atom_exp> ::= CTE_INT
91 <mult_exp_aux>:: empty
92 <add_exp_aux>:: empty
93 <comparative_exp_aux>:: empty
94 <eq_expression_aux>:: empty
95 <expression_aux>:: empty
96 <attribution> ::= OP_ATR <value> SP
97 <commands> ::= empty
98 <scope> ::= { <commands> FEC_CH SP
99 <commands> ::= PR_WHILE <while> <commands>
100 <expression>:: <eq_expression> <expression_aux>
101 <eq_expression>:: <comparative_exp> <eq_expression_aux>
102 <comparative_exp>:: <add_exp> <comparative_exp_aux>
103 <add_exp>:: <mult_exp> <add_exp_aux>
104 <mult_exp>:: <neg_exp> <mult_exp_aux>
105 <neg_exp>:: <exp_aux>
106 <exp_aux>:: <atom_exp>
107 <atom_exp> ::= ID
108 <mult_exp_aux>:: empty
109 <add_exp_aux>:: empty
110 <comparative_exp_aux>:: OP_REL1 <comparative_exp>
111 <comparative_exp>:: <add_exp> <comparative_exp_aux>
112 <add_exp>:: <mult_exp> <add_exp_aux>
113 <mult_exp>:: <neg_exp> <mult_exp_aux>
114 <neg_exp>:: <exp_aux>
115 <exp_aux>:: <atom_exp>
116 <atom_exp> ::= CTE_INT
117 <mult_exp_aux>:: empty
118 <add_exp_aux>:: empty
119 <comparative_exp_aux>:: empty
120 <eq_expression_aux>:: empty
121 <expression_aux>:: empty
122 <while> ::= ABPAR <expression> FEC_PAR <scope>
123 <commands> ::= gap <attribution_or_function_call> <commands>
124 <attribution_or_function_call> ::= <attribution>
125 <value> ::= <expression>
126 <expression>:: <eq_expression> <expression_aux>
127 <eq_expression>:: <comparative_exp> <eq_expression_aux>
128 <comparative_exp>:: <add_exp> <comparative_exp_aux>
129 <add_exp>:: <mult_exp> <add_exp_aux>

```

```

130 <mult_exp>:: <neg_exp> <mult_exp_aux>
131 <neg_exp>:: <exp_aux>
132 <exp_aux>:: <atom_exp>
133 <atom_exp> ::= ID
134 <mult_exp_aux>:: OP_MULT <mult_exp>
135 <mult_exp>:: <neg_exp> <mult_exp_aux>
136 <neg_exp>:: <exp_aux>
137 <exp_aux>:: <atom_exp>
138 <atom_exp> ::= CTE_INT
139 <mult_exp_aux>:: empty
140 <add_exp_aux>:: empty
141 <comparative_exp_aux>:: empty
142 <eq_expression_aux>:: empty
143 <expression_aux>:: empty
144 <attribution> ::= OP_ATR <value> SP
145 <commands> ::= PR_FOR <for> <commands>
146 <for_steps> ::= TYPE_VALUE ID OP_ATR CTE_INT SP ID OP_REL1 ID SP ID
147 <for> ::= AB_PAR <for_steps> FEC_PAR <scope>
148 <commands> ::= value <attribution_or_function_call> <commands>
149 <attribution_or_function_call> ::= <attribution>
150 <value> ::= <expression>
151 <expression>:: <eq_expression> <expression_aux>
152 <eq_expression>:: <comparative_exp> <eq_expression_aux>
153 <comparative_exp>:: <add_exp> <comparative_exp_aux>
154 <add_exp>:: <mult_exp> <add_exp_aux>
155 <mult_exp>:: <neg_exp> <mult_exp_aux>
156 <neg_exp>:: <exp_aux>
157 <exp_aux>:: <atom_exp>
158 <parameters_call> ::= ID <parameters_call>
159 <parameters_call> ::= SP <parameters_call>
160 <parameters_call> ::= ID <parameters_call>
161 <parameters_call> ::= empty
162 <atom_exp> ::= ID AB_PAR <parameters_call> FEC_PAR
163 <mult_exp_aux>:: empty
164 <add_exp_aux>:: empty
165 <comparative_exp_aux>:: empty
166 <eq_expression_aux>:: empty
167 <expression_aux>:: empty
168 <attribution> ::= OP_ATR <value> SP
169 <commands> ::= int ID <declaration> <commands>
170 <declaration> ::= <attribution>

```

```

171 <value> ::= <expression>
172 <expression>:: <eq_expression> <expression_aux>
173 <eq_expression>:: <comparative_exp> <eq_expression_aux>
174 <comparative_exp>:: <add_exp> <comparative_exp_aux>
175 <add_exp>:: <mult_exp> <add_exp_aux>
176 <mult_exp>:: <neg_exp> <mult_exp_aux>
177 <neg_exp>:: <exp_aux>
178 <exp_aux>:: <atom_exp>
179 <atom_exp> ::= ID
180 <mult_exp_aux>:: empty
181 <add_exp_aux>:: OP_AD <add_exp>
182 <add_exp>:: <mult_exp> <add_exp_aux>
183 <mult_exp>:: <neg_exp> <mult_exp_aux>
184 <neg_exp>:: <exp_aux>
185 <exp_aux>:: <atom_exp>
186 <atom_exp> ::= ID
187 <mult_exp_aux>:: empty
188 <add_exp_aux>:: empty
189 <comparative_exp_aux>:: empty
190 <eq_expression_aux>:: empty
191 <expression_aux>:: empty
192 <attribution> ::= OP_ATR <value> SP
193 <commands> ::= PR_WHILE <while> <commands>
194 <expression>:: <eq_expression> <expression_aux>
195 <eq_expression>:: <comparative_exp> <eq_expression_aux>
196 <comparative_exp>:: <add_exp> <comparative_exp_aux>
197 <add_exp>:: <mult_exp> <add_exp_aux>
198 <mult_exp>:: <neg_exp> <mult_exp_aux>
199 <neg_exp>:: <exp_aux>
200 <exp_aux>:: <atom_exp>
201 <atom_exp> ::= ID
202 <mult_exp_aux>:: empty
203 <add_exp_aux>:: empty
204 <comparative_exp_aux>:: OP_REL1 <comparative_exp>
205 <comparative_exp>:: <add_exp> <comparative_exp_aux>
206 <add_exp>:: <mult_exp> <add_exp_aux>
207 <mult_exp>:: <neg_exp> <mult_exp_aux>
208 <neg_exp>:: <exp_aux>
209 <exp_aux>:: <atom_exp>
210 <atom_exp> ::= CTE_INT
211 <mult_exp_aux>:: empty

```

```

212 <add_exp_aux>:: empty
213 <comparative_exp_aux>:: empty
214 <eq_expression_aux>:: empty
215 <expression_aux>:: OP_AND <expression>
216 <expression>:: <eq_expression> <expression_aux>
217 <eq_expression>:: <comparative_exp> <eq_expression_aux>
218 <comparative_exp>:: <add_exp> <comparative_exp_aux>
219 <add_exp>:: <mult_exp> <add_exp_aux>
220 <mult_exp>:: <neg_exp> <mult_exp_aux>
221 <neg_exp>:: <exp_aux>
222 <exp_aux>:: <atom_exp>
223 <atom_exp> ::= ID
224 <mult_exp_aux>:: empty
225 <add_exp_aux>:: empty
226 <comparative_exp_aux>:: OP_REL1 <comparative_exp>
227 <comparative_exp>:: <add_exp> <comparative_exp_aux>
228 <add_exp>:: <mult_exp> <add_exp_aux>
229 <mult_exp>:: <neg_exp> <mult_exp_aux>
230 <neg_exp>:: <exp_aux>
231 <exp_aux>:: <atom_exp>
232 <parameters_call> ::= ID <parameters_call>
233 <parameters_call> ::= SP <parameters_call>
234 <parameters_call> ::= ID <parameters_call>
235 <parameters_call> ::= empty
236 <atom_exp> ::= ID ABPAR <parameters_call> FECPAR
237 <mult_exp_aux>:: empty
238 <add_exp_aux>:: empty
239 <comparative_exp_aux>:: empty
240 <eq_expression_aux>:: empty
241 <expression_aux>:: empty
242 <while> ::= ABPAR <expression> FECPAR <scope>
243 <commands> ::= int ID <declaration> <commands>
244 <declaration> ::= <attribution>
245 <value> ::= <expression>
246 <expression>:: <eq_expression> <expression_aux>
247 <eq_expression>:: <comparative_exp> <eq_expression_aux>
248 <comparative_exp>:: <add_exp> <comparative_exp_aux>
249 <add_exp>:: <mult_exp> <add_exp_aux>
250 <mult_exp>:: <neg_exp> <mult_exp_aux>
251 <neg_exp>:: <exp_aux>
252 <exp_aux>:: <atom_exp>

```



```

253 <atom_exp> ::= ID
254 <mult_exp_aux>:: empty
255 <add_exp_aux>:: OP_AD <add_exp>
256 <add_exp>:: <mult_exp> <add_exp_aux>
257 <mult_exp>:: <neg_exp> <mult_exp_aux>
258 <neg_exp>:: <exp_aux>
259 <exp_aux>:: <atom_exp>
260 <atom_exp> ::= ID
261 <mult_exp_aux>:: empty
262 <add_exp_aux>:: empty
263 <comparative_exp_aux>:: empty
264 <eq_expression_aux>:: empty
265 <expression_aux>:: empty
266 <attribution> ::= OP_ATR <value> SP
267 <commands> ::= int ID <declaration> <commands>
268 <declaration> ::= <attribution>
269 <value> ::= <expression>
270 <expression>:: <eq_expression> <expression_aux>
271 <eq_expression>:: <comparative_exp> <eq_expression_aux>
272 <comparative_exp>:: <add_exp> <comparative_exp_aux>
273 <add_exp>:: <mult_exp> <add_exp_aux>
274 <mult_exp>:: <neg_exp> <mult_exp_aux>
275 <neg_exp>:: <exp_aux>
276 <exp_aux>:: <atom_exp>
277 <parameters_call> ::= ID <parameters_call>
278 <parameters_call> ::= SP <parameters_call>
279 <parameters_call> ::= ID <parameters_call>
280 <parameters_call> ::= empty
281 <atom_exp> ::= ID ABPAR <parameters_call> FECPAR
282 <mult_exp_aux>:: empty
283 <add_exp_aux>:: empty
284 <comparative_exp_aux>:: empty
285 <eq_expression_aux>:: empty
286 <expression_aux>:: empty
287 <attribution> ::= OP_ATR <value> SP
288 <commands> ::= setValue <attribution_or_function_call> <commands>
289 <parameters_call> ::= ID <parameters_call>
290 <parameters_call> ::= SP <parameters_call>
291 <parameters_call> ::= ID <parameters_call>
292 <parameters_call> ::= SP <parameters_call>
293 <parameters_call> ::= ID <parameters_call>

```

```

294 <parameters_call> ::= empty
295 <attribution_or_function_call> ::= ( <parameters_call> FEC_PAR SP
296 <commands> ::= j <attribution_or_function_call> <commands>
297 <attribution_or_function_call> ::= <attribution>
298 <value> ::= <expression>
299 <expression>:: <eq_expression> <expression_aux>
300 <eq_expression>:: <comparative_exp> <eq_expression_aux>
301 <comparative_exp>:: <add_exp> <comparative_exp_aux>
302 <add_exp>:: <mult_exp> <add_exp_aux>
303 <mult_exp>:: <neg_exp> <mult_exp_aux>
304 <neg_exp>:: <exp_aux>
305 <exp_aux>:: <atom_exp>
306 <atom_exp> ::= ID
307 <mult_exp_aux>:: empty
308 <add_exp_aux>:: OP_AD <add_exp>
309 <add_exp>:: <mult_exp> <add_exp_aux>
310 <mult_exp>:: <neg_exp> <mult_exp_aux>
311 <neg_exp>:: <exp_aux>
312 <exp_aux>:: <atom_exp>
313 <atom_exp> ::= ID
314 <mult_exp_aux>:: empty
315 <add_exp_aux>:: empty
316 <comparative_exp_aux>:: empty
317 <eq_expression_aux>:: empty
318 <expression_aux>:: empty
319 <attribution> ::= OP_ATR <value> SP
320 <commands> ::= empty
321 <scope> ::= { <commands> FEC_CH SP
322 <commands> ::= k <attribution_or_function_call> <commands>
323 <attribution_or_function_call> ::= <attribution>
324 <value> ::= <expression>
325 <expression>:: <eq_expression> <expression_aux>
326 <eq_expression>:: <comparative_exp> <eq_expression_aux>
327 <comparative_exp>:: <add_exp> <comparative_exp_aux>
328 <add_exp>:: <mult_exp> <add_exp_aux>
329 <mult_exp>:: <neg_exp> <mult_exp_aux>
330 <neg_exp>:: <exp_aux>
331 <exp_aux>:: <atom_exp>
332 <atom_exp> ::= ID
333 <mult_exp_aux>:: empty
334 <add_exp_aux>:: OP_AD <add_exp>

```

```

335 <add_exp>:: <mult_exp> <add_exp_aux>
336 <mult_exp>:: <neg_exp> <mult_exp_aux>
337 <neg_exp>:: <exp_aux>
338 <exp_aux>:: <atom_exp>
339 <atom_exp> ::= ID
340 <mult_exp_aux>:: empty
341 <add_exp_aux>:: empty
342 <comparative_exp_aux>:: empty
343 <eq_expression_aux>:: empty
344 <expression_aux>:: empty
345 <attribution> ::= OP_ATR <value> SP
346 <commands> ::= setValue <attribution_or_function_call> <commands>
347 <parameters_call> ::= ID <parameters_call>
348 <parameters_call> ::= SP <parameters_call>
349 <parameters_call> ::= ID <parameters_call>
350 <parameters_call> ::= SP <parameters_call>
351 <parameters_call> ::= ID <parameters_call>
352 <parameters_call> ::= empty
353 <attribution_or_function_call> ::= ( <parameters_call> FEC_PAR SP
354 <commands> ::= empty
355 <scope> ::= { <commands> FEC_CH SP
356 <commands> ::= empty
357 <scope> ::= { <commands> FEC_CH SP
358 <commands> ::= empty
359 <scope> ::= { <commands> FEC_CH SP

```
