GRAMATICA

1. <PROGRAM> → class (id) <DECLARA> <ESTATUTOS> endclass (0)
2. <DECLARA> → declare <B> of <TIPO>; <AUX> (1)
3. <DECLARA> → € (2)
4. <B> → <ID\_DIM> <AUX2> (3)
5. <ID\_DIM> → id <C> (4)
6. <C> → [cteentera] <AUX3> (5)
7. <AUX3> → <C> (6)
8. <AUX3> → € (7)
9. <AUX2> → <B> (8)
10. <AUX2> → € (9)
11. <TIPO> → int (10)
12. <TIPO> → float (11)
13. <TIPO> → carácter (12)
14. <TIPO> → string (13)
15. <AUX> → <DECLARA> (14)
16. <AUX> → € (15)
17. <ESTATUTOS> → <E> ; <AUX4> (16)
18. <ESTATUTOS> → € (17)
19. <E> → <EST\_ASIG> (18)
20. <E> → <EST\_IF> (19)
21. <E> → <EST\_WHILE> (20)
22. <E> → <EST\_DO> (21)
23. <E> → <EST\_READ> (22)
24. <E> → <EST\_WRITE> (23)
25. <AUX4> → <ESTATUTOS> (24)
26. <AUX4> → € (25)
27. <EST\_ASIG> → <ASIG> = <EXPR> (26)
28. <ASIG> → id <DIM\_ASIG> (27)
29. <DIM\_ASIG> → [<EXPR> <AUX5>] (28)
30. <DIM\_ASIG> → € (29)
31. <AUX5> → , <EXPR> <AUX5> (30)
32. <AUX5> → € (31)
33. <EST\_IF> → if(<EXPR>) <ESTATUTOS> <H> endif (32)
34. <H> → else <ESTATUTOS> (33)
35. <H> → € (34)
36. <EST\_WHILE> → while (<EXPR>) <ESTATUTOS> endwhile (35)
37. <EST\_DO> → do <ESTATUTOS> dowhile (<EXPR>) enddo (36)
38. <EST\_READ> → read (<I>) (37)
39. <I> → id <AUX6> (38)
40. <AUX6> → , <I> (39)
41. <AUX6> → € (40)
42. <EST\_WRITE> → write (<J>) (41)
43. <J> → <EXPR> <AUX7> (42)
44. <AUX7> → , <J> (43)
45. <AUX7> → € (44)
46. <EXPR> → <EXPR2> <AUX8> (45)
47. <AUX8> → || <EXPR2> <AUX8> (46)
48. <AUX8> → € (47)
49. <EXPR2> → <EXP3> <AUX9> (48)
50. <AUX9> → && <EXP3> <AUX9> (49)
51. <AUX9> → € (50)
52. <EXPR3> → **<NOT>** <EXPR4> (51)
53. **<NOT>** → ! (52)
54. **<NOT>** → € (53)
55. <EXPR4> → <EXPR5> <M> (54)
56. <M> → <OPREL> <EXPR5> (55)
57. **<M> → € (56)**
58. <OPREL> → == (57)
59. <OPREL> → != (58)
60. <OPREL> → < (59)
61. <OPREL> → <= (60)
62. <OPREL> → > (61)
63. <OPREL> → >= (62)
64. <EXPR5> → <TERM> <AUX10> (63)
65. <AUX10> → + <EXPR5> (64)
66. <AUX10> → - <EXPR5> (65)
67. <AUX10> → € (66)
68. <TERM> → <FACT> <AUX11> (67)
69. <AUX11> → \* **<TERM> (68)**
70. <AUX11> → / **<TERM> (69)**
71. <AUX11> → % **<TERM> (70)**
72. <AUX11> → € (71)
73. <FACT> → <ASIG> (72)
74. <FACT> → cteentera (73)
75. <FACT> → ctereal (74)
76. <FACT> → ctenotacion (75)
77. <FACT> → ctecaracter (76)
78. <FACT> → ctestring (77)
79. <FACT> → ( <EXPR> ) (78)
80. <C> → € (79)

FIRST

<PROGRAM> {class}

<DECLARA> {declare, €}

<B> {id}

<ID\_DIM> {id}

<C> { [, €}

<AUX3> { [, €}

<AUX2> {id , €}

<TIPO> {int, float, char, string}

<AUX> {declare €}

<ESTATUTOS> {id, if, while, do, read, write, €}

<E> {id, if, while, do, read, write}

<AUX4> {id, if, while, do, read, write, €}

<EST\_ASIG> {id}

<ASIG> {id}

<DIM\_ASIG> { [ € }

<AUX5> { , , €}

<EST\_IF> {if}

<H> {else, €}

<EST\_WHILE> {while}

<EST\_DO> {do}

<EST\_READ> {read}

<I> {id}

<AUX6> { , , €}

<EST\_WRITE> {write}

<J> {!, id, cteentera, ctereal, ctenotacion, ctecaracter, ctestring,(}

<AUX7> {,, €}

**<EXPR> {!, €} \*igual\***

<AUX8> {||, €}

**<EXPR2> {! €} \*igual\***

<AUX9> {&&, €}

<EXPR3> {!,€} \*diferente\*

**<NOT> {!, €}**

<EXPR4> {id, cteentera, ctereal, ctenotacion, ctecaracter, ctestring,(}

<M> {==, !=, <, <=, >, >= € }

<OPREL> {==, !=, <, <=, >, >=}

<EXPR5> {id, cteentera, ctereal, ctenotacion, ctecaracter, ctestring,(}

<AUX10> {+, -, €}

<TERM> {id, cteentera, ctereal, ctenotacion, ctecaracter, ctestring,(}

<AUX11> {\*, /, %, €}

<FACT> {id, cteentera, ctereal, ctenotacion, ctecaracter, ctestring,(}

FOLLOW

<PROGRAM> {$}

<DECLARA> {id, if, while, do, read, write, endclass, else, endwhile, dowhile, endif}

<B> {of}

<ID\_DIM> {id , of }

<C> {id , of }

<AUX3> {id , of }

<AUX2> {of}

<TIPO> {;}

<AUX> {id, if, while, do, read, write, endclass, else, endwhile, dowhile, endif }

<ESTATUTOS> {endclass, else, endwhile, dowhile, endif}

<E> {;}

<AUX4> {endclass, else, endwhile, dowhile, endif}

<EST\_ASIG> {;}

<ASIG> {=}

<DIM\_ASIG> {=}

<AUX5> { ] }

<EST\_IF> {;}

<H> {endif}

<EST\_WHILE> {;}

<EST\_DO> {;}

<EST\_READ> {;}

<I> { ) }

<AUX6> { ) }

<EST\_WRITE> {;}

<J> { ) }

<AUX7> { ) }

<EXPR> {;, ], ,, )} {; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

<AUX8> {;, ], ,, )} {; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

<EXPR2> {||, ;, ], ,, )} {||, ; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

<AUX9> {||, ;, ], ,, )} {||, ; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

<EXPR3> {&&, ||, ;, ], ,, )} {&&,||, ; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

**<NOT> {id, cteentera, ctereal, ctenotacion, ctecaracter, ctestring,( }**

<EXPR4> {&&, ||, ;, ], ,, )} {&&,||, ; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

<M> {&&, ||, ;, ], ,, )} {&&,||, ; , , , ], )} \*es lo mismo pero conforme lo hice cambie el orden nada mas\*

<OPREL> {id, cteentera, cteeal, ctenotacion, ctecaracter, ctestring, (}

<EXPR5> {==, !=, <, <=, >, >=} **{==, !=, <, <=, >, >= , &&,||, ; , , , ], )} \*AQUI YA ME CAMBIARON LOS FOLLOWS\***

<AUX10> {==, !=, <, <=, >, >=} >=} **{==, !=, <, <=, >, >= , &&,||, ; , , , ], )} \*AQUI YA ME CAMBIARON LOS FOLLOWS\***

<TERM> {==, !=, <, <=, >, >=, +, - } **{+, -,==, !=, <, <=, >, >= , &&,||, ; , , , ], )} \*AQUI YA ME CAMBIARON LOS FOLLOWS\***

<AUX11> {==, !=, <, <=, >, >=, +, - } **{+, -,==, !=, <, <=, >, >= , &&,||, ; , , , ], )} \*AQUI YA ME CAMBIARON LOS FOLLOWS\***

<FACT> {\*, /, %, ==, !=, <, <=, >, >=, +, -} **{\*, /, %,** **+, -,==, !=, <, <=, >, >= , &&,||, ; , , , ], )} \*AQUI YA ME CAMBIARON LOS FOLLOWS\***