

## Syntax Analyzer Grammar

T = {IDENT\_T, NUMLIT\_T, LISTOP1\_T, PLUS\_T, MINUS\_T, GT\_T, LT\_T, TRUE\_T, FALSE\_T, DIV\_T, MULT\_T, EQUALTO\_T, GTE\_T, LTE\_T, LPAREN\_T, RPAREN\_T, SQUOTE\_T, IDENT\_T, IF\_T, COND\_T, DISPLAY\_T, NEWLINE\_T, AND\_T, OR\_T, NOT\_T, DEFINE\_T, LET\_T, LISTOP2\_T, NUMBERP\_T, LISTP\_T, ZEROP\_T, NULLP\_T, EOF\_T, MODULO\_T, ROUND\_T, READ\_T, ELSE\_T, STRLIT\_T, ERROR\_T, EOF\_T, MAX\_TOKENS};

NT = {<program>, <more\_defines>, <define>, <stmt\_list>, <stmt>, <literal>, <quoted\_lit>, <logical\_lit>, <more\_tokens>, <param\_list>, <else\_part>, <stmt\_pair>, <stmt\_pair\_body>, <assign\_pair>, <more\_assigns>, <action>, <any\_other\_token>}

S = <program>

P = {

1. <program> -> LPAREN\_T <define> LPAREN\_T <more\_defines> EOF\_T
2. <more\_defines> -> <define> LPAREN\_T <more\_defines>
3. <more\_defines> -> IDENT\_T <stmt\_list> RPAREN\_T
4. <define> -> DEFINE\_T LPAREN\_T IDENT\_T <param\_list> RPAREN\_T <stmt> <stmt\_list>  
RPAREN\_T
5. <stmt\_list> -> <stmt> <stmt\_list>
6. <stmt\_list> ->  $\lambda$
7. <stmt> -> <literal>
8. <stmt> -> IDENT\_T
9. <stmt> -> LPAREN\_T <action> RPAREN\_T
10. <literal> -> NUMLIT\_T
11. <literal> -> STRLIT\_T
12. <literal> -> SQUOTE\_T <quoted\_lit>
13. <literal> -> <logical\_lit>
14. <quoted\_lit> -> <any\_other\_token>
15. <logical\_lit> -> TRUE\_T
16. <logical\_lit> -> FALSE\_T
17. <more\_tokens> -> <any\_other\_token> <more\_tokens>
18. <more\_tokens> ->  $\lambda$
19. <param\_list> -> IDENT\_T <param\_list>
20. <param\_list> ->  $\lambda$
21. <else\_part> -> <stmt>
22. <else\_part> ->  $\lambda$
23. <stmt\_pair> -> LPAREN\_T <stmt\_pair\_body>
24. <stmt\_pair> ->  $\lambda$
25. <stmt\_pair\_body> -> <stmt> <stmt> RPAREN\_T <stmt\_pair>
26. <stmt\_pair\_body> -> ELSE\_T <stmt> RPAREN\_T
27. <assign\_pair> -> LPAREN\_T IDENT\_T <stmt> RPAREN\_T
28. <more\_assigns> -> <assign\_pair> <more\_assigns>
29. <more\_assigns> ->  $\lambda$
30. <action> -> IF\_T <stmt> <stmt> <else\_part>
31. <action> -> COND\_T LPAREN\_T <stmt\_pair\_body>
32. <action> -> LET\_T LPAREN\_T <more\_assigns> RPAREN\_T <stmt> <stmt\_list>
33. <action> -> LISTOP1\_T <stmt>
34. <action> -> LISTOP2\_T <stmt> <stmt>
35. <action> -> AND\_T <stmt\_list>
36. <action> -> OR\_T <stmt\_list>
37. <action> -> NOT\_T <stmt>

38. <action> -> NUMBERP\_T <stmt>  
 39. <action> -> LISTP\_T <stmt>  
 40. <action> -> ZEROP\_T <stmt>  
 41. <action> -> NULLP\_T <stmt>  
 42. <action> -> EOFP\_T <stmt>  
 43. <action> -> PLUS\_T <stmt\_list>  
 44. <action> -> MINUS\_T <stmt> <stmt\_list>  
 45. <action> -> DIV\_T <stmt> <stmt\_list>  
 46. <action> -> MULT\_T <stmt\_list>  
 47. <action> -> MODULO\_T <stmt> <stmt>  
 48. <action> -> ROUND\_T <stmt>  
 49. <action> -> EQUALTO\_T <stmt\_list>  
 50. <action> -> GT\_T <stmt\_list>  
 51. <action> -> LT\_T <stmt\_list>  
 52. <action> -> GTE\_T <stmt\_list>  
 53. <action> -> LTE\_T <stmt\_list>  
 54. <action> -> IDENT\_T <stmt\_list>  
 55. <action> -> DISPLAY\_T <stmt>  
 56. <action> -> NEWLINE\_T  
 57. <action> -> READ\_T  
 58. <any\_other\_token> -> LPAREN\_T <more\_tokens> RPAREN\_T  
 59. <any\_other\_token> -> IDENT\_T  
 60. <any\_other\_token> -> NUMLIT\_T  
 61. <any\_other\_token> -> STRLIT\_T  
 62. <any\_other\_token> -> IF\_T  
 63. <any\_other\_token> -> DISPLAY\_T  
 64. <any\_other\_token> -> NEWLINE\_T  
 65. <any\_other\_token> -> READ\_T  
 66. <any\_other\_token> -> LISTOP1\_T  
 67. <any\_other\_token> -> LISTOP2\_T  
 68. <any\_other\_token> -> AND\_T  
 69. <any\_other\_token> -> OR\_T  
 70. <any\_other\_token> -> NOT\_T  
 71. <any\_other\_token> -> DEFINE\_T  
 72. <any\_other\_token> -> LET\_T  
 73. <any\_other\_token> -> NUMBERP\_T  
 74. <any\_other\_token> -> LISTP\_T  
 75. <any\_other\_token> -> ZEROP\_T  
 76. <any\_other\_token> -> NULLP\_T  
 77. <any\_other\_token> -> EOFP\_T  
 78. <any\_other\_token> -> PLUS\_T  
 79. <any\_other\_token> -> MINUS\_T  
 80. <any\_other\_token> -> DIV\_T  
 81. <any\_other\_token> -> MULT\_T  
 82. <any\_other\_token> -> MODULO\_T  
 83. <any\_other\_token> -> ROUND\_T  
 84. <any\_other\_token> -> EQUALTO\_T  
 85. <any\_other\_token> -> GT\_T  
 86. <any\_other\_token> -> LT\_T  
 87. <any\_other\_token> -> GTE\_T  
 88. <any\_other\_token> -> LTE\_T  
 89. <any\_other\_token> -> QUOTE\_T  
 <any\_other\_token> 90. <any\_other\_token> -> COND\_T

```
91. <any_other_token> -> ELSE_T
92. <any_other_token> -> TRUE_T
93. <any_other_token> -> FALSE_T
}
```