

1. `<start>` -> require "ifj21" `<program>`
2. `<program>` -> `<fnc_dec>` `<program>`
3. `<program>` -> `<fnc_call>` `<program>`
4. `<program>` -> `<fnc_def>` `<program>`
5. `<program>` -> ϵ
6. `<fnc_dec>` -> global id_fnc : function(`<params_dec>`) `<return_type>`
7. `<params_dec>` -> ϵ
8. `<params_dec>` -> `<data_type>` `<params_dec2>`
9. `<params_dec2>` -> ϵ
10. `<params_dec2>` -> , `<data_type>` `<params_dec2>`
11. `<return_type>` -> : `<data_type>`
12. `<return_type>` -> ϵ
13. `<data_type>` -> integer
14. `<data_type>` -> number
15. `<data_type>` -> string
16. `<fnc_call>` -> id_fnc (`<value>`)
17. `<value>` -> ϵ
18. `<value>` -> `<value_last>` `<value2>`
19. `<value2>` -> , `<value_last>` `<value2>`
20. `<value2>` -> ϵ
21. `<value_last>` -> id_var
22. `<value_last>` -> integer_value
23. `<value_last>` -> number_value
24. `<value_last>` -> string_value
25. `<value_last>` -> nil
26. `<fnc_def>` -> `<fnc_head>` `<fnc_def2>` end
27. `<fnc_head>` -> function id_fnc (`<params_def>`)
28. `<fnc_def2>` -> `<fnc_body>` `<return_void>`
29. `<fnc_def2>` -> : `<data_type>` `<fnc_body>` `<return>`
30. `<params_def>` -> ϵ
31. `<params_def>` -> `<var_def>` `<params_def2>`
32. `<params_def2>` -> ϵ
33. `<params_def2>` -> , `<var_def>` `<params_def2>`
34. `<var_def>` -> id_var : `<data_type>`

35. $\langle \text{return} \rangle \rightarrow \text{return } \langle \text{expr} \rangle$

36. $\langle \text{return_void} \rangle \rightarrow \text{return}$

37. $\langle \text{return_void} \rangle \rightarrow \epsilon$

38. $\langle \text{fnc_body} \rangle \rightarrow \langle \text{if} \rangle \langle \text{fnc_body} \rangle$

39. $\langle \text{fnc_body} \rangle \rightarrow \langle \text{loop} \rangle \langle \text{fnc_body} \rangle$

40. $\langle \text{fnc_body} \rangle \rightarrow \langle \text{statement} \rangle \langle \text{fnc_body} \rangle$

41. $\langle \text{fnc_body} \rangle \rightarrow \epsilon$

42. $\langle \text{statement} \rangle \rightarrow \langle \text{var_dec} \rangle$

43. $\langle \text{statement} \rangle \rightarrow \text{id_var} = \langle \text{var_assign} \rangle$

44. $\langle \text{statement} \rangle \rightarrow \langle \text{fnc_call} \rangle$

45. $\langle \text{var_dec} \rangle \rightarrow \text{local } \langle \text{var_def} \rangle \langle \text{var_dec_init} \rangle$

46. $\langle \text{var_dec_init} \rangle \rightarrow = \langle \text{var_assign} \rangle$

47. $\langle \text{var_dec_init} \rangle \rightarrow \epsilon$

48. $\langle \text{var_assign} \rangle \rightarrow \langle \text{expr} \rangle$

49. $\langle \text{var_assign} \rangle \rightarrow \langle \text{fnc_call} \rangle$

50. $\langle \text{if} \rangle \rightarrow \text{if } \langle \text{expr} \rangle \text{ then } \langle \text{statements} \rangle \text{ else } \langle \text{statements} \rangle \text{ end}$

51. $\langle \text{loop} \rangle \rightarrow \text{while } \langle \text{expr} \rangle \text{ do } \langle \text{statements} \rangle \text{ end}$

52. $\langle \text{statements} \rangle \rightarrow \epsilon$

53. $\langle \text{statements} \rangle \rightarrow \langle \text{statement} \rangle \langle \text{statements} \rangle$

POZN.: Slova a znaky psaná černě jsou neterminály.

POZN.2: Znak mezery v pravidlech je irelevantní a nic neznačí, slouží pouze k zpřehlednění.

POZN.3: Neterminál $\langle \text{expr} \rangle$ není rozšířen a zpracuje ho analýza shora dolů.