Syntaktická analýza neobsahující výrazy je implementován pomocí rekurzivního sestupu pro LL gramatiku, kde každý neterminál je charakterizován pomocí funkce.

1. <FUNC> -> eps

3 <FUNC> -> function id (<ARG>) : <TYPE> ; <FORWARD>

4 <FORWARD> -> <DEK> <SLOZ> ; <FUNC>

5 <FORWARD> -> forward ; <FUNC>

Zde máme LL pravidla pro deklarace a definice funkcí, první pravidlo popisuje, že vstupní soubor nemusí obsahovat žádnou funkci a tudíž ostatní pravidla ani nebudou využity. Druhé pravidlo už je pro program, který alespoň jednu deklaraci či definici funkcí obsahuje, přičemž syntakticky správně napsaná funkce musí obsahovat: popsat, třeba celou, fiction id, závorka arg nemusí být, typ, Neterminál forward slouží pro rozlišení zda se jedná o Deklaraci či definici funkce, k rozlišení se používá klíčové slovo forward. Neterminál forward dále obsahuje neterminál func, který se může rozdělit na další deklaraci či definici funkce

Při volání funkcí nastává lehce odlišná situace, nejprve parser v glob tab symbol zjistí zda identifikátor je funkce (function\_header), Pokud ano a pokud je na vstupu i očekávaná závorka, s každým parametrem odděleným čárkou se volá vyhodnocování výrazů s následnou typovou kountrolou

a:=s(5)