# Analyzátor struktury programu

Analýza struktury programu je implementována metodou shora dolů rekurzivním sestupem. Vstupem této analýzy jsou tokeny, které vrací lexikální analyzátor. Výstupem je při chybě označení chyby, ke které při analýze došlo. Pokud nedošlo k žádné chybě, je výstupem instrukční páska pro interpret, jejíž posloupnost instrukcí odpovídá vstupnímu programu, a také globální tabulka symbolů obsahující informace o funkcích (např. místo potřebné pro uložení parametrů a proměnných funkce).

První funkcí analýzy je syntaktická analýza, která kontroluje správnou strukturu programu. Tato struktura je definovaná LL gramatikou. Další funkcí analýzy je sémantická analýza, ve které se s přístupem do tabulky symbolů kontroluje sémantická správnost programu, a následně se vygeneruje odpovídající kód do instrukční pásky.

Pokud analyzátor struktury programu očekává výraz (dostane se v LL gramatice k neterminálu EXPR), předává řízení analyzátoru výrazů. Po zpracování výrazu se řízení vrátí analýze struktury programu.

Při implementaci přiložené LL gramatiky se narazilo na 2 konflikty.

Prvním z nich je nejednoznačnost mezi pravidly 38 a 39. Pravidlo 39 očekává identifikátor, ten ale může být jako první obsažen i ve výrazu. Vyřešení této nejednoznačnosti je přístup do tabulky symbolů, a na základě sémantické hodnoty identifikátoru se rozhodne o použitém pravidlu. Pokud se jedná o identifikátor proměnné, musí se jednat o výraz a řízení je tedy předáno analýze výrazů (spolu s předáním již načteného tokenu id). Pokud se jedná o funkci, očekává se volání funkce, tedy pravidlo 39. Pokud identifikátor v tabulce symbolů není, jedná se v obou případech o chybu nedefinované proměnné nebo funkce.

Druhý z nich je nejednoznačnost mezi pravidlem 44 a 45, kde může existovat více derivačních stromů if/else struktur. Tato nejednoznačnost je vyřešena jednoduše použitím pravidla 44 pokud je to možné, takže se bude vždy else vztahovat k nejbližšímu předchozímu if.

# Tabulka symbolů

Tabulka symbolů je implementována jako zásobníková struktura (ačkoliv přistupovat lze i k jiným položkám než jen na vrcholu zásobníku), kde jednotlivé položky jsou hashovací tabulky. Jednotlivé položky odpovídají vždy jednomu bloku ve zdrojovém kódu. Takto je vyřešena viditelnost proměnných ve vícenásobně vnořených blocích, protože tabulku symbolů lze prohledávat od nejvíce vnořeného bloku (hashovací tabulka na vrcholu zásobníku) a vrácené informace identifikátoru odpovídají první nalezené shodě.

První hashovací tabulka na zásobníku je globální tabulka symbolů (odpovídá bloku, kterým je celý program), která z definice jazyka může obsahovat jen funkce a k nim potřebné informace (jazyk nepodporuje globální proměnné). Ve všech dalších hashovacích tabulkách jsou jen informace potřebné pro proměnné (jazyk nepodporuje vnořené funkce). Po opuštění bloku při analýze struktury programu se odpovídající hashovací tabulka spolu se všemi informacemi o proměnných uvolní. Na konci analýzy programu je tedy jedinou hashovací tabulkou na zásobníku ta, která obsahuje informace o funkcích. K těmto informacím se přistupuje ještě v době interpretace programu.

# Hashovací tabulka

Pro uchovávání informací o funkcích nebo proměnných je implementována hashovací tabulka (tabulka s rozptýlenými položkami) s explicitním zřetězením, tedy rozptylové pole obsahuje vždy ukazatel na na první položku v lineárním seznamu synonym. Jednotlivé položky obsahují kromě klíče (název proměnné nebo funkce) strukturu, která obsahuje informace o dané proměnné nebo funkci. Explicitní zřetězení bylo zvoleno, protože velikost rozptylového pole je implementačně pevně daná, ale počet identifikátorů může být neomezený. Velikost rozptylového pole byla zvolena tak, aby docházelo k minimálnímu počtu kolizí při obvyklém počtu identifikátorů ve zdrojovém programu, ale přesto nebyla zbytečně velká, protože může existovat několik hashovacích tabulek současně, podle vnoření bloků ve zdrojovém kódu. Rozptylová funkce byla převzata ze zadání programu hashovací tabulky do předmětu IJC (Jazyk C), kde byla využívána pro výpočet klíče anglických slov. Při vytvoření položky hashovací tabulka inicializuje některé složky struktury.