

Deterministická syntaktická analýza

Thursday, May 30, 2013 8:41 AM

Např. s pomocí **prediktivního parseru** = rekurzivně sestupný parser, který nevyžaduje zpětné kroky. Je ho možné vytvořit jen pro LL(k) gramatiky, což jsou bezkontextové gramatiky, pro které existuje kladné k , které umožní rekurzivně sestupnému parseru rozhodnout se, které přepisovací pravidlo použít na základě k dalších načtených symbolů. LL(k) gramatiky vylučují mnohoznačnost a levou rekurzi. Jakákoliv bezkontextová gramatika může být transformována na ekvivalentní nelevorekurzivní gramatiku, ale odstranění levé rekurze ne vždy vede k LL(k) gramatice. Prediktivní parser běží v lineárním čase.

Deterministická syntaktická analýza využívá další informace získané při překladu – obsah zásobníku a obsah nepřechtených vstupních tokenů. Na základě těchto informací umožňují následující funkce vhodný výběr přepisovacích pravidel, a tím prediktivní parsování:

FIRST(A) = množina terminálů, kterými mohou začínat řetězce odvozené z A (**terminály na začátku A**)

Funkce first zjišťuje, co vznikne přepsáním jednotlivých neterminálů na levé straně všech pravidel.

- $A \Rightarrow bxAyB \rightarrow b$ náleží FIRST(A)

Algoritmus výpočtu

- FIRST(A) pro A = terminál nebo e: je terminál/e
- Když je A neterminál:
 - Je to první terminál ve všech přepisovacích pravidlech s A na levé straně
 - Pokud jsou v přepisovacím pravidle na P straně jen neterminály, hledá se FIRST prvního neterminálu na pravé straně
 - Pokud lze nějaký z těchto neterminálů přepsat na prázdný řetězec e, je potřeba se podívat na *first* neterminálu následujícího po něm v některém z přepisovacích pravidel

FOLLOW(A) = množina terminálů, které mohou následovat za A v některé větě formě v derivacích (**terminály hned za A**)

- $S \Rightarrow bxCyZ \rightarrow y$ náleží FOLLOW(A)

Algoritmus výpočtu

1. Polož FOLLOW (A) = \emptyset
2. Je-li A počáteční symbol G, přidej e do FOLLOW(A)
3. Pro všechny pravé strany pravidel z G tvaru $\alpha A \beta$ přidej FIRST (β) do FOLLOW (A), nepřidávej ale e.
4. Je-li v G pravidlo $L \rightarrow \alpha A$ nebo $L \rightarrow \alpha A \beta$, kde FIRST (β) obsahuje e, pak přidej do FOLLOW (A) množinu FOLLOW (L)

Vytváříme vždy pro všechny neterminály zároveň!

FIRST_k(A), FOLLOW_k(A) = zobecnění na množiny terminálních řetězců o délce nejvýše k

Tyto funkce slouží k vytvoření **rozkladové tabulky**, která nahrazuje přechodovou funkci. V první řádce jsou uvedeny všechny možné vstupy, v prvním sloupci všechny možné stavy vrcholu zásobníku

(vč. dna zásobníku #). Má stavy:

- **Srovnání (pop)** – na vstupu i na vrcholu zásobníku jsou stejné hodnoty
- **Přijetí (accept)** – bylo dosaženo dna zásobníku a přijímá se prázdný symbol ϵ
- **expanze (expand)** – aplikace přepisovacího pravidla, které je uvedené v buňce tabulky určené vstupem (který sloupec) a vrcholem zásobníku (který řádek)
- **chyba (error)** – pokud pro vstup a hodnotu na vrcholu zásobníku je buňka v tabulce prázdná → vstupní řetězec není větou jazyka

Syntactic Analysis in terms of **bottom up algorithms: LR, SLR, LALR**

From <<https://d.docs.live.net/e3534876709763a3/Dokumenty/ZCU/Statnice/Statnice.docx>>

