

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií

SIMPLE URL

Opis a preklad programovacích jazykov

Ján Suchal, Ivan Škovran
SI-2, FIIT

Zadanie

Pre definovanú gramatiku vytvorte tabuľkou riadený syntaktický analyzátor (SA) zhora nadol. Pri tvorbe SA postupujte v nasledujúcich fázach a krokoch:

Analýza

1. Vytvorte niekoľko príkladov viet daného jazyka.
2. Prepíšte gramatiku z BNF (Backus Naur Form) do gramatických pravidiel s alternatívami. Vyznačte miesta, kde gramatika nespĺňa podmienky pre deterministickú analýzu.
3. Transformujte gramatiku tak, aby bola LL(1):
 - odstráňte ľavú rekurziu (ak treba)
 - urobte ľavú faktorizáciu (ak treba)
4. Nájdite
 - množinu FIRST pre každý neterminál v transformovanej gramatike
 - množinu FOLLOW pre každý neterminál v transformovanej gramatikePresvedčte sa, že transformovaná gramatika je LL(1).
5. Pre transformovanú LL(1) gramatiku vytvorte tabuľku prechodov.

Implementácia

6. Implementujte lexikálny analyzátor rozpoznávajúci potrebné lexikálne jednotky.
7. Vytvorte tabuľkou riadený syntaktický analyzátor (SA), ktorý
 - analyzuje postupnosť lexikálnych jednotiek na vstupe
 - ak postupnosť zodpovedá vete jazyka, skončí SA prijatím; inak oznámi chybu
 - vypisuje protokol o svojej činnosti obsahujúci informáciu o uskutočnených akciách pri parsovaní vstupu
8. SA testujte na vetách jazyka vytvorených v 1)

simpleURL je jazyk pre zápis URL:

```
W list := url {;url}
V url := httpaddress | ftpaddress | telnetaddress | mailtoaddress.
H httpaddress := "http://" hostport ["/" path] ["?" search].
F ftpaddress := "ftp://" login "/" path.
T telnetaddress := "telnet://" login.
M mailtoaddress := "mailto:" xalphas "@" hostname.
L login := [user [":" password ] "@" ] hostport.
O hostport := hostname [":" port].
N hostname := xalphas { "." xalphas }.
R port := digits.
C path := segment { "/" segment }.
S search := xalphas { "+" xalphas }.
U user := xalphas.
P password := xalphas.
G segment := { xalpha }.
X xalphas := xalpha { xalpha }.
Y xalpha := alpha | digit.
I digits := digit { digit }.
A alpha := "A" | .. | "Z" | "a" | .. | "z".
D digit := "0" | .. | "9".
```

Analýza a návrh

1. Príklady viet zadaného jazyka

<http://joky.info/index.php?page=1>

<ftp://ftp.jsmf.net/cohen/>

<telnet://acidlog.heslo@fiit.stuba.sk:8752>

<mailto:janosik@pobox.sk>

2. Prepis do gramatických pravidiel s alternatívami

$W \rightarrow V \mid V \text{“;”} W$

$V \rightarrow H \mid F \mid T \mid M$

$H \rightarrow \text{“http://”} O \alpha \beta$

$\alpha \rightarrow \epsilon \mid \text{“/”} C$

$\beta \rightarrow \epsilon \mid \text{“?”} S$

$O \rightarrow N \gamma$

$\gamma \rightarrow \epsilon \mid \text{“.”} R$

$N \rightarrow X \delta$

$\delta \rightarrow \epsilon \mid \text{“.”} X \delta$

$X \rightarrow \epsilon \mid Y X$

$Y \rightarrow A \mid D$

$R \rightarrow I$

*

$I \rightarrow D I \mid \epsilon$

$C \rightarrow G \pi$

$\pi \rightarrow \epsilon \mid \text{“/”} G \pi$

$G \rightarrow \epsilon \mid Y G$

$S \rightarrow X \Omega$

$\Omega \rightarrow \epsilon \mid \text{“+”} X \Omega$

$F \rightarrow \text{“ftp://”} L \text{“/”} C$

$L \rightarrow \Sigma O$

$\Sigma \rightarrow U \Theta \text{“@”} \mid \epsilon$

$U \rightarrow X$

*

$\Theta \rightarrow \text{“.”} P \mid \epsilon$

$P \rightarrow X$

*

$T \rightarrow \text{“telnet://”} L$

$M \rightarrow \text{“mailto:”} X \text{“@”} N$

$A \rightarrow \text{“A”} \mid \dots \mid \text{“Z”} \mid \text{“a”} \mid \dots \mid \text{“z”}$

$D \rightarrow \text{“0”} \mid \dots \mid \text{“9”}$

3. Transformácia na gramatiku LL(1)

V riadkoch sú hviezdičkami označené premenovávacie pravidlá, ktoré môžeme jednoducho odstrániť. V gramatike sme urobili ľavú faktorizáciu a odstúpili ľavú rekurziu. Jednotlivé pravidlá sme očísloval, aby bolo možné zostaviť zrozumiteľnú tabuľku prechodov v 5. časti.

1. $W \rightarrow V Q$

2. $Q \rightarrow \text{“;”} V Q$

3. $Q \rightarrow \epsilon$

4. $V \rightarrow H$

5. $V \rightarrow F$

6. $V \rightarrow T$

7. $V \rightarrow M$
 8. $H \rightarrow \text{"http://"} O \alpha \beta$
 9. $\alpha \rightarrow \text{"/"}$ C
 10. $\alpha \rightarrow \varepsilon$
 11. $\beta \rightarrow \text{"?"}$ S
 12. $\beta \rightarrow \varepsilon$
 13. $O \rightarrow N \gamma$
 14. $\gamma \rightarrow \text{"."}$ I
 15. $\gamma \rightarrow \varepsilon$
 16. $N \rightarrow X \delta$
 17. $\delta \rightarrow \text{"."}$ X δ
 18. $\delta \rightarrow \varepsilon$
 19. $X \rightarrow Y X$
 20. $X \rightarrow \varepsilon$
 21. $Y \rightarrow A$
 22. $Y \rightarrow D$
 23. $I \rightarrow D I$
 24. $I \rightarrow \varepsilon$
 25. $C \rightarrow G \pi$
 26. $\pi \rightarrow \text{"/"}$ G π
 27. $\pi \rightarrow \varepsilon$
 28. $G \rightarrow Y G$
 29. $G \rightarrow \varepsilon$
 30. $S \rightarrow X \Omega$
 31. $\Omega \rightarrow \text{"+"}$ X Ω
 32. $\Omega \rightarrow \varepsilon$
 33. $F \rightarrow \text{"ftp://"} L \text{"/"}$ C
 34. $L \rightarrow \Sigma O$
 35. $\Sigma \rightarrow X \Theta \text{"@"}$
 36. $\Sigma \rightarrow \varepsilon$
 37. $\Theta \rightarrow \text{"."}$ X
 38. $\Theta \rightarrow \varepsilon$
 39. $T \rightarrow \text{"telnet://"} L$
 40. $M \rightarrow \text{"mailto:"} X \text{"@"}$ N
 41. $A \rightarrow \text{"A"} | \dots | \text{"Z"} | \text{"a"} | \dots | \text{"z"}$
 42. $D \rightarrow \text{"0"} | \dots | \text{"9"}$
- (A) $L \rightarrow X \mu$
 - (B) $\mu \rightarrow \varphi$
 - (C) $\mu \rightarrow \varepsilon$
 - (D) $\varphi \rightarrow \text{"."}$ N χ
 - (E) $\varphi \rightarrow \text{":"}$ ψ
 - (F) $\varphi \rightarrow \text{"@"}$ O
 - (G) $\chi \rightarrow \text{"."}$ I
 - (H) $\chi \rightarrow \varepsilon$
 - (I) $\psi \rightarrow A G \text{"@"}$ O
 - (J) $\psi \rightarrow I \rho$
 - (K) $\rho \rightarrow A G \text{"@"}$ O
 - (L) $\rho \rightarrow \text{"@"}$ O
 - (M) $\rho \rightarrow \varepsilon$

Keďže pravidlá 34 až 38 vedú k nedeterministickému automatu, navrhli som alternatívne pravidlá (A) až (M), ktoré vedú k deterministickému automatu (je to zrejme aj z tabuľky prechodov v časti 5).

4. Množiny FIRST a FOLLOW

	FIRST	FOLLOW
W	http:// ftp:// telnet:// mailto:	ϵ
Q	; ϵ	ϵ
V	http:// ftp:// telnet:// mailto:	; ϵ
H	http://	; ϵ
α	/ ϵ	? ; ϵ
β	? ϵ	; ϵ
O	A..z 0..9 : ϵ	/ ? ; ϵ
γ	: ϵ	/ ? ; ϵ
N	A..z 0..9 . ϵ	: / ? ; ϵ
δ	. ϵ	: / ? ; ϵ
X	A..z 0..9 ϵ	. + : @ / ? ; ϵ
Y	A..z 0..9	A..z 0..9 . + : @ / ? ; ϵ
I	0..9 ϵ	/ ? ; ϵ
C	A..z 0..9 / ϵ	? ; ϵ
π	/ ϵ	? ; ϵ
G	A..z 0..9 ϵ	/ ? ; ϵ
S	A..z 0..9 + ϵ	; ϵ
Ω	+ ϵ	; ϵ
F	ftp://	; ϵ
L	A..z 0..9 : ϵ	/ ; ϵ
ϕ	. : @	/ ; ϵ
μ	. : @ ϵ	/ ; ϵ
χ	: ϵ	/ ; ϵ
ψ	A..z 0..9 @ ϵ	/ ; ϵ
ρ	A..z @ ϵ	/ ; ϵ
T	telnet://	; ϵ
M	mailto:	; ϵ
A	A..z	všetky terminály
D	0..9	všetky terminály

Implementácia

6. a 7. Implementácia lexikálneho analyzátora a vytvorenie tabuľkou riadeného syntaktického analyzátora

Lexikálny analyzátor je implementovaný funkciou `lexer()`;, ktorá prekonvertuje vstupný reťazec na pole tokenov. Ak sa niekde v lexikálnej analýze vyskytne chyba tak sa program automaticky ukončí a vypíše sa pozícia a znak, pri ktorom nastala táto chyba.

Funkcie použité pri implemetácii syntaktického analyzátora:

`void syntax(Token *t)`

- ako vstup dostane lexikálnu jednotku (token) a podľa tabuľky prechodov overí syntaktickú správnosť

`void checkToken(int token)`

- testuje terminály a ak je správny, posunie sa na ďalší

Ak pri syntaktickej analýze nastane chyba, program vypíše “syntax error” a ukončí sa.

8. Testovanie na vetách jazyka

Implementovaný tabuľkou riadený syntaktický analyzátor bol testovaný na vetách uvedených v časti 1.

Vytvorený program spĺňa požiadavky zadania, keďže testovanie bolo úspešné pre správne i nesprávne zadané vety kontrolovaného jazyka.

Podiel práce autorov je rovnaký na oboch častiach zadania.