Θεωρία Υπολογισμού και Πολυπλοκότητα

Κεφάλαιο 7.
Κατηγορηματικές Γραμματικές

7 Μαρτίου 2008

Δρ. Παπαδοπούλου Βίκη

1. Απόδειξη ότι μια μη κανονική γλώσσα είναι κατηγορηματική

Παράδειγμα. $Pal = \{ w \in \{ 0, 1 \}^* \mid w = w^R.$

Να δείξετε ότι L(G)=Pal,

$G = \langle \{ S \}, \quad \{ 0, 1 \}, \quad \{ S \rightarrow 0S0 \mid 1S1 \mid \varepsilon \mid 0 \mid 1 \}, S \rangle$

- Πρώτο σκέλος: Θα δείξουμε $Pal \subseteq L(G)$.
  - Για κάθε $w \in Pal$ θα δείξω με επαγωγή πάνω στο μήκος $|w|$,ότι $S \Rightarrow^*_G w$.
  - Βάση επαγωγής: Για κάθε λέξη $w \in Pal$ με μήκος $|w|=1$, $S \Rightarrow^*_G w$.
  - Επαγωγική Υπόθεση: Για κάποιο ακέραιο $n \geq 1$,για κάθε λέξη $w \in Pal$ με μήκος $|w| \leq n$, $S \Rightarrow^*_G w$.
  - Επαγωγικό Βήμα: Έστω $w \in Pal$ με μήκος $|w| \geq n+1 \geq 2$.
    - Τότε $w = 0u0$ ή $w = 1u1$ για κάποιο $u \in Pal$ με μήκος $|u| = n-1$.
    - Από Επαγωγική Υπόθεση, $S \Rightarrow^*_G u$.
    - Από κανόνα $S \rightarrow 0S0 \mid 1S1$ έπεται ότι $S \Rightarrow^*_G 0u0$ και $S \Rightarrow^*_G 1u1$.
    - Έπεται ότι $S \Rightarrow^*_G w$.

1. Απόδειξη ότι μια μη κανονική γλώσσα είναι κατηγορηματική

- 2ο σκέλος: Θα δείξουμε $\subseteq$ Pal.
  - Για κάθε $w \in L(G)$, ($S \Rightarrow^*_G w$) θα δείξω με επαγωγή πάνω στο μήκος $|w|$ ότι $w \in$ Pal.
  - Βάση επαγωγής: Για κάθε λέξη $w \in L(G)$ με μήκος $|w|=1$, $w \in$ Pal.
  - Επαγωγική Υπόθεση: Για κάποιο ακέραιο $n \geq 1$,για κάθε λέξη $w \in \{0,1\}^*$ με μήκος $|w| \leq n$ ώστε $S \Rightarrow^*_G w$, ισχύει $w \in$ Pal.
  - Επαγωγικό Βήμα: Έστω $w \in w \in \{0,1\}^*$ με $|w| \geq n+1 \geq 2$ ώστε $S \Rightarrow^*_G w$.
    - Αφού $|w| \geq 2$, πρώτα εφαρμόστηκε ο κανόνας $S \rightarrow 0S0$ ή $S \rightarrow 0S1$
    - Κατόπιν, εφαρμόστηκε $S \Rightarrow^*_G u$, με $|u| = n\text{-}1$, έτσι ώστε $w = 0u0$ ή $w = 1u1$.
    - Από Επαγωγική Υπόθεση, $u \in$ Pal.
    - Έπεται ότι $u \in$ Pal

2 Θεώρημα Άντλησης: Δείξτε ότι μια γλώσσα είναι μη κανονική.

$L_1=\{w\in\{0,1\}^* \mid$ υπάρχει πρόθεμα u της w όπου $\#_1(u)=2\#_0(u)+4\}$

- Έστω L_1 κανονική
- Κάθε λέξη της μορφής $0^k 1^{2k+4} \in L_1$, όπου $k \geq 0$.
- Θεώρημα Άντλησης $\Rightarrow$ Για κάθε $w \in L_1$ με $|w| \geq M$,
  - υπάρχουν x, y, z, ώστε $w = x\, y\, z$, $|xy| \leq M$, $y \neq e$,

    $\Rightarrow \forall\, n \geq 0,\ x\, y^n z \in L_1$,
- Επιλέγω $w= 0^M 1^{2M+4}$. Ισχύει $\#_1(u)=2\#_0(u)+4$, για $w = u$.
- Προφανώς , $|w| = M + 2M + 4 = 3M + 4 > M$.

  $\Rightarrow$ Το Θεώρημα Άντλησης ισχύει για την w.

  $\Rightarrow$ $x = 0^i$, $0 \leq i < M$

  $\Rightarrow$ $x = 0^j$, $0 < j \leq M$

  $\Rightarrow$ $z = 0^{M-i-j}\, 1^{2M+4}$

  $\Rightarrow$ Επιλέγω n=2. Τότε $x\, y^2 z \in L_1$.

  $\Rightarrow$ Τότε $x\, y^2 z = 0^i\, 0^{2j}\, 0^{M-i-j}\, 1^{2M+4} = 0^{M+j}\, 1^{2M+4} \in L_1$.

  $\Rightarrow$ Αντίφαση.

3 Θεώρημα Άντλησης: Δείξτε ότι μια γλώσσα είναι μη κανονική.

$L_2=\{w\in\{0,1\}^* |$ σε κάθε πρόθεμα u της w με $|w|\geq 4$, $|\#_0(u)-2\#_1(u)\geq 2\}$

- Έστω L_2 κανονική
- Κάθε λέξη της μορφής $0^{4+k} \in L_2$, όπου $k \geq 0$.
- Θεώρημα Άντλησης $\Rightarrow$ Για κάθε $w \in L_2$ με $|w| \geq M$,
  - υπάρχουν x, y, z, ώστε $w = x\,y\,z$, $|xy| \leq M$, $y \neq e$,

    $\Rightarrow \forall\, n\geq 0, x\, y^{n}\, z \in L_2$,
- Επιλέγω $w= 0^{M+2}\, 1^{M}$. $w \in L_2$, αφού σε κάθε πρόθεμα u της w με $|u|\geq 4$, $\#_0(u)-2\#_1(u)\geq 2$ (περιπτώσεις u περιέχει μόνο 0 ή 0 και 1)
- Επιπλέον, $|w| = M + 2 + M = 2M + 2 > M$.

  $\Rightarrow$ Το Θεώρημα Άντλησης ισχύει για την w.

  $\Rightarrow$ $x = 0^{i}$, $0\leq i < M$

  $\Rightarrow$ $x = 0^{j}$, $0 < j \leq M$

  $\Rightarrow$ $z = 0^{M+2-i-j}\, 1^{M}$

  $\Rightarrow$ Επιλέγω n=0. Τότε $x\, y^{0}\, z \in L_2$.

  $\Rightarrow$ Τότε $x\, y^{0}\, z = 0^{i}\, 0^{M+2-i-j}\, 1^{M} = 0^{M+2-j}\, 1^{M} \in L_2$.

  Δείχνω ότι για j\leq 2 ή ξ>2 δεν ικανοποιείται η συνθήκη για να ανήκει $x\, y^{0}\, z$ στη γλώσσα.

4.Κατασκευή γραμματικής χωρίς συμφραζόμενα

. Ένα πρόγραμμα σε μια πραγματική γλώσσα προγραμματισμού, όπως η C ή Pascal, αποτελείται από εντολές, όπου κάθε **εντολή** ανήκει σε έναν από τους εξής τύπους:

(a) Εντολή εκχώρησης: μορφής $id := E$, E είναι οποιαδήποτε αριθμητική έκφραση (όπως παράγεται από την γραμματική της άσκησης 2).

(b) Εντολή συνθήκης: της μορφής If $E < E$ then **εντολή** ή της μορφής while $E < E$ do **εντολή**.

(c) Εντολή goto label · πριν από κάθε εντολή μπορεί να προηγηθεί μια επιγραφή (label).

(d) Σύνθετη εντολή, δηλ. πολλές εντολές πριν από τις οποίες προηγείται ένα begin, χωρίζονται μεταξύ τους με ένα ";" και σύνθετη εντολή τελείωνει με ένα end.

Δώστε μια γραματική χωρίς συμφραζόμενα που να παράγει όλες τις δυνατές εντολές στην απλοποιημένη γλώσσα προγραμματισμού παραπάνω.

4.Κατασκευή γραμματικής χωρίς συμφραζόμενα

G = (V,Σ,R,S) όπου

V = {:= ,<, label: , ; ,if, then, while, do, begin, end, (,),+,*,id, E,S,M}

Σ = {:= ,<, label: , ; ,if, then, while, do, begin, end, (,),+,*,id }

R= { $S \to id := E,$

$S \to$ if E < E then S,

$S \to$ while E < E do S,

$S \to$ goto lbl,

$S \to$ begin M end,

$S \to$ lbl: S,

$M \to$ S; M,

$E \to (E+E)\,|\,(E*E)\,|\,id$ }

5.Κατασκευή γραμματικής χωρίς συμφραζόμενα για μια γλώσσα

- Δείξτε ότι η γλώσσα $a^m b^n c^p$: m+n = p} παράγεται από μια γραμματική χωρίς συμφραζόμενα κατασκευάζοντας μια τέτοια γραμματική για τη γλώσσα.

G = (V,Σ,R,S) όπου

V = {α,b, A, B, S}

Σ = {a,b}

$$
\begin{aligned}
R = \{ \quad & S \rightarrow A \;|B\; |e, \\
& A \rightarrow aAc \;|B\; |e, \\
& B \rightarrow bBc \;|e, \\
\}
\end{aligned}
$$