T-79.1001/2 **Syksy 2005**
Tietojenkäsittelyteorian perusteet (T/Y)
Harjoitus 5
Demonstraatiotehtävien ratkaisut

4. **Tehtävä**: Osoita, että yhteydettömien kielten luokka on suljettu yhdiste-, katenaatio- ja sulkeumaoperaatioiden suhteen, so. jos kielet $L_1, L_2 \subseteq \Sigma^*$ ovat yhteydettömiä, niin samoin ovat myös kielet $L_1 \cup L_2$, L_1L_2 ja L_1^*.

   Vastaus: Olkoon L_1 ja L_2 yhteydettömiä kieliä. Tällöin on olemassa kieliopit $G_1 = (V_1, \Sigma_1, R_1, S_1)$ ja $G_2 = (V_2, \Sigma_2, R_2, S_2)$, siten, että $L(G_1) = L_1$ ja $L(G_2) = L_2$. Vaaditaan lisäksi, että $(V_1 - \Sigma_1) \cap (V_2 - \Sigma_2) = \emptyset$, eli kieliopeissa ei esiinny samoja välikkeitä. Koska kieliopin välikkeet voidaan tarvittaessa nimetä uudelleen, ei tämä aseta oleellista rajoitusta.

   Unioni: Olkoon S uusi välike ja $G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \to S_1 \mid S_2\}, S\}$. Nyt $L(G) = L(G_1) \cup L(G_2) = L_1 \cup L_2$. Näin on, koska S:stä voidaan johtaa vain S_1 tai S_2, joista voidaan edelleen johtaa vain sanoja jotka kuuluvat jompaan kumpaan aiemmista kielistä (sääntöjen sekaannukselta vältytään, koska välikejoukot ovat pistevieraita).

   Katenaatio: Tällä kertaa uusi kielioppi $G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \to S_1S_2\}, S\}$. Nyt $L(G) = L_1L_2$.

   Kleenen tähti: Tällä kertaa uusi kielioppi $G = (V_1 \cup \{S\}, \Sigma_1, R_1 \cup \{S \to \epsilon \mid SS_1\}, S\}$. Nyt $L(G) = L_1^*$

5. **Tehtävä**: Laadi yhteydetön kielioppi, joka tuottaa kaikki seuraavan esimerkin tapaiset, yksinkertaisista sisäkkäisistä `for`-silmukoista, `begin`- ja `end`-sulkeilla kootuista lauseista ja alkeisoperaatioista `a` rakentuvat "ohjelmat":

```
a;
for 3 times do
begin
  for 5 times do a;
  a; a
end
```

   Silmukkalaskureiden voit olettaa olevan kokonaislukuja väliltä 0,..., 9.

   Vastaus: Ohjelmointikielten kieliopit määritellään useimmiten siten, että aakkostoksi otetaan kielessä esiintyvät syntaktiset elementit (lekseemit). Tässä tapauksessa niitä ovat numerot, `a` sekä varatut sanat. Ohjelman jäsentäminen jaetaan kahteen osaan:

   1. Muutetaan ohjelman teksti jonoksi lekseemeitä tilakoneiden avulla.
   2. Muodostetaan lekseemijonon jäsennyspuu.

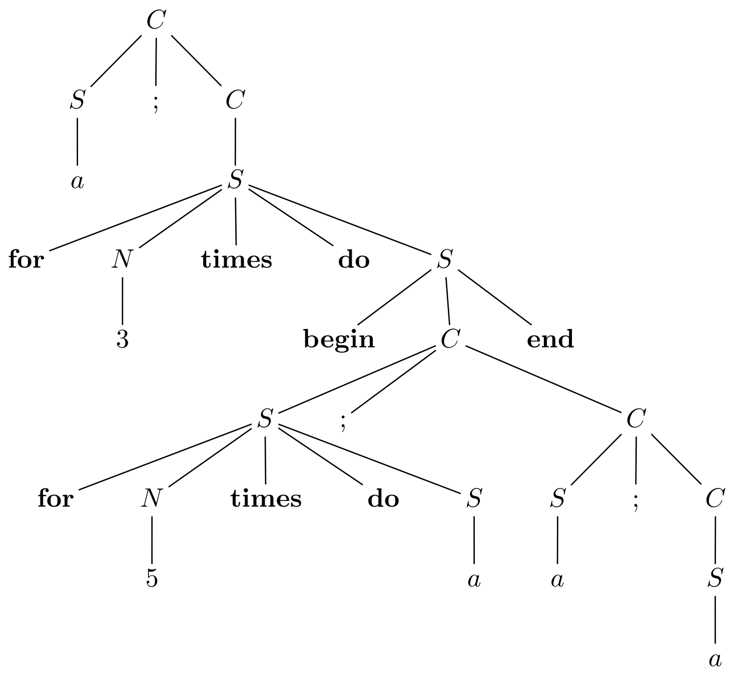
   Tehtävän kieliopin voi määritellä monellakin eri tapaa, tässä on yksi mahdollinen tulkinta:

$$
\begin{aligned}
G =& (V, \Sigma, P, C) \\
V =& \{C, S, N, \mathbf{begin}, \mathbf{do}, \mathbf{end}, \mathbf{for}, \mathbf{times}, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ;, a\} \\
\Sigma =& \{\mathbf{begin}, \mathbf{do}, \mathbf{end}, \mathbf{for}, \mathbf{times}, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, ;, a\}
\end{aligned}
$$

   Tässä välikkeen S tulkintana on "lause" (*statement*), C:n "yhdistetty lause" (*compound statement*) ja N:n "numero". Kieliopin säännöt määritellään seuraavasti:

$$
\begin{aligned}
P = \{C &\to S \mid S; C \\
S &\to a \mid \mathbf{begin}\ C\ \mathbf{end} \mid \mathbf{for}\ N\ \mathbf{times\ do}\ S \\
N &\to 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9\}
\end{aligned}
$$

Esimerkki. Tehtävänannossa esiintyneen ohjelman jäsennyspuu:



6. **Tehtävä**:

1. Osoita, että seuraava yhteydetön kielioppi on moniselitteinen:

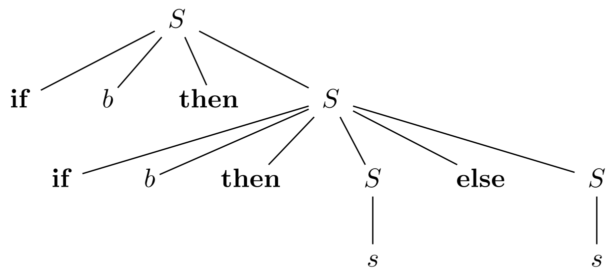
$$
\begin{aligned}
S &\rightarrow \textbf{if } b \textbf{ then } S \\
S &\rightarrow \textbf{if } b \textbf{ then } S \textbf{ else } S \\
S &\rightarrow s.
\end{aligned}
$$

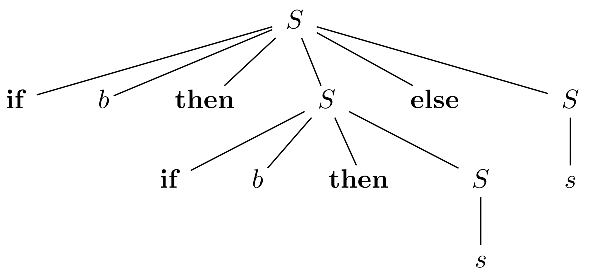
2. Muodosta (a)-kohdan kieliopin kanssa ekvivalentti, so. saman kielen tuottava yksiselitteinen kielioppi.

Vastaus: Yhteydetön kielioppi G on moniselitteinen, mikäli on olemassa sana $w \in L(G)$ siten, että w:llä on ainakin kaksi erilaista jäsennyspuuta. Tehtävän kieliopille yksinkertaisin tällainen sana on:

$$\textbf{if } b \textbf{ then if } b \textbf{ then } s \textbf{ else } s,$$

joka voidaan jäsentää kahdella tapaa:





Yleensä ohjelmointikielissä halutaan **else**-lause liittää lähinpään mahdolliseen **if**-lauseeseen. Ylläolevista puista ensimmäinen vastaa tätä käytäntöä.

Määritellään kielioppi seuraavasti:

$$
\begin{aligned}
&G = (V, \Sigma, P, S)\\
&V = \{S, B, U, s, b, \mathbf{if}, \mathbf{then}, \mathbf{else}\}\\
&\Sigma = \{s, b, \mathbf{if}, \mathbf{then}, \mathbf{else}\}\\
&P = \{S \to B \mid U\\
&\qquad B \to \mathbf{if}\ b\ \mathbf{then}\ B\ \mathbf{else}\ B\ \mid s\\
&\qquad U \to \mathbf{if}\ b\ \mathbf{then}\ S \mid \mathbf{if}\ b\ \mathbf{then}\ B\ \mathbf{else}\ U\}
\end{aligned}
$$

Tässä välikeellä B saadaan johdettua vain ohjelmia, joissa kaikilla **if**-lauseilla on sekä **then**- että **else**-haarat. Välikkeellä U johdetaan sitten **if**-lauseet, joista puuttuu **else**-haara.