T-79.1001/2 **Kevät 2006**
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 $L_1 \cup L_2$, $L_1 L_2$ 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 $G = (V_1 \cup V_2 \cup \{S\}, \Sigma_1 \cup \Sigma_2, R_1 \cup R_2 \cup \{S \to S_1 S_2\}, S\}$. Nyt $L(G) = L_1 L_2$.

   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.

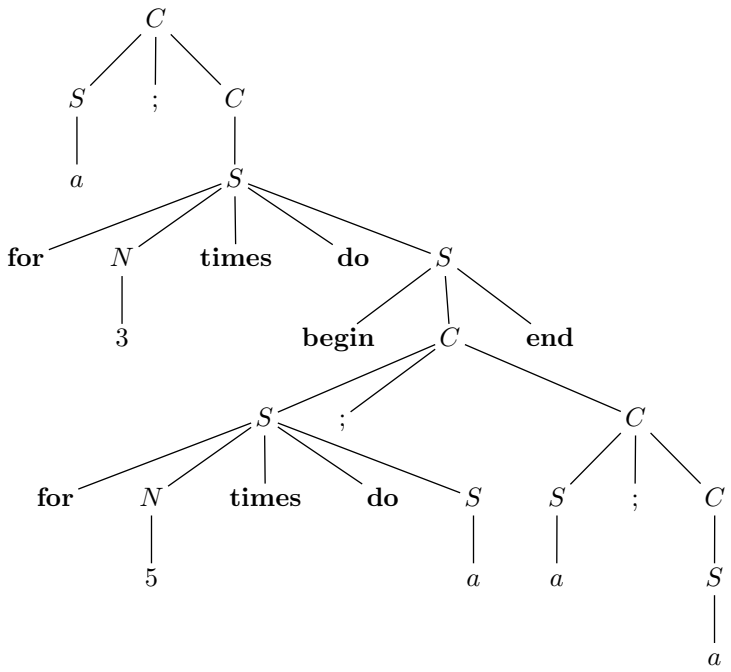
   Tehtävän kieliopin voi määritellä monellakin eri tapaa, tässä on yksi mahdollinen tulkinta:

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

   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**:Modernissa WWW-sivujen kuvaamiseen käytetyssä XML-kielessä on sivujen suunnittelijan mahdollista laatia omia ns. dokumenttityyppimäärityksiä (engl. *Document Type Definition*, lyh. DTD), jotka ovat oleellisesti sivulla esitettävän tekstin tai muun datan rakennetta kuvaavia yhteydettömiä kielioppeja. Tutustu tämän XML/DTD-kuvauskielen notaatioon (esim. WWW-sivulta `http://www.rpbourret.com/xml/xmldtd.htm`), ja laadi seuraavaa XML/DTD-kuvausta vastaava yhteydetön kielioppi:

```
<!DOCTYPE Book [
   <!ELEMENT Book (Title, Chapter+)>
   <!ATTLIST Book Author CDATA #REQUIRED>
   <!ELEMENT Title (#PCDATA)>
   <!ELEMENT Chapter (#PCDATA)>
   <!ATTLIST Chapter id ID #REQUIRED>
]>
```

Vastaus: Yllä oleva DTD-kuvaus määrittelee rakenteen kirjalle. Määrittelyssä esiintyy kahdenlaisia asioita: *osia* (element) ja *attribuutteja* (attlist). Perusajatuksena on, että kirja itsessään koostuu osista, ja attribuutit puolestaan liittävät kirjan osiin ylimääräistä tietoa.

Yleisesti ottaen attribuutteja ei voida esittää puhtailla kieliopeilla, vaan niitä varten tarvitaan *attribuuttikieliopit* (ks. opetusmoniste s. 60–68). Näin ollen DTD-kuvauksesta mallinnetaan ensin vain osat, eli käytännössä ainoastaan rivit, jotka alkavat merkinnällä `!ELEMENT`.

Näistä riveistä ensimmäinen:

```
<!ELEMENT Book (Title, Chapter+)>
```

kertoo, että kirja (Book) sisältää otsikon (Title) ja listan lukuja (Chapter). Lukuja täytyy olla vähintään yksi. Seuraava rivi:

```
<!ELEMENT Title (#PCDATA)>
```

puolestaan määrittelee otsikon merkkijoukoksi (#PCDATA). Merkkijoukkoon voi kuulua periaattessa mitä tahansa tietokoneen merkkijärjestelmään kuuluvia symboleita.

Lopuksi, rivi:

```
<!ELEMENT Chapter (#PCDATA)>
```

kertoo luvun olevan taas joukko merkkejä.

Kirjan rakenteen kuvaa siis kielioppi:[1]

$$\begin{aligned} \mathit{Book} &\rightarrow \mathit{Title}\ \mathit{Chapters} \\ \mathit{Title} &\rightarrow \mathbf{data} \\ \mathit{Chapters} &\rightarrow \mathit{Chapter}\ \mathit{Chapters} \mid \mathit{Chapter} \\ \mathit{Chapter} &\rightarrow \mathbf{data} \end{aligned}$$

XML-kielessä erotetaan dokumentin osat toisistaan käyttäen apuna `<A>` ja `</A>` koodeja. Kun nämä koodit lisätään yllä olevaan kielioppiin, saadaan tulokseksi seuraavanlainen yhteydetön kielioppi:

$$\begin{aligned} \mathit{Book} &\rightarrow \langle\mathbf{Book}\rangle\ \mathit{Title}\ \mathit{Chapters}\ \langle\mathbf{/Book}\rangle \\ \mathit{Title} &\rightarrow \langle\mathbf{Title}\rangle\ \mathbf{data}\ \langle\mathbf{/Title}\rangle \\ \mathit{Chapters} &\rightarrow \mathit{Chapter}\ \mathit{Chapters} \mid \mathit{Chapter} \\ \mathit{Chapter} &\rightarrow \langle\mathbf{Chapter}\rangle\ \mathbf{data}\ \langle\mathbf{/Chapter}\rangle \end{aligned}$$

Vaikka attribuutteja ei voidakaan täysin esittää yhteydettömällä kieliopilla, niiden syntaksi voidaan kuitenkin kuvata. XML-kielessä osan attribuutit kirjoitetaan osan aloittavan koodin sisään. Lisäämällä nämä ylläolevaan kielioppiin saadaan tulokseksi:

$$\begin{aligned} \mathit{Book} &\rightarrow \langle\ \mathbf{Book}\ \mathit{BookAttributes}\ \rangle\ \mathit{Title}\ \mathit{Chapters}\ \langle\mathbf{/Book}\rangle \\ \mathit{Title} &\rightarrow \langle\mathbf{Title}\rangle\ \mathbf{data}\ \langle\mathbf{/Title}\rangle \\ \mathit{Chapters} &\rightarrow \mathit{Chapter}\ \mathit{Chapters} \mid \mathit{Chapter} \\ \mathit{Chapter} &\rightarrow \langle\ \mathbf{Chapter}\ \mathit{ChapterAttributes}\ \rangle\ \mathbf{data}\ \langle\mathbf{/Chapter}\rangle \\ \mathit{BookAttributes} &\rightarrow \mathbf{author = data} \\ \mathit{ChapterAttributes} &\rightarrow \mathbf{id = data} \end{aligned}$$

Tässä on huomattava, että yhteydettömällä kieliopilla ei voida määritellä ehtoa, jonka mukaan kaikilla luvuilla on eri tunnus. Tällaisten ehtojen toteutuminen täytyy tarkistaa jollain erillisellä ohjelmakoodilla.

Liite: oikealle lineaariset kieliopit

Yhteydetön kielioppi G on oikealle lineaarinen, mikäli sen kaikki säännöt ovat muotoa:

$$A \rightarrow \alpha B,$$

missä $\alpha \in \Sigma^*$ ja $B \in V \cup \{\varepsilon\}$. Toisin sanoen, säännön oikealla puolella saa esiintyä korkeintaan yksi välike, ja sen täytyy olla säännön lopussa. Esimerkiksi säännöt $A \rightarrow abC$ ja $A \rightarrow \varepsilon$ ovat oikealle lineaarisia, mutta $A \rightarrow Ba$ ja $A \rightarrow abCD$ eivät ole. Vastaavasti vasemmalle lineaarisen kieliopin kaikki säännöt ovat muotoa $A \rightarrow B\alpha$.

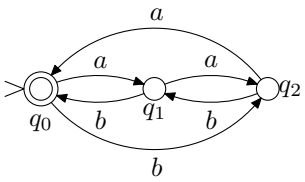
Lineaariset kieliopit ovat ilmaisuvoimaltaan samalla tasolla kuin äärelliset automaatit, eli niillä voidaan ilmaista kaikki säännölliset kielet.

Säännöllisen kielen tuottava oikealle lineaarinen kielioppi voidaan lukea suoraan kielen tunnistavasta automaatista. Kieliopin välikkeiksi otetaan automaatin tilat, ja kustakin

[1] *Kursiivilla* kirjoitetut sanat ovat välikkeitä, **vahvennetut** päätemerkkejä.

automaatin siirtymästä $q_i \xrightarrow{a} q_j$ saadaan sääntö $Q_i \to aQ_j$. Lisäksi kielioppiin lisätään säännöt $Q_f \to \varepsilon$ kaikille hyväksyville lopputiloille $q_f \in F$.

Esimerkiksi automaattia:



vastaava kielioppi on:

$$
\begin{aligned}
Q_0 &\to aQ_1 \mid bQ_2 \mid \varepsilon \\
Q_1 &\to aQ_2 \mid bQ_0 \\
Q_2 &\to aQ_0 \mid bQ_1 \ .
\end{aligned}
$$