Yhteydettömät kieliopit [Sipser luku 2.1]

Johdantoesimerkkinä tarkastelemme kieltä $L = \{\, \mathrm{a}^n\mathrm{b}^m\mathrm{a}^n \mid n > 0, m > 0 \,\}$, joka on yhteydetön (mutta ei säännöllinen). Vastaavan kieliopin ytimenä on *säännöt* eli *produktiot*

$$\begin{aligned} S &\rightarrow \mathrm{a}S\mathrm{a} \\ S &\rightarrow \mathrm{a}B\mathrm{a} \\ B &\rightarrow \mathrm{b}B \\ B &\rightarrow \mathrm{b}. \end{aligned}$$

Kieliopin *muuttujat* eli *välikkeet* (variable, nonterminal symbol) ovat S ja B. Näistä S on erityisasemassa *lähtösymbolina*. *Päätemerkit* eli -symbolit (terminal) ovat a ja b. Kieliopista saadaan esim. merkkijonolle aabbaa $\in L$ seuraava *johto* (derivation):

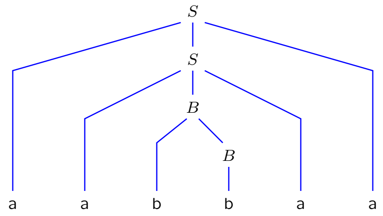
$$S \Rightarrow \mathrm{a}S\mathrm{a} \Rightarrow \mathrm{aa}B\mathrm{aa} \Rightarrow \mathrm{aab}B\mathrm{aa} \Rightarrow \mathrm{aabbaa}.$$

Johdossa siis aloitetaan lähtösymbolista ja korvataan muuttujat yksi kerrallaan vastaavan produktion oikealla puolella olevalla merkkijonolla.

Merkkijonon aabbaa johtoa

$$S \Rightarrow \mathrm{a}S\mathrm{a} \Rightarrow \mathrm{aa}B\mathrm{aa} \Rightarrow \mathrm{aab}B\mathrm{aa} \Rightarrow \mathrm{aabbaa}$$

vastaa seuraava *jäsennyspuu* (parse tree):



Lehtinä on johdetun merkkijonon merkit esijärjestyksessä ("vasemmalta oikealle"), juurena lähtösymboli ja sisäsolmuina välikkeitä. Kunkin välikkeen lapset vastaavat välikeeseen liittyvän säännön oikeaa puolta.

Muodollisesti *yhteydetön* eli *kontekstiton kielioppi* (context-free grammar, CFG) on nelikko (V, Σ, R, S), missä

1. V on äärellinen *muuttujien* (eli muuttujasymbolien eli välikesymbolien) joukko
2. Σ on äärellinen *päätesymbolien* joukko, joka ei saa sisältää muuttujasymboleja
3. R on äärellinen joukko *sääntöjä* eli *produktioita* muotoa $A \rightarrow w$, missä $A \in V$ on muuttuja ja $w \in (V \cup \Sigma)^*$ on mielivaltainen (mahdollisesti tyhjä) jono muuttujia ja päätesymboleja
4. jokin muuttuja $S \in V$ on asetettu *lähtösymboliksi* .

Esimerkki 2.1: Edellä todettiin, että kielelle $\{\, \mathrm{a}^n\mathrm{b}^m\mathrm{a}^n \mid n, m > 0 \,\}$ on olemassa yhteydetön kielioppi $G = (V, \Sigma, R, S)$, missä

- $V = \{\, S, B \,\}$
- $\Sigma = \{\, \mathrm{a}, \mathrm{b} \,\}$
- $R = \{\, S \rightarrow \mathrm{a}S\mathrm{a}\ ,\ S \rightarrow \mathrm{a}B\mathrm{a}\ ,\ B \rightarrow \mathrm{b}B\ ,\ B \rightarrow \mathrm{b} \,\}$
- S on lähtösymboli.

Jatkossa esittämme tämän kieliopin yksinkertaisesti muodossa

$$\begin{aligned} S &\rightarrow \mathrm{a}S\mathrm{a} \mid \mathrm{a}B\mathrm{a} \\ B &\rightarrow \mathrm{b}B \mid \mathrm{b} \end{aligned}$$

Siis

- muuttujia ovat ne, jotka ovat jonkin säännön vasempana puolena,
- lähtösymboli on ensimmäisenä listatun säännön vasen puoli
- muut symbolit ovat päätesymboleita.
- samaan muuttujaan liittyvät säännöt kootaan yhteen ja niiden oikeat puolet erotetaan pystyviivalla toisistaan. □

Intuitiivisesti yhteydetön kielioppi $G = (V, \Sigma, R, S)$ tuottaa aakkoston Σ merkkijonoja seuraavalla proseduurilla:

1. Alusta $r := S$. (Jatkossa r saa arvokseen aakkoston $V \cup \Sigma$ merkkijonoja.)
2. Valitse merkkijonosta r jokin kohta i, jossa esiintyy muuttujasymboli; ts. olkoon
$$r = r_1 \dots r_{i-1}\, A\, r_{i+1} \dots r_n,$$
missä $n = |r|$, $r_i \in V \cup \Sigma$ kaikilla i ja $A \in V$.
3. Valitse sääntöjoukosta R jonkin sääntö $A \to w_1 \dots w_m$, jonka vasen puoli on sama kuin merkkijonosta r valittu muuttujasymboli.
4. Aseta muuttujalle r uusi arvo
$$r := r_1 \dots r_{i-1}\, w_1 \dots w_m\, r_{i+1} \dots r_k.$$
5. Jos r sisältää pelkkiä päätemerkkejä (eli $r \in \Sigma^*$), niin tulosta r. Muuten palaa kohtaan 2.

Kieliopin tuottama kieli koostuu niistä aakkoston Σ merkkijonoista, jotka tämä proseduuri voi tulostaa, kun sovellettavat säännöt valitaan sopivasti.

Määritellään nyt täsmällisemmin kieliopin $G = (V, \Sigma, R, S)$ tuottama kieli.

Jos $x = u\,A\,v$ ja $y = u\,w\,v$, missä $A \to w \in R$, niin x *johtaa suoraan* eli *tuottaa suoraan* (yields, derives directly) merkkijonon y. Tätä merkitään
$$x \Rightarrow y\ .$$
Jos on olemassa aakkoston $V \cup \Sigma$ merkkijonot $u_0, \dots, u_k$, missä $u_{i-1} \Rightarrow u_i$ kun $i = 1, \dots, k$, eli
$$u_0 \Rightarrow u_1 \Rightarrow u_2 \Rightarrow \cdots \Rightarrow u_{k-1} \Rightarrow u_k$$
sanotaan, että u_0 *johtaa* eli *tuottaa* (derives) merkkijonon u_k ja merkitään
$$u_0 \overset{*}{\Rightarrow} u_k\ .$$
Tässä k voi olla myös 0: kaikilla $u \in (V \cup \Sigma)^*$, $u \overset{*}{\Rightarrow} u$ eli jokainen merkkijono tuottaa itsensä. Tapauksessa $k > 0$ voidaan merkitä $u_0 \overset{+}{\Rightarrow} u_k$.

Aakkoston $V \cup \Sigma$ merkkijono w on *lausejohdos* (sentential form), jos $S \overset{*}{\Rightarrow} w$. Jos lisäksi w sisältää vain päätesymboleja, se on *lause*. Ketjua
$$S = u_0 \Rightarrow u_1 \Rightarrow \dots \Rightarrow u_k = w$$
sanotaan merkkijonon w *johdoksi* (derivation), ja sen *pituus* on k.

Kieliopin G *tuottama* eli *kuvaama* kieli on sen lauseiden joukko
$$L(G) = \left\{ w \in \Sigma^* \mid S \overset{*}{\Rightarrow} w \right\}.$$
Kieli on *yhteydetön*, jos jokin yhteydetön kielioppi tuottaa sen.

Selitys termille "yhteydetön": "Yhteydetön" viittaa siihen, että kieliopin säännöt ovat muotoa $A \to w$, mikä voidaan tulkita

> Muuttuja A voidaan korvata merkkijonolla w, oli ympärillä mitä tahansa.

Sääntöjä siis voidaan soveltaa "kontekstista riippumatta".

Yhteydettömän kielioppin eräs mahdollinen yleistys on *yhteysherkkä kielioppi* (context-sensitive grammar). Tällaisen kieliopin säännöt ovat muotoa $uAv \to uwv$, missä u, v ja w ovat aakkoston $V \cup \Sigma$ merkkijonoja. Sääntö tulkitaan

> Muuttuja A voidaan korvata merkkijonolla w, jos sen ympärillä on merkkijonot u ja v.

Voidaan osoittaa, että esim. kieli
$$\{\, \mathrm{a}^n\mathrm{b}^n\mathrm{c}^n \mid n \in \mathbb{N} \,\}$$
ei ole yhteydetön, mutta voidaan esittää käyttämällä yhteysherkkiä sääntöjä.

Lausejohdoksen $u \in \Sigma^*$ *jäsennyspuu* on mikä tahansa seuraavat ehdot täyttävä puu:

1. Puun solmut on nimetty joukon $V \cup \Sigma \cup \{\varepsilon\}$ alkioilla.
2. Sisäsolmut on nimetty muuttujilla ja erityisesti juuri lähtösymbolilla S.
3. Jos A on sisäsolmu nimi ja sen lasten nimet vasemmalta oikealle ovat $w_1, \ldots, w_m$, niin kieliopissa on sääntö $A \to w_1 \ldots w_m$.
4. Lehdissä ovat merkkijonon u merkit järjestyksessä vasemmalta oikealle sekä mahdollisesti tyhjän merkkijonon symboleja ε.

Merkkijonon u johdosta voidaan rakentaa sille jäsennyspuu:

1. Lähdetään liikkeelle puusta, jossa on pelkkä juuri S.
2. Kun johdossa sovelletaan sääntöä $A \to w_1 \ldots w_m$, niin puussa vastaavalle lehdelle A lisätään lapset $w_1, \ldots, w_m$.

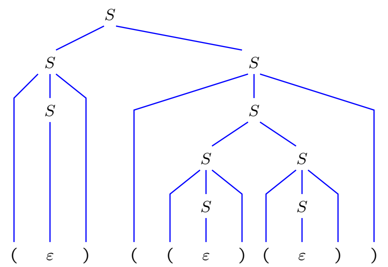
Esimerkki 2.2: Yhteydetön kielioppi

$$S \to SS \mid (\,S\,) \mid \varepsilon$$

tuotaa kaikki oikein muodostetut sulkulausekkeet. Tässä siis päätesymbolit ovat vasen ja oikea sulkumerkki. Sulkulauseke on oikein muodostettu, jos vasemmat ja oikeat sulkumerkit voidaan pariuttaa siten, että mikään pari ei mene ristiin.

Sulkulausekkeen $()(()())$ eräs johto ja sitä vastaava jäsennyspuu ovat

$$\begin{aligned} S &\Rightarrow SS \\ &\Rightarrow (S)S \\ &\Rightarrow ()S \\ &\Rightarrow ()(S) \\ &\Rightarrow ()(SS) \\ &\Rightarrow ()((S)S) \\ &\Rightarrow ()(()S) \\ &\Rightarrow ()(()(S)) \\ &\Rightarrow ()(()()) \end{aligned}$$



□

Yhteydettömien kielioppien laatiminen [Sipser s. 106–107]

Yhteydettömiä kielioppeja on helppo tuottaa soveltamalla vastaavia operaatioita kuin säännöllisissä lausekkeissa.

Lause 2.3: Jos A ja B ovat yhteydettömiä kieliä, niin myös $A \cup B$, $A \circ B$ ja A^* ovat.

Todistus: Tarkastellaan esimerkkinä yhdistettä $A \cup B$; muut kohdat jätetään harjoitustehtäväksi.

Olkoot A ja B aakkoston Σ yhteydettömiä kieliä. Siis $A = L(G_A)$ ja $B = L(G_B)$ joillain yhteydettömillä kieliopeilla $G_A = (V_A, \Sigma, R_A, S_A)$ ja $G_B = (V_B, \Sigma, R_B, S_B)$. Kielioppien muuttujasymboleita voidaan tarvittaessa vaihtaa niin, että $V_A \cap V_B = \emptyset$.

Olkoon $S \notin V_A \cup V_B$. Määritellään $G = (V_A \cup V_B \cup \{S\}, \Sigma, R, S)$, missä

$$R = R_A \cup R_B \cup \{S \to S_A\} \cup \{S \to S_B\}$$

Selvästi $L(G) = A \cup B$. □

Lauseesta 2.3 seuraa erityisesti

Korollaari 2.4: Kaikki säännölliset kielet ovat yhteydettömiä.

Todistus: Olkoon Σ mielivaltainen aakkosto. Määritellään

$$\begin{aligned} G_\emptyset &= (\{S\}, \Sigma, \emptyset, S) \\ G_\varepsilon &= (\{S\}, \Sigma, \{S \to \varepsilon\}, S) \\ G_a &= (\{S\}, \Sigma, \{S \to a\}, S) \text{ kaikilla } a \in \Sigma. \end{aligned}$$

Selvästi $L(G_\emptyset) = \emptyset$, $L(G_\varepsilon) = \{\varepsilon\}$ ja $L(G_a) = \{a\}$ kaikilla $a \in \Sigma$.

Jokainen säännöllinen kieli voidaan esittää säännöllisenä lausekkeena, joka kertoo, miten kyseinen kieli muodostetaan peruskielistä $\emptyset$, $\{\varepsilon\}$ ja $\{a\}$ käyttäen operaatioita $\cup$, $\circ$ ja $*$. Yllä olevasta ja lauseesta 2.3 seuraa, että näille kielille on olemassa myös yhteydetön kielioppi. □

Jatkossa osoittautuu, että yhteydettömien kielten luokka **ei** ole suljettu leikkauksen ja komplementin suhteen, toisin kuin säännöllisten kielten luokka.

Yhteydettömät kielet ovat aidosti laajempi luokka kuin säännölliset, kuten seuraavat perusesimerkit osoittavat.

Esimerkki 2.5: Kieli $\{\, 0^n 1^n \mid n \in \mathbb{N} \,\}$ tunnetusti ei ole säännöllinen. Se voidaan kuitenkin tuottaa yhteydettömällä kieliopilla

$$S \to 0S1 \mid \varepsilon$$

$\square$

Esimerkki 2.6: Aakkoston $\{\, \mathsf{a}, \mathsf{b} \,\}$ palindromien joukko $\{\, w \in \{\, \mathsf{a}, \mathsf{b} \,\}^* \mid w = w^{\mathcal{R}} \,\}$ voidaan tuottaa yhteydettömällä kieliopilla

$$S \to \varepsilon \mid \mathsf{a} \mid \mathsf{b} \mid \mathsf{a}S\mathsf{a} \mid \mathsf{b}S\mathsf{b}.$$

Vastaavasti kieli $\{\, ww^{\mathcal{R}} \mid w \in \{\, \mathsf{a}, \mathsf{b} \,\}^* \,\}$ voidaan tuottaa yhteydettömällä kieliopilla

$$S \to \varepsilon \mid \mathsf{a}S\mathsf{a} \mid \mathsf{b}S\mathsf{b}.$$

Sen sijaan kieli $\{\, ww \mid w \in \{\, \mathsf{a}, \mathsf{b} \,\}^* \,\}$ ei ole yhteydetön, kuten jatkossa näemme. $\square$

Esimerkki 2.7: Luonnollinen kieli on ollut tärkeä motivaatio kielioppiformalismien kehittämiselle. Leikkikaluesimerkki:

$$
\begin{array}{rcl}
\langle\texttt{sentence}\rangle & \to & \langle\texttt{noun-phrase}\rangle\langle\texttt{verb-phrase}\rangle \\
 & & \mid \langle\texttt{noun-phrase}\rangle\langle\texttt{verb}\rangle\langle\texttt{direct-object-phrase}\rangle \\
\langle\texttt{noun-phrase}\rangle & \to & \langle\texttt{proper-noun}\rangle \mid \langle\texttt{determiner}\rangle\langle\texttt{common-noun}\rangle \\
\langle\texttt{proper-noun}\rangle & \to & \textbf{John} \mid \textbf{Jill} \\
\langle\texttt{common-noun}\rangle & \to & \textbf{car} \mid \textbf{hamburger} \\
\langle\texttt{determiner}\rangle & \to & \textbf{a} \mid \textbf{the} \\
\langle\texttt{verb-phrase}\rangle & \to & \langle\texttt{verb}\rangle\langle\texttt{adverb}\rangle \mid \langle\texttt{verb}\rangle \\
\langle\texttt{verb}\rangle & \to & \textbf{drives} \mid \textbf{eats} \\
\langle\texttt{adverb}\rangle & \to & \textbf{slowly} \mid \textbf{frequently} \\
\langle\texttt{direct-object-phrase}\rangle & \to & \langle\texttt{adjective-list}\rangle\langle\texttt{proper-noun}\rangle \\
 & & \mid \langle\texttt{determiner}\rangle\langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
\langle\texttt{adjective-list}\rangle & \to & \varepsilon \mid \langle\texttt{adjective}\rangle\langle\texttt{adjective-list}\rangle \\
\langle\texttt{adjective}\rangle & \to & \textbf{big} \mid \textbf{juicy} \mid \textbf{brown}
\end{array}
$$

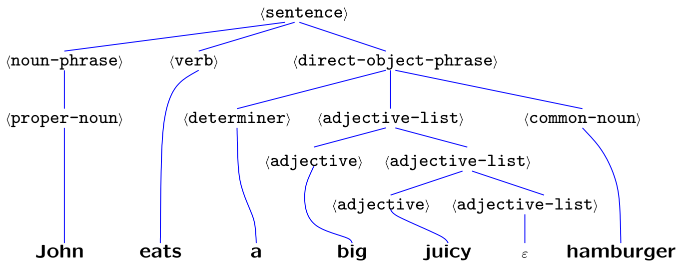
Tässä siis $\langle\texttt{sentence}\rangle$ jne. ovat muuttujasymboleja ja **John** jne. päätesymboleja. Huomaa muuttujaa $\langle\texttt{adjective-list}\rangle$ käyttävä toistorakenne.

Edellisestä kieliopista saadaan esim. seuraavanlaisia johtoja:

$$
\begin{array}{rcl}
\langle\texttt{sentence}\rangle & \Rightarrow & \langle\texttt{noun-phrase}\rangle\langle\texttt{verb}\rangle\langle\texttt{direct-object-phrase}\rangle \\
 & \Rightarrow & \langle\texttt{proper-noun}\rangle\langle\texttt{verb}\rangle\langle\texttt{direct-object-phrase}\rangle \\
 & \Rightarrow & \textbf{John}\ \langle\texttt{verb}\rangle\langle\texttt{direct-object-phrase}\rangle \\
 & \Rightarrow & \textbf{John eats}\ \langle\texttt{direct-object-phrase}\rangle \\
 & \Rightarrow & \textbf{John eats}\ \langle\texttt{determiner}\rangle\langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a}\ \langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a}\ \langle\texttt{adjective}\rangle\langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a big}\ \langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a big}\ \langle\texttt{adjective}\rangle\ \langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a big juicy}\ \langle\texttt{adjective-list}\rangle\langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a big juicy}\ \langle\texttt{common-noun}\rangle \\
 & \Rightarrow & \textbf{John eats a big juicy hamburger}
\end{array}
$$

Tämä on esimerkki *vasemmasta johdosta* (leftmost derivation), mihin palataan pian.

Edellistä johtoa vastaava jäsennyspuu:



Esimerkki 2.8: Toinen sekä historiallisesti että nykykäytännön kannalta tärkeä yhteydettömien kielioppien sovellus on ohjelmointikielten syntaksin kuvaaminen.

Tarkastellaan esimerkkinä pientä Pascal-kielen osaa:

$$
\begin{array}{rcl}
\langle\texttt{lause}\rangle & \rightarrow & \langle\texttt{ehtolause}\rangle \mid \langle\texttt{koottu-lause}\rangle \mid \langle\texttt{sijoitus}\rangle \mid \langle\texttt{kutsu}\rangle \\
\langle\texttt{ehtolause}\rangle & \rightarrow & \textbf{if}\ \langle\texttt{ehto}\rangle\ \textbf{then}\ \langle\texttt{lause}\rangle\ \textbf{else}\ \langle\texttt{lause}\rangle \\
\langle\texttt{ehto}\rangle & \rightarrow & \texttt{x=0} \\
\langle\texttt{koottu-lause}\rangle & \rightarrow & \textbf{begin}\ \langle\texttt{lausejono}\rangle\ \textbf{end} \\
\langle\texttt{lausejono}\rangle & \rightarrow & \langle\texttt{lause}\rangle \mid \langle\texttt{lause}\rangle\ \textbf{;}\ \langle\texttt{lausejono}\rangle \\
\langle\texttt{sijoitus}\rangle & \rightarrow & \texttt{x:= 0} \\
\langle\texttt{kutsu}\rangle & \rightarrow & \mathsf{a} \mid \mathsf{b} \mid \mathsf{c}
\end{array}
$$

Kielioppi toimii paitsi ohjelmoijan muistilistana myös kääntäjän keskeisen vaiheen, jäsentäjän, laatimisen lähtökohtana. Jäsennystä edeltävä vaihe on selaus, joka pilkkoo ohjelman perusalkioihin kuten lukuvakiot, varatut sanat ja muuttujien nimet. Jäsentäjä ja selaaja muodostetaan usein automaattisesti kieliopista ja perusalkioita kuvaavista säännöllisistä lausekkeista (ks. sivu 90).

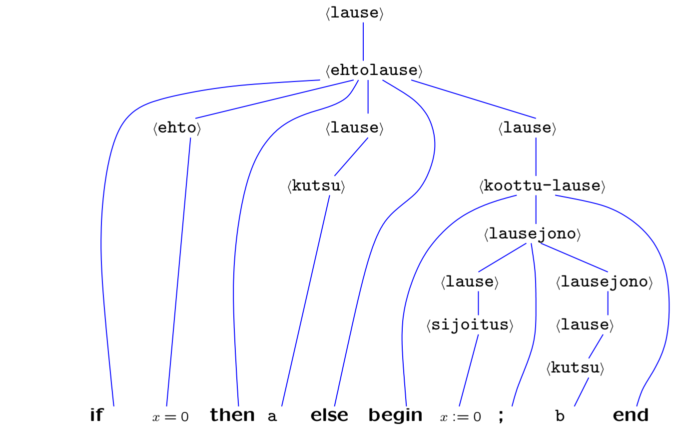
Esimerkkijohto:

$$
\begin{array}{rl}
\langle\texttt{lause}\rangle & \Rightarrow \langle\texttt{ehtolause}\rangle \\
& \Rightarrow \textbf{if}\ \langle\texttt{ehto}\rangle\ \textbf{then}\ \langle\texttt{lause}\rangle\ \textbf{else}\ \langle\texttt{lause}\rangle \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \langle\texttt{lause}\rangle\ \textbf{else}\ \langle\texttt{lause}\rangle \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \langle\texttt{kutsu}\rangle\ \textbf{else}\ \langle\texttt{lause}\rangle \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else}\ \langle\texttt{lause}\rangle \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else}\ \langle\texttt{koottu-lause}\rangle \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \langle\texttt{lausejono}\rangle\ \textbf{end} \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \langle\texttt{lause}\rangle\ \textbf{;}\ \langle\texttt{lausejono}\rangle\ \textbf{end} \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \langle\texttt{sijoitus}\rangle\ \textbf{;}\ \langle\texttt{lausejono}\rangle\ \textbf{end} \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \texttt{x:=0}\ \textbf{;}\ \langle\texttt{lausejono}\rangle\ \textbf{end} \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \texttt{x:=0}\ \textbf{;}\ \langle\texttt{lause}\rangle\ \textbf{end} \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \texttt{x:=0}\ \textbf{;}\ \langle\texttt{kutsu}\rangle\ \textbf{end} \\
& \Rightarrow \textbf{if}\ \texttt{x=0}\ \textbf{then}\ \mathsf{a}\ \textbf{else begin}\ \texttt{x:=0}\ \textbf{;}\ \texttt{b}\ \textbf{end}
\end{array}
$$

Vastaava ohjelmanpätkä:

```
if x=0 then
    a
else begin
    x:= 0;
    b
end
```


Vielä jäsennyspuu samalle lauseelle:



Ohjelman käsittely kääntäjässä jäsennyksen jälkeen perustuu nimenomaan sen jäsennyspuuhun.

Moniselitteisyys [Sipser s. 107–108]

Kontekstittoman kielen lauseella on tyypillisesti useita johtoja. Esim. kieliopissa

$$
\begin{array}{rcl}
S & \rightarrow & AB \\
A & \rightarrow & \mathsf{a} \\
B & \rightarrow & \mathsf{b}
\end{array}
$$

merkkijono ab voidaan johtaa kahdella eri tavalla:

$$
\begin{array}{c}
S \Rightarrow AB \Rightarrow aB \Rightarrow ab \\
S \Rightarrow AB \Rightarrow Ab \Rightarrow ab
\end{array}
$$

Intuitiivisesti näissä johdoissa on vain käytetty samoja sääntöjä eri järjestyksessä. Johdot myös vastaavat samaa jäsennyspuuta.

Tällaisen triviaalin monijohtoisuuden poissulkemiseksi tarkastelemme *vasempia johtoja* (leftmost derivation). Vasemmassa johdossa sovelletaan aina produktiota merkkijonon vasemmanpuoleisimpaan muuttujaan. Siis em. kieliopissa merkkijonon ab ainoa vasen johto on

$$S \Rightarrow AB \Rightarrow aB \Rightarrow ab.$$