
Problem 2.27. Let $G = (V, \Sigma, R, \langle \text{STMT} \rangle)$ be the following grammar.

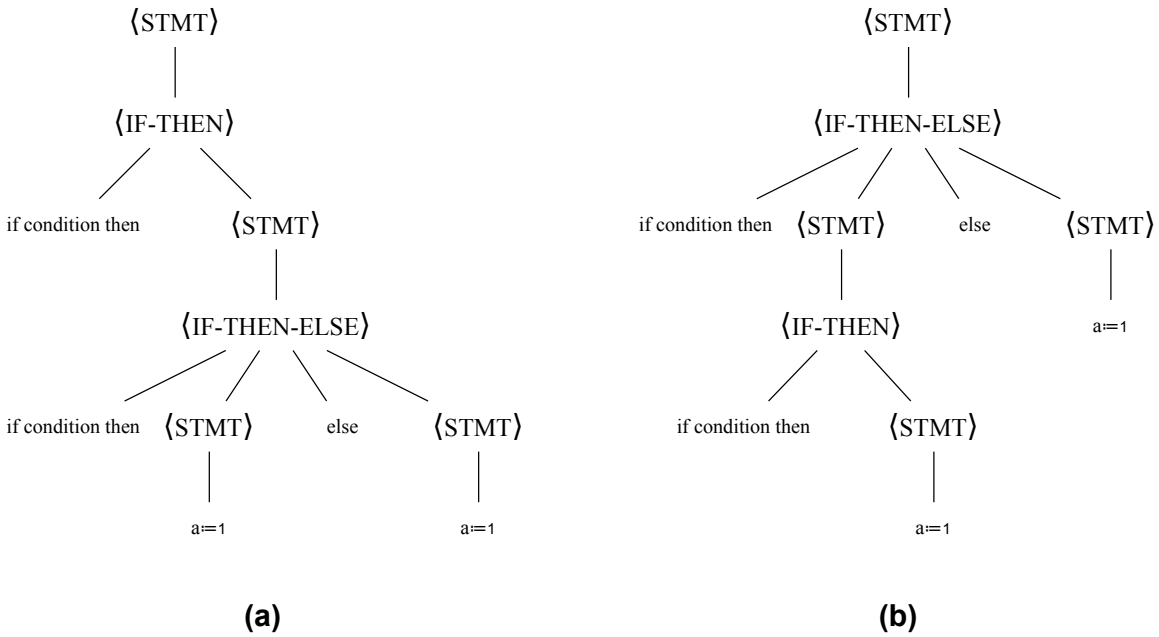
$$\begin{aligned} \langle \text{STMT} \rangle &\to \langle \text{ASSIGN} \rangle \mid \langle \text{IF-THEN} \rangle \mid \langle \text{IF-THEN-ELSE} \rangle \\ \langle \text{IF-THEN} \rangle &\to \texttt{if condition then } \langle \text{STMT} \rangle \\ \langle \text{IF-THEN-ELSE} \rangle &\to \texttt{if condition then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\ \langle \text{ASSIGN} \rangle &\to \texttt{a:=1} \end{aligned}$$

$$\Sigma = \{\texttt{if}, \texttt{condition}, \texttt{then}, \texttt{else}, \texttt{a:=1}\}$$
$$V = \{\langle \text{STMT} \rangle, \langle \text{IF-THEN} \rangle, \langle \text{IF-THEN-ELSE} \rangle, \langle \text{ASSIGN} \rangle\}$$

G is a natural-looking grammar for a fragment of a programming language, but G is ambiguous.

Part a. Show that G is ambiguous.

The string "if condition then if condition then a:=1 else a:=1" has two different parse trees in G.



Two different parse trees of the string
"if condition then if condition then a:=1 else a:=1".

Part b. Give a new unambiguous grammar for the same language.

$$
\begin{aligned}
\langle \text{STMT} \rangle &\rightarrow \langle \text{ASSIGN} \rangle \mid \langle \text{IF-THEN} \rangle \\
\langle \text{IF-THEN} \rangle &\rightarrow \text{if condition then } \langle \text{IF-BODY} \rangle \\
\langle \text{IF-BODY} \rangle &\rightarrow \langle \text{ASSIGN} \rangle \mid \langle \text{IF-THEN} \rangle \mid \langle \text{ASSIGN} \rangle \ \langle \text{ELSE} \rangle \\
\langle \text{ELSE} \rangle &\rightarrow \text{else } \langle \text{ELSE-BODY} \rangle \\
\langle \text{ELSE-BODY} \rangle &\rightarrow \langle \text{ASSIGN} \rangle \mid \langle \text{IF-THEN} \rangle \\
\langle \text{ASSIGN} \rangle &\rightarrow \text{a:=1}
\end{aligned}
$$