INF 5110: Compiler construction

Handout 5

Handout 5: Bottom-up parsing: sample grammars

Issued: 10. 3. 2023

For reference, to follow the slides, the handout includes some grammars we repeatedly used for illustration. These are various versions of the context-free grammar for expressions and other things. The first version is the "obvious" one.

$$
\begin{aligned}
exp &\rightarrow exp\ op\ exp \mid (\,exp\,) \mid \textbf{number} \\
op &\rightarrow +\ \mid\ -\ \mid\ *
\end{aligned}
\tag{1}
$$

The second version is the slightly less obvious one, used to take care of precedences (like multiplication over addition). The fact that in this grammar we don't just stipulate "multiplication binds stronger than addition and substraction" on top of the obvious grammar rules, but encode that in the productions without resorting to addition conditions on top of the grammar, makes the grammar slightly less readable.

$$
\begin{aligned}
exp &\rightarrow exp\ addop\ term \mid term \\
addop &\rightarrow +\ \mid\ - \\
term &\rightarrow term\ mulop\ factor \mid factor \\
mulop &\rightarrow * \\
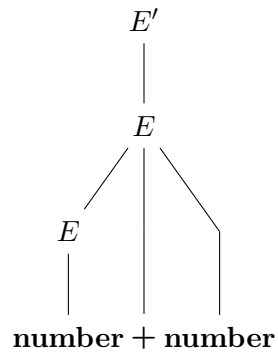factor &\rightarrow (\,exp\,) \mid \textbf{number}
\end{aligned}
\tag{2}
$$

Grammars to illustrate bottom-up

The following 2 (artificial) grammars (and the parse-trees) are used to illustrate the bottom-up parsing process.

Simplistic addition expressions

$$
\begin{aligned}
E' &\rightarrow E \\
E &\rightarrow E + \textbf{number} \mid \textbf{number}
\end{aligned}
$$

```
           E'
           |
           E
          /|\
        E  | \
        |  |  \
     number + number
```

Artificial grammar

$$
\begin{aligned}
S' &\rightarrow S \\
S &\rightarrow AB\mathbf{t_7} \mid \ldots \\
A &\rightarrow \mathbf{t_4 t_5} \mid \mathbf{t_1} B \mid \ldots \\
B &\rightarrow \mathbf{t_2 t_3} \mid A\mathbf{t_6} \mid \ldots
\end{aligned}
$$

```
                  S'
                  |
                  S
              /   |    \
            A     B      \
           / \   / \      \
             B     A       \
            /|\   /|\       \
          t1 t2 t3 t4 t5 t6  t7
```
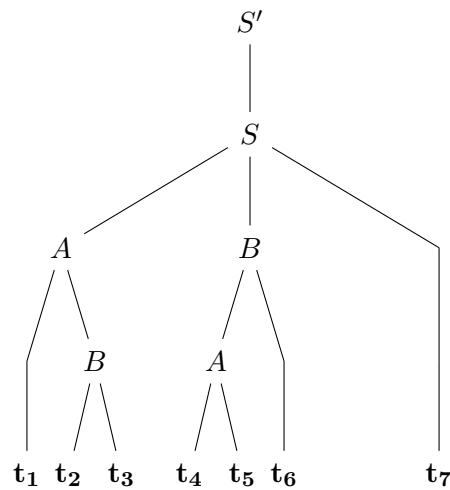
Grammars to illustrate LR(0) construction

Another example used in the lecture is the "simplistic additions" (see before).

Parentheses

$$
\begin{aligned}
S' &\rightarrow S \\
S &\rightarrow (\,S\,)\,S \mid \epsilon
\end{aligned}
$$

$$
\begin{aligned}
S' &\rightarrow .S \\
S' &\rightarrow S. \\
S &\rightarrow .(\,S\,)\,S \\
S &\rightarrow (.S\,)\,S \\
S &\rightarrow (\,S.\,)\,S \\
S &\rightarrow (\,S\,).S \\
S &\rightarrow (\,S\,)\,S. \\
S &\rightarrow .
\end{aligned}
$$

Simplistic addition

$$
\begin{array}{rcl}
E' & \to & .E \\
E' & \to & E. \\
E & \to & .E + \textbf{number} \\
E & \to & E. + \textbf{number} \\
E & \to & E + .\textbf{number} \\
E & \to & E + \textbf{number}. \\
E & \to & .\textbf{number} \\
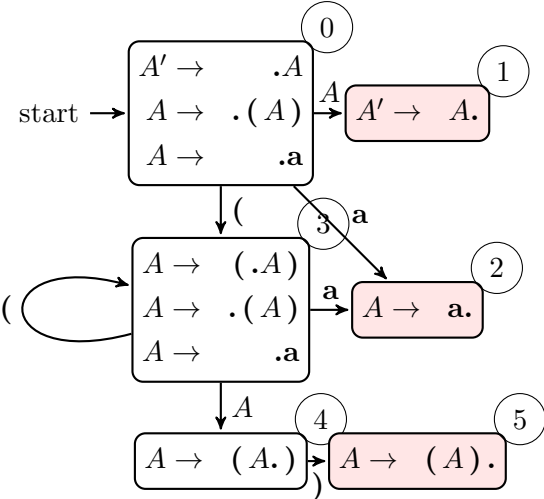E & \to & \textbf{number}. \\
\end{array}
$$
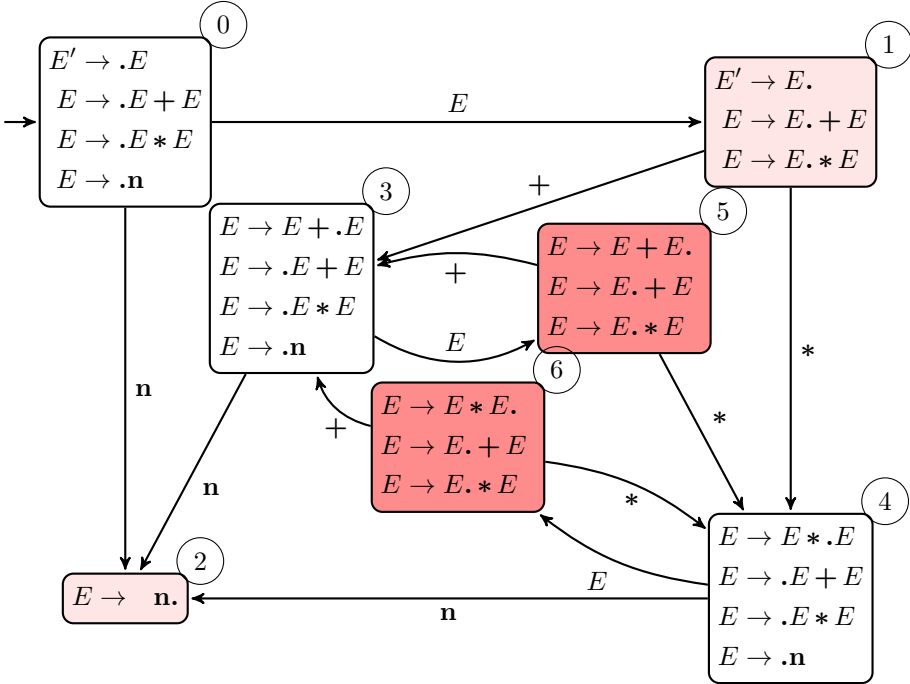
Figure 1: DFA for simple parentheses
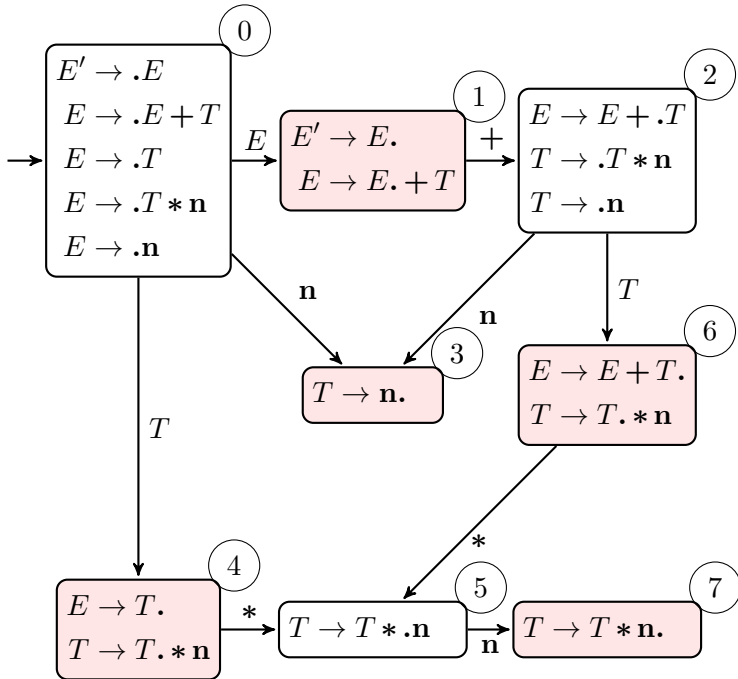
Figure 2: DFA for addition and multiplication (ambiguius)

Figure 3: DFA for addition and multiplication (unambiguious)