Theory of Formal Languages and Automata

Lecture 9

Mahdi Dolati

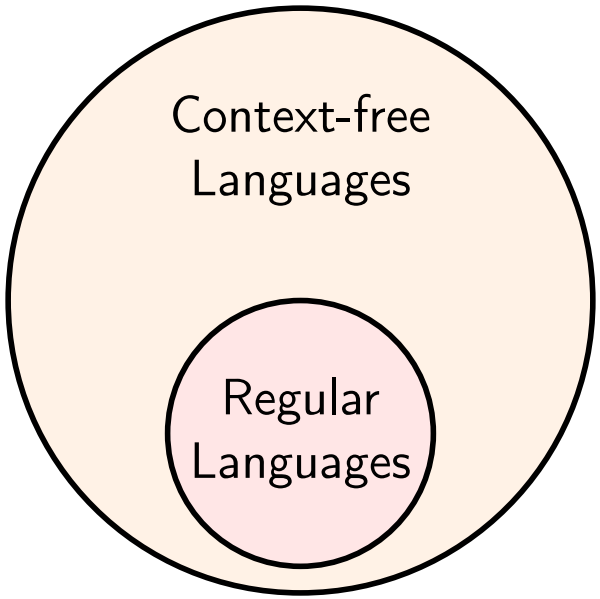
Sharif University of Technology

Fall 2023

October 30, 2023

Context-free Languages

- Context-free grammars specify context-free languages
  - More powerful tool
  - Used in compilation of programming languages (parser):
    - Tools need the CF grammar to construct a parser
- Pushdown automata: Recognize the CF languages



Context-Free Grammars

- A grammar consists of:
  - A set of substitution rules (productions),
  - A set of variables,
  - A start variable: Typically at the left-hand side of the topmost rule,
  - A set of terminals (similar to alphabets),

Example

A grammar, called G_1:

$$A \to 0A1$$
$$A \to B$$
$$B \to \#$$

Context-Free Grammars

- A grammar generates all strings of the language that it describes.
- Derivation: A sequence of substitutions that gives a string, starting from the start variable,
  1. Write down the start variable,
  2. Find a variable and rule that starts with that variable. Replace the variable with the right-hand side of that rule.
  3. If there is variable go to 2.

Example

$$A \to 0A1$$
$$A \to B$$
$$B \to \#$$

Derivation of string "$000\#111$":

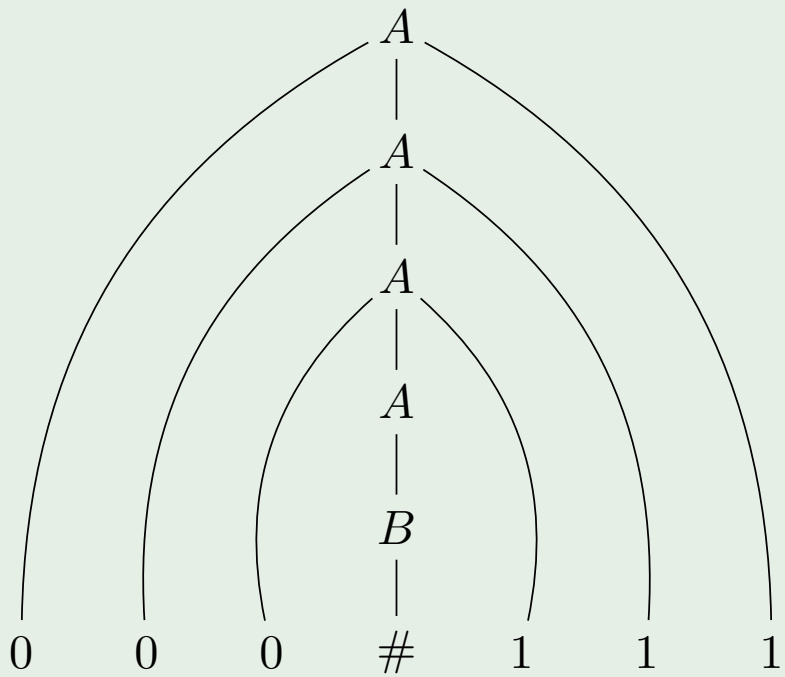
$$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111.$$

Context-Free Grammars

- Parse tree.

Example

$$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111.$$



Context-Free Grammars

- Language of a grammar: Set of all generated strings,
  - $L(G_1)$ is the language of grammar G_1,
  - Example: $L(G_1) = \{0^n \# 1^n | n \geq 0\}$.
- Notation:

$$\begin{array}{l} A \to 0A1 \\ A \to B \end{array} \quad \equiv \quad A \to 0A1 \mid B$$

Context-Free Grammars

Example

$$
\begin{aligned}
\langle \text{SENTENCE} \rangle &\to \langle \text{NOUN-PHRASE} \rangle \langle \text{VERB-PHRASE} \rangle \\
\langle \text{NOUN-PHRASE} \rangle &\to \langle \text{CMPLX-NOUN} \rangle \mid \langle \text{CMPLX-NOUN} \rangle \langle \text{PREP-PHRASE} \rangle \\
\langle \text{VERB-PHRASE} \rangle &\to \langle \text{CMPLX-VERB} \rangle \mid \langle \text{CMPLX-VERB} \rangle \langle \text{PREP-PHRASE} \rangle \\
\langle \text{PREP-PHRASE} \rangle &\to \langle \text{PREP} \rangle \langle \text{CMPLX-NOUN} \rangle \\
\langle \text{CMPLX-NOUN} \rangle &\to \langle \text{ARTICLE} \rangle \langle \text{NOUN} \rangle \\
\langle \text{CMPLX-VERB} \rangle &\to \langle \text{VERB} \rangle \mid \langle \text{VERB} \rangle \langle \text{NOUN-PHRASE} \rangle \\
\langle \text{ARTICLE} \rangle &\to \text{a} \mid \text{the} \\
\langle \text{NOUN} \rangle &\to \text{boy} \mid \text{girl} \mid \text{flower} \\
\langle \text{VERB} \rangle &\to \text{touches} \mid \text{likes} \mid \text{sees} \\
\langle \text{PREP} \rangle &\to \text{with}
\end{aligned}
$$

Example (Cont.)

$$
\begin{aligned}
\langle \text{SENTENCE} \rangle &\Rightarrow \langle \text{NOUN-PHRASE} \rangle \langle \text{VERB-PHRASE} \rangle \\
&\Rightarrow \langle \text{CMPLX-NOUN} \rangle \langle \text{VERB-PHRASE} \rangle \\
&\Rightarrow \langle \text{ARTICLE} \rangle \langle \text{NOUN} \rangle \langle \text{VERB-PHRASE} \rangle \\
&\Rightarrow \text{a } \langle \text{NOUN} \rangle \langle \text{VERB-PHRASE} \rangle \\
&\Rightarrow \text{a boy } \langle \text{VERB-PHRASE} \rangle \\
&\Rightarrow \text{a boy } \langle \text{CMPLX-VERB} \rangle \\
&\Rightarrow \text{a boy } \langle \text{VERB} \rangle \\
&\Rightarrow \text{a boy sees}
\end{aligned}
$$

Context-Free Grammars

Definition (Context-free Grammar)

A CF grammer is a 4-tuple (V, Σ, R, S), where

1. V: Finite set of variables,
2. Σ: Finite set of variables,
   - $V \cap \Sigma = \emptyset$
3. R: Finite set of rules,
   - $V \times (V \cup \Sigma)^*$
4. $S \in V$: The start variable.

Context-Free Grammars

Example

$$S \to aSb \mid \varepsilon$$

1. $V = ?$,
2. $\Sigma = ?$,
3. $R = ?$,
4. The start variable $= ?$.

Example (Cont.)

$$S \to aSb \mid \varepsilon$$

1. $V = \{S\}$,
2. $\Sigma = \{a, b\}$,
3. $R = \{S \to aSb, S \to \varepsilon\}$,
4. The start variable $= S$.

Context-Free Grammars

- $uAv \Rightarrow uwv$
  - uAv yields uwv
- $u \overset{*}{\Rightarrow} v$
  - u drives v
  - $u \Rightarrow u_1 \Rightarrow \cdots \Rightarrow v$
- Language of a grammar: $\{w \in \Sigma^* | S \overset{*}{\Rightarrow} w\}$

Context-Free Grammars

Example

The language of all strings of properly nested parentheses:

$$S \to aSb \mid SS \mid \varepsilon.$$

Think of a as a left parenthesis and b as a right one:

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

Context-Free Grammars

Example

The language of all palindromes of even length:

$$G_p \ : \ S \to aSa \mid bSb \mid \varepsilon.$$

$$L(G_p) = \{ww^R \ : \ w \in \{a, b\}^*\}.$$

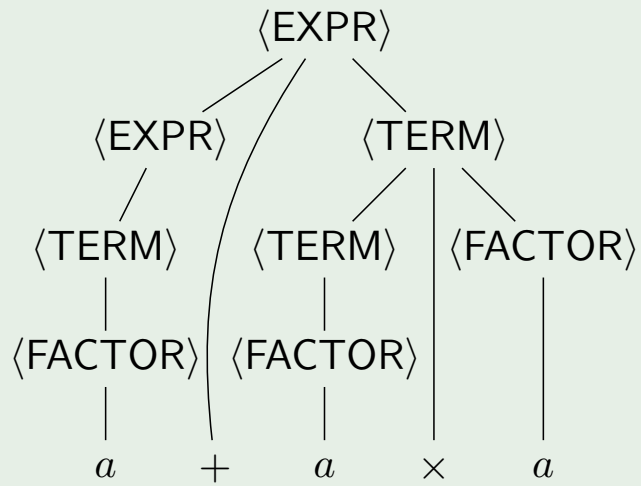
Context-Free Grammars

Example

A fragment of a programming language concerned with arithmetic expressions:

$$\langle \text{EXPR} \rangle \to \langle \text{EXPR} \rangle + \langle \text{TERM} \rangle \mid \langle \text{TERM} \rangle$$
$$\langle \text{TERM} \rangle \to \langle \text{TEM} \rangle \times \langle \text{FACTOR} \rangle \mid \langle \text{FACTOR} \rangle$$
$$\langle \text{FACTOR} \rangle \to (\langle \text{EXPR} \rangle) \mid a$$

Parse tree represents a meaning:



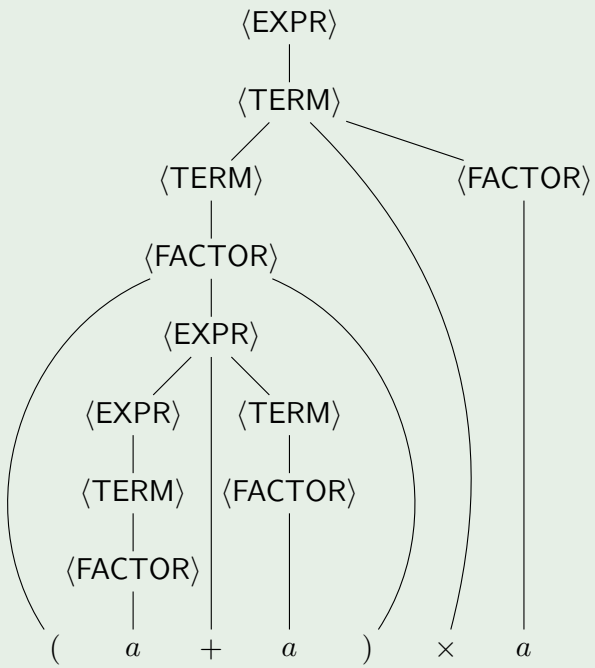
Context-Free Grammars

Example

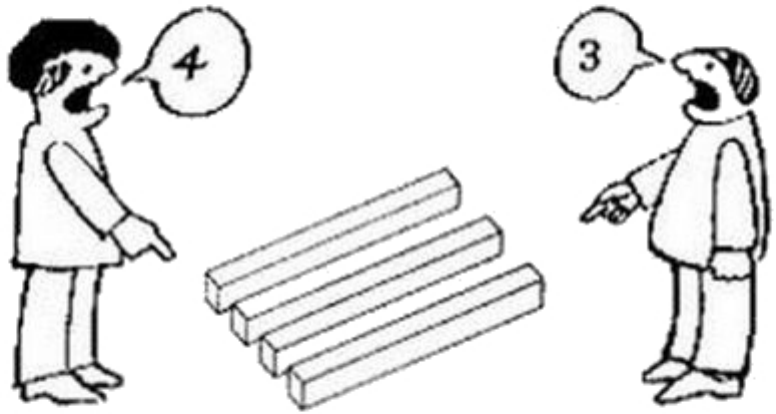
A fragment of a programming language concerned with arithmetic expressions:

$$\langle \textbf{EXPR} \rangle \rightarrow \langle \textbf{EXPR} \rangle + \langle \textbf{TERM} \rangle \mid \langle \textbf{TERM} \rangle$$
$$\langle \textbf{TERM} \rangle \rightarrow \langle \textbf{TEM} \rangle \times \langle \textbf{FACTOR} \rangle \mid \langle \textbf{FACTOR} \rangle$$
$$\langle \textbf{FACTOR} \rangle \rightarrow (\langle \textbf{EXPR} \rangle) \mid a$$

Parse tree represents a meaning:



Ambiguity



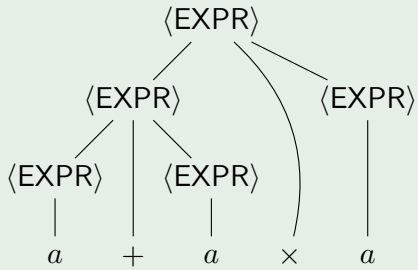
Ambiguity

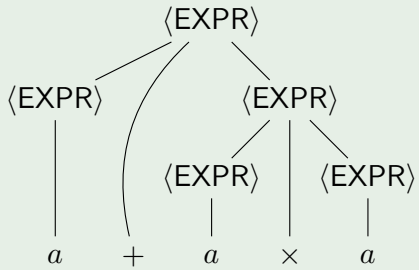
- Two different parse trees: The string is derived ambiguously,

Example

$G_5 : \langle \text{EXPR} \rangle \to \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \mid \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \mid (\langle \text{EXPR} \rangle) \mid a$

"$a + a \times a$" $\in L(G_5)$.





Ambiguity

- Parse tree vs. derivation (substitution order),
  - Fix an order to relate the derivation to the structure:
    - Leftmost derivation: Always substitute the leftmost variable.

Definition

- A string is derived ambiguously if it has more than one different leftmost derivations.

Example

$$G_5 : \langle \text{EXPR} \rangle \to \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \mid \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \mid (\langle \text{EXPR} \rangle) \mid a$$

$$\begin{aligned} \langle \text{EXPR} \rangle &\Rightarrow \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \\ &\Rightarrow \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \\ &\Rightarrow a + \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \\ &\Rightarrow a + a \times \langle \text{EXPR} \rangle \\ &\Rightarrow a + a \times a \end{aligned}$$

$$\begin{aligned} \langle \text{EXPR} \rangle &\Rightarrow \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \\ &\Rightarrow a + \langle \text{EXPR} \rangle \\ &\Rightarrow a + \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \\ &\Rightarrow a + a \times \langle \text{EXPR} \rangle \\ &\Rightarrow a + a \times a \end{aligned}$$

Ambiguity

Quiz

Show following grammar is ambiguous by giving two different parse trees for a single sentence.

$$\langle \text{SENTENCE} \rangle \to \langle \text{NOUN-PHRASE} \rangle \langle \text{VERB-PHRASE} \rangle$$
$$\langle \text{NOUN-PHRASE} \rangle \to \langle \text{CMPLX-NOUN} \rangle \mid \langle \text{CMPLX-NOUN} \rangle \langle \text{PREP-PHRASE} \rangle$$
$$\langle \text{VERB-PHRASE} \rangle \to \langle \text{CMPLX-VERB} \rangle \mid \langle \text{CMPLX-VERB} \rangle \langle \text{PREP-PHRASE} \rangle$$
$$\langle \text{PREP-PHRASE} \rangle \to \langle \text{PREP} \rangle \langle \text{CMPLX-NOUN} \rangle$$
$$\langle \text{CMPLX-NOUN} \rangle \to \langle \text{ARTICLE} \rangle \langle \text{NOUN} \rangle$$
$$\langle \text{CMPLX-VERB} \rangle \to \langle \text{VERB} \rangle \mid \langle \text{VERB} \rangle \langle \text{NOUN-PHRASE} \rangle$$
$$\langle \text{ARTICLE} \rangle \to \text{a} \mid \text{the}$$
$$\langle \text{NOUN} \rangle \to \text{boy} \mid \text{girl} \mid \text{flower}$$
$$\langle \text{VERB} \rangle \to \text{touches} \mid \text{likes} \mid \text{sees}$$
$$\langle \text{PREP} \rangle \to \text{with}$$

Ambiguity

Example

$$\begin{aligned}
\langle \text{IF-STMT} \rangle &\to \texttt{if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \\
&\quad \mid \texttt{if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
\langle \text{EXPR} \rangle &\to E_1 \mid E_2 \\
\langle \text{STMT} \rangle &\to S_1 \mid S_2 \mid \langle \text{IF-STMT} \rangle
\end{aligned}$$

$$\begin{aligned}
\langle \text{IF-STMT} \rangle &\Rightarrow \texttt{if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then } \langle \text{IF-STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } E_2 \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } E_2 \texttt{ then } S_1 \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } E_2 \texttt{ then } S_1 \texttt{ else } S_2
\end{aligned}$$

$$\begin{array}{l}
\texttt{if } E_1 \texttt{ then} \\
\qquad \texttt{if } E_2 \texttt{ then} \\
\qquad\qquad S_1 \\
\qquad \texttt{else} \\
\qquad\qquad S_2
\end{array}$$

Ambiguity

Example

$$\begin{aligned}
\langle \text{IF-STMT} \rangle &\to \texttt{if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \\
&\quad \mid \texttt{if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
\langle \text{EXPR} \rangle &\to E_1 \mid E_2 \\
\langle \text{STMT} \rangle &\to S_1 \mid S_2 \mid \langle \text{IF-STMT} \rangle
\end{aligned}$$

$$\begin{aligned}
\langle \text{IF-STMT} \rangle &\Rightarrow \texttt{if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then } \langle \text{IF-STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } \langle \text{EXPR} \rangle \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } E_2 \texttt{ then } \langle \text{STMT} \rangle \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } E_2 \texttt{ then } S_1 \texttt{ else } \langle \text{STMT} \rangle \\
&\Rightarrow \texttt{if } E_1 \texttt{ then if } E_2 \texttt{ then } S_1 \texttt{ else } S_2
\end{aligned}$$

```
if E1 then
        if E2 then
                S1
else
        S2
```

Ambiguity

- Rightmost derivation: Similar to leftmost derivation,
- Can be used to check ambiguity.

Definition

- A string is derived ambiguously if it has more than one different rightmost derivations.

Example

$$G_5: \ \langle \text{EXPR} \rangle \to \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \mid \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \mid (\langle \text{EXPR} \rangle) \mid a$$

$$\begin{aligned} \langle \text{EXPR} \rangle &\Rightarrow \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \\ &\Rightarrow \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \\ &\Rightarrow \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \times a \\ &\Rightarrow \langle \text{EXPR} \rangle + a \times a \\ &\Rightarrow a + a \times a \end{aligned}$$

$$\begin{aligned} \langle \text{EXPR} \rangle &\Rightarrow \langle \text{EXPR} \rangle \times \langle \text{EXPR} \rangle \\ &\Rightarrow \langle \text{EXPR} \rangle \times a \\ &\Rightarrow \langle \text{EXPR} \rangle + \langle \text{EXPR} \rangle \times a \\ &\Rightarrow \langle \text{EXPR} \rangle + a \times a \\ &\Rightarrow a + a \times a \end{aligned}$$

Ambiguity

Definition

- A grammar is ambiguous if it generates at least one string ambiguously.

- A language is inherently ambiguous: There is no unambiguous grammar that generates it.

Example

$$\{a^i b^j c^k \mid i = j \text{ or } j = k\}.$$

Ambiguity

- To determine if a context free grammar is ambiguous or inherently ambiguous is undecidable,
- There is no algorithm which will correctly say “yes” or “no” in a finite time for all grammars.