Compiler Construction

Lecture 8: Syntactic Analysis III ($LL(k)$ Grammars)

Thomas Noll

Lehrstuhl für Informatik 2
(Software Modeling and Verification)

RWTH Aachen University

noll@cs.rwth-aachen.de

http://www-i2.informatik.rwth-aachen.de/i2/cc10/

Winter semester 2010/11

Outline

Top-Down Parsing

Approach:

1. Given $G \in CFG_\Sigma$, construct a nondeterministic pushdown automaton (PDA) which accepts $L(G)$ and which additionally computes corresponding leftmost derivations (similar to the proof of "$L(CFG_\Sigma) \subseteq L(PDA_\Sigma)$")
   - input alphabet: Σ
   - pushdown alphabet: X
   - output alphabet: $[p]$
   - state set: omitted
2. Remove nondeterminism by allowing lookahead on the input: $G \in LL(k)$ iff $L(G)$ recognizable by deterministic PDA with lookahead of k symbols

The Nondeterministic Top-Down Automaton I

Definition (Nondeterministic top-down parsing automaton)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$. The nondeterministic top-down parsing automaton of G, NTA(G), is defined by the following components.

- Input alphabet: Σ
- Pushdown alphabet: X
- Output alphabet: $[p]$
- Configurations: $\Sigma^* \times X^* \times [p]^*$ (top of pushdown to the left)
- Transitions for $w \in \Sigma^*$, $\alpha \in X^*$, and $z \in [p]^*$:
  expansion steps: if $\pi_i = A \to \beta$, then $(w, A\alpha, z) \vdash (w, \beta\alpha, zi)$
  matching steps: for every $a \in \Sigma$, $(aw, a\alpha, z) \vdash (w, \alpha, z)$
- Initial configuration for $w \in \Sigma^*$: (w, S, ε)
- Final configurations: $\{\varepsilon\} \times \{\varepsilon\} \times [p]^*$

Remark: NTA(G) is nondeterministic iff G contains $A \to \beta \mid \gamma$

The Nondeterministic Top-Down Automaton II

Theorem (Correctness of NTA(G))

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ *and* NTA(G) *as before. Then, for every* $w \in \Sigma^*$ *and* $z \in [p]^*$,

$$(w, S, \varepsilon) \vdash^* (\varepsilon, \varepsilon, z) \quad \textit{iff} \quad z \textit{ is a leftmost analysis of } w$$

Proof.

$\Longrightarrow$ (soundness): see exercises

$\Longleftarrow$ (completeness): on the board

□

Outline

Adding Lookahead

Goal: resolve nondeterminism of NTA(G) by supporting lookahead of $k \in \mathbb{N}$ symbols on the input

$\Longrightarrow$ determination of expanding A-production by next k symbols

Adding Lookahead

Goal: resolve nondeterminism of NTA(G) by supporting lookahead of $k \in \mathbb{N}$ symbols on the input

$\Longrightarrow$ determination of expanding A-production by next k symbols

Definition 8.1 (first_k set)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$, $\alpha \in X^*$, and $k \in \mathbb{N}$. Then the first_k set of α, $\text{first}_k(\alpha) \subseteq \Sigma^*$, is given by

$$\text{first}_k(\alpha) := \{v \in \Sigma^k \mid \text{ex. } w \in \Sigma^* \text{ such that } \alpha \Rightarrow^* vw\} \cup \{v \in \Sigma^{<k} \mid \alpha \Rightarrow^* v\}$$

Adding Lookahead

Goal: resolve nondeterminism of NTA(G) by supporting lookahead of $k \in \mathbb{N}$ symbols on the input

$\implies$ determination of expanding A-production by next k symbols

Definition 8.1 (first_k set)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$, $\alpha \in X^*$, and $k \in \mathbb{N}$. Then the first_k set of α, $\mathrm{first}_k(\alpha) \subseteq \Sigma^*$, is given by

$$\mathrm{first}_k(\alpha) := \{v \in \Sigma^k \mid \text{ex. } w \in \Sigma^* \text{ such that } \alpha \Rightarrow^* vw\} \cup \{v \in \Sigma^{<k} \mid \alpha \Rightarrow^* v\}$$

Remark: $\mathrm{first}_k(\alpha)$ is effectively computable

Adding Lookahead

Goal: resolve nondeterminism of NTA(G) by supporting lookahead of $k \in \mathbb{N}$ symbols on the input
$\implies$ determination of expanding A-production by next k symbols

Definition 8.1 (first$_k$ set)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$, $\alpha \in X^*$, and $k \in \mathbb{N}$. Then the first$_k$ set of α, $\text{first}_k(\alpha) \subseteq \Sigma^*$, is given by

$$\text{first}_k(\alpha) := \{v \in \Sigma^k \mid \text{ex. } w \in \Sigma^* \text{ such that } \alpha \Rightarrow^* vw\} \cup \{v \in \Sigma^{<k} \mid \alpha \Rightarrow^* v\}$$

Remark: $\text{first}_k(\alpha)$ is effectively computable

Example 8.2 (first$_k$ set)

Let $G : S \rightarrow aSb \mid \varepsilon$.

1. $\text{first}_1(ab) = \{a\} = \text{first}_2(a)$

Adding Lookahead

Goal: resolve nondeterminism of NTA(G) by supporting lookahead of $k \in \mathbb{N}$ symbols on the input

$\implies$ determination of expanding A-production by next k symbols

Definition 8.1 (first$_k$ set)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$, $\alpha \in X^*$, and $k \in \mathbb{N}$. Then the first$_k$ set of α, $\text{first}_k(\alpha) \subseteq \Sigma^*$, is given by

$$\text{first}_k(\alpha) := \{v \in \Sigma^k \mid \text{ex. } w \in \Sigma^* \text{ such that } \alpha \Rightarrow^* vw\} \cup \{v \in \Sigma^{<k} \mid \alpha \Rightarrow^* v\}$$

Remark: $\text{first}_k(\alpha)$ is effectively computable

Example 8.2 (first$_k$ set)

Let $G : S \to aSb \mid \varepsilon$.

1. $\text{first}_1(ab) = \{a\} = \text{first}_2(a)$
2. $\text{first}_3(S) = \{\varepsilon, ab, aab, aaa\}$

Goal: resolve nondeterminism of NTA(G) by supporting lookahead of $k \in \mathbb{N}$ symbols on the input

$\Longrightarrow$ determination of expanding A-production by next k symbols

Definition 8.1 (first$_k$ set)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$, $\alpha \in X^*$, and $k \in \mathbb{N}$. Then the first$_k$ set of α, $\text{first}_k(\alpha) \subseteq \Sigma^*$, is given by

$$\begin{aligned}\text{first}_k(\alpha) := \{v \in \Sigma^k \mid \text{ex. } w \in \Sigma^* \text{ such that } \alpha \Rightarrow^* vw\} \cup \\ \{v \in \Sigma^{<k} \mid \alpha \Rightarrow^* v\}\end{aligned}$$

Remark: $\text{first}_k(\alpha)$ is effectively computable

Example 8.2 (first$_k$ set)

Let $G : S \to aSb \mid \varepsilon$.

1. $\text{first}_1(ab) = \{a\} = \text{first}_2(a)$
2. $\text{first}_3(S) = \{\varepsilon, ab, aab, aaa\}$
3. $\text{first}_3(Sa) = \{a, aba, aab, aaa\}$

Outline

$LL(k)$ Grammars I

$LL(k)$: reading of input from left to right with k-lookahead, computing a leftmost analysis

$LL(k)$ Grammars I

$LL(k)$: reading of input from left to right with k-lookahead, computing a leftmost analysis

Definition 8.3 ($LL(k)$ grammar)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$ and $k \in \mathbb{N}$. Then G has the $LL(k)$ property (notation: $G \in LL(k)$) if for all leftmost derivations of the form

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \Rightarrow_l^* wx \\ \Rightarrow_l w\gamma\alpha \Rightarrow_l^* wy \end{cases}$$

such that $\beta \neq \gamma$, it follows that $\text{first}_k(x) \neq \text{first}_k(y)$
(i.e., different productions must not yield the same lookahead).

$LL(k)$ Grammars II

Remarks:

- If $G \in LL(k)$, then the leftmost derivation step for $wA\alpha$ in

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \Rightarrow_l^* wx \\ \Rightarrow_l w\gamma\alpha \Rightarrow_l^* wy \end{cases}$$

  is determined by the next k symbols following w.

$LL(k)$ Grammars II

Remarks:

- If $G \in LL(k)$, then the leftmost derivation step for $wA\alpha$ in

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \Rightarrow_l^* wx \\ \Rightarrow_l w\gamma\alpha \Rightarrow_l^* wy \end{cases}$$

  is determined by the next k symbols following w.

- Corresponding computations of NTA(G):

$$\begin{array}{ccccccc} (wx, S, \varepsilon) & \vdash^* & (x, A\alpha, z) & \overset{(*)}{\vdash} & (x, \beta\alpha, zi) & \vdash^* & (\varepsilon, \varepsilon, ziz') \\ (wy, S, \varepsilon) & \vdash^* & (y, A\alpha, z) & \overset{(*)}{\vdash} & (y, \gamma\alpha, zj) & \vdash^* & (\varepsilon, \varepsilon, zjz'') \end{array}$$

  where $\pi_i = A \rightarrow \beta$ and $\pi_j = A \rightarrow \gamma$

- Deterministic decision in $(*)$ possible if $\text{first}_k(x) \neq \text{first}_k(y)$

$LL(k)$ Grammars II

Remarks:

- If $G \in LL(k)$, then the leftmost derivation step for $wA\alpha$ in

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \Rightarrow_l^* wx \\ \Rightarrow_l w\gamma\alpha \Rightarrow_l^* wy \end{cases}$$

  is determined by the next k symbols following w.
- Corresponding computations of NTA(G):

$$\begin{array}{ccccccc} (wx, S, \varepsilon) & \vdash^* & (x, A\alpha, z) & \overset{(*)}{\vdash} & (x, \beta\alpha, zi) & \vdash^* & (\varepsilon, \varepsilon, ziz') \\ (wy, S, \varepsilon) & \vdash^* & (y, A\alpha, z) & \overset{(*)}{\vdash} & (y, \gamma\alpha, zj) & \vdash^* & (\varepsilon, \varepsilon, zjz'') \end{array}$$

  where $\pi_i = A \to \beta$ and $\pi_j = A \to \gamma$
- Deterministic decision in $(*)$ possible if $\text{first}_k(x) \neq \text{first}_k(y)$
- **Problem:** how to determine the A-production from the lookahead (potentially infinitely many derivations $\beta\alpha \Rightarrow_l^* x$ / $\gamma\alpha \Rightarrow_l^* y$)?

Lemma 8.4 (Characterization of $LL(k)$)

$G \in LL(k)$ iff for all leftmost derivations of the form

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \\ \Rightarrow_l w\gamma\alpha \end{cases}$$

such that $\beta \neq \gamma$, it follows that $\mathrm{first}_k(\beta\alpha) \cap \mathrm{first}_k(\gamma\alpha) = \emptyset$.

Lemma 8.4 (Characterization of $LL(k)$)

$G \in LL(k)$ iff for all leftmost derivations of the form

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \\ \Rightarrow_l w\gamma\alpha \end{cases}$$

such that $\beta \neq \gamma$, it follows that $\text{first}_k(\beta\alpha) \cap \text{first}_k(\gamma\alpha) = \emptyset$.

Proof.

omitted □

$LL(k)$ Grammars III

Lemma 8.4 (Characterization of $LL(k)$)

$G \in LL(k)$ iff for all leftmost derivations of the form

$$S \Rightarrow_l^* wA\alpha \begin{cases} \Rightarrow_l w\beta\alpha \\ \Rightarrow_l w\gamma\alpha \end{cases}$$

such that $\beta \neq \gamma$, it follows that $\text{first}_k(\beta\alpha) \cap \text{first}_k(\gamma\alpha) = \emptyset$.

Proof.

omitted □

Remarks:

- If $G \in LL(k)$, then the A-production is determined by the lookahead sets $\text{first}_k(\beta\alpha)$ (for every $A \to \beta \in P$).
- **Problem:** still infinitely many right contexts α to be considered (if β [or γ] "too short", i.e., $\text{first}_k(\beta\alpha) \neq \text{first}_k(\beta)$).
- **Idea:** α derives to "everything that follows A"

Outline

Goal: determine all possible lookaheads from production alone
(by combining all possible right contexts)

The follow_k Sets

Goal: determine all possible lookaheads from production alone (by combining all possible right contexts)

Definition 8.5 (follow_k set)

Let $G = \langle N, \Sigma, P, S \rangle \in CFG_\Sigma$, $A \in N$, and $k \in \mathbb{N}$. Then the follow_k set of A, $\text{follow}_k(A) \subseteq \Sigma^*$, is given by

$$\text{follow}_k(A) := \{v \in \text{first}_k(\alpha) \mid \text{ex. } w \in \Sigma^*, \alpha \in X^* \text{ such that } S \Rightarrow_l^* wA\alpha\}.$$

Outline

The Case $k = 1$

Motivation:

- $k = 1$ sufficient to resolve nondeterminism in “most” practical applications
- Implementation of $LL(k)$ parsers for $k > 1$ rather involved (cf. ANTLR [ANother Tool for Language Recognition; formerly PCCTS] at `http://www.antlr.org/`)

The Case $k = 1$

Motivation:

- $k = 1$ sufficient to resolve nondeterminism in "most" practical applications
- Implementation of $LL(k)$ parsers for $k > 1$ rather involved (cf. ANTLR [ANother Tool for Language Recognition; formerly PCCTS] at `http://www.antlr.org/`)

Abbreviations: $\mathrm{fi} := \mathrm{first}_1$, $\mathrm{fo} := \mathrm{follow}_1$, $\Sigma_\varepsilon := \Sigma \cup \{\varepsilon\}$

Corollary 8.6

1. *For every* $\alpha \in X^*$,
$$\mathrm{fi}(\alpha) = \{a \in \Sigma \mid \textit{ex. } w \in \Sigma^* : \alpha \Rightarrow^* aw\} \cup \{\varepsilon \mid \alpha \Rightarrow^* \varepsilon\} \subseteq \Sigma_\varepsilon$$
2. *For every* $A \in N$,
$$\mathrm{fo}(A) = \{x \in \mathrm{fi}(\alpha) \mid \textit{ex. } w \in \Sigma^*, \alpha \in X^* : S \Rightarrow_l^* wA\alpha\} \subseteq \Sigma_\varepsilon.$$

Lookahead Sets

Definition 8.7 (Lookahead set)

Given $\pi = A \to \beta \in P$,

$$\mathrm{la}(\pi) := \mathrm{fi}(\beta \cdot \mathrm{fo}(A)) \subseteq \Sigma_\varepsilon$$

is called the lookahead set of π (where $\mathrm{fi}(\Gamma) := \bigcup_{\gamma \in \Gamma} \mathrm{fi}(\gamma)$).

Lookahead Sets

Definition 8.7 (Lookahead set)

Given $\pi = A \to \beta \in P$,

$$\mathrm{la}(\pi) := \mathrm{fi}(\beta \cdot \mathrm{fo}(A)) \subseteq \Sigma_\varepsilon$$

is called the lookahead set of π (where $\mathrm{fi}(\Gamma) := \bigcup_{\gamma \in \Gamma} \mathrm{fi}(\gamma)$).

Corollary 8.8

1. *For all* $a \in \Sigma$,

$$a \in \mathrm{la}(A \to \beta) \textit{ iff } a \in \mathrm{fi}(\beta) \textit{ or } (\beta \Rightarrow^* \varepsilon \textit{ and } a \in \mathrm{fo}(A))$$

2. $\varepsilon \in \mathrm{la}(A \to \beta)$ *iff* $\beta \Rightarrow^* \varepsilon$ *and* $\varepsilon \in \mathrm{fo}(A)$

Characterization of $LL(1)$

Theorem 8.9 (Characterization of $LL(1)$)

$G \in LL(1)$ iff for all pairs of rules $A \to \beta \mid \gamma \in P$ (where $\beta \neq \gamma$):

$$\mathrm{la}(A \to \beta) \cap \mathrm{la}(A \to \gamma) = \emptyset.$$

Characterization of $LL(1)$

Theorem 8.9 (Characterization of $LL(1)$)

$G \in LL(1)$ *iff for all pairs of rules* $A \to \beta \mid \gamma \in P$ *(where* $\beta \neq \gamma$*):*

$$\mathrm{la}(A \to \beta) \cap \mathrm{la}(A \to \gamma) = \emptyset.$$

Proof.

on the board □

Characterization of $LL(1)$

Theorem 8.9 (Characterization of $LL(1)$)

$G \in LL(1)$ iff for all pairs of rules $A \to \beta \mid \gamma \in P$ (where $\beta \neq \gamma$):

$$\mathrm{la}(A \to \beta) \cap \mathrm{la}(A \to \gamma) = \emptyset.$$

Proof.

on the board □

Remark: the above theorem generally does not hold if $k > 1$ (cf. exercises)

Outline

Computing Lookahead Sets I

(see Waite/Goos: *Compiler Construction*, p. 164f)

Lemma 8.10 (Computation of fi/fo)

The sets $\mathrm{fi}(\alpha) \subseteq \Sigma_\varepsilon$ *(for* $\alpha \in X^*$*) and* $\mathrm{fo}(A) \subseteq \Sigma_\varepsilon$ *(for* $A \in N$*) are the least sets such that:*

1. $\mathrm{fi}(Y)$ *for* $Y \in X$*:*
   - $Y \in \Sigma \implies \mathrm{fi}(Y) = \{Y\}$
   - $Y \to A_1 \ldots A_k Z\alpha \in P, k \in \mathbb{N}, Z \in X, \varepsilon \in \mathrm{fi}(A_1) \cap \ldots \cap \mathrm{fi}(A_k)$, $a \in \mathrm{fi}(Z) \implies a \in \mathrm{fi}(Y)$
   - $Y \to A_1 \ldots A_k \in P, k \in \mathbb{N}, \varepsilon \in \mathrm{fi}(A_1) \cap \ldots \cap \mathrm{fi}(A_k) \implies \varepsilon \in \mathrm{fi}(Y)$

Computing Lookahead Sets I

(see Waite/Goos: *Compiler Construction*, p. 164f)

Lemma 8.10 (Computation of fi/fo)

The sets $\mathrm{fi}(\alpha) \subseteq \Sigma_\varepsilon$ *(for* $\alpha \in X^*$*) and* $\mathrm{fo}(A) \subseteq \Sigma_\varepsilon$ *(for* $A \in N$*) are the least sets such that:*

1. $\mathrm{fi}(Y)$ *for* $Y \in X$*:*
   - $Y \in \Sigma \implies \mathrm{fi}(Y) = \{Y\}$
   - $Y \to A_1 \ldots A_k Z\alpha \in P, k \in \mathbb{N}, Z \in X, \varepsilon \in \mathrm{fi}(A_1) \cap \ldots \cap \mathrm{fi}(A_k)$, $a \in \mathrm{fi}(Z) \implies a \in \mathrm{fi}(Y)$
   - $Y \to A_1 \ldots A_k \in P, k \in \mathbb{N}, \varepsilon \in \mathrm{fi}(A_1) \cap \ldots \cap \mathrm{fi}(A_k) \implies \varepsilon \in \mathrm{fi}(Y)$
2. $\mathrm{fi}(Y_1 \ldots Y_n)$ *for* $n \in \mathbb{N}, Y_i \in X$*:*
   - $\varepsilon \in \mathrm{fi}(Y_1 \ldots Y_{k-1}), a \in \mathrm{fi}(Y_k), k \in [n] \implies a \in \mathrm{fi}(Y_1 \ldots Y_n)$
   - $\varepsilon \in \mathrm{fi}(Y_1) \cap \ldots \cap \mathrm{fi}(Y_n) \implies \varepsilon \in \mathrm{fi}(Y_1 \ldots Y_n)$

Computing Lookahead Sets I

(see Waite/Goos: *Compiler Construction*, p. 164f)

Lemma 8.10 (Computation of fi/fo)

The sets $\text{fi}(\alpha) \subseteq \Sigma_\varepsilon$ *(for* $\alpha \in X^*$*) and* $\text{fo}(A) \subseteq \Sigma_\varepsilon$ *(for* $A \in N$*) are the least sets such that:*

1. $\text{fi}(Y)$ *for* $Y \in X$*:*
   - $Y \in \Sigma \implies \text{fi}(Y) = \{Y\}$
   - $Y \to A_1 \dots A_k Z\alpha \in P, k \in \mathbb{N}, Z \in X, \varepsilon \in \text{fi}(A_1) \cap \dots \cap \text{fi}(A_k),$ $a \in \text{fi}(Z) \implies a \in \text{fi}(Y)$
   - $Y \to A_1 \dots A_k \in P, k \in \mathbb{N}, \varepsilon \in \text{fi}(A_1) \cap \dots \cap \text{fi}(A_k) \implies \varepsilon \in \text{fi}(Y)$
2. $\text{fi}(Y_1 \dots Y_n)$ *for* $n \in \mathbb{N}, Y_i \in X$*:*
   - $\varepsilon \in \text{fi}(Y_1 \dots Y_{k-1}), a \in \text{fi}(Y_k), k \in [n] \implies a \in \text{fi}(Y_1 \dots Y_n)$
   - $\varepsilon \in \text{fi}(Y_1) \cap \dots \cap \text{fi}(Y_n) \implies \varepsilon \in \text{fi}(Y_1 \dots Y_n)$
3. $\text{fo}(A)$ *for* $A \in N$*:*
   - $\varepsilon \in \text{fo}(S)$
   - $A \to \alpha B\beta \in P, a \in \text{fi}(\beta) \implies a \in \text{fo}(B)$
   - $A \to \alpha B\beta \in P, \varepsilon \in \text{fi}(\beta), x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \text{fi}(A)$
2. $A \to B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
4. $\text{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \text{fi}(A)$
2. $A \to B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
4. $\text{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$
\begin{aligned}
G_{AE}: \quad E &\to E\texttt{+}T \mid T \\
T &\to T\texttt{*}F \mid F \\
F &\to \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b}
\end{aligned}
$$

Corollary 8.11

1. $A \rightarrow a\alpha \in P \implies a \in \mathrm{fi}(A)$
2. $A \rightarrow B\alpha \in P, a \in \mathrm{fi}(B) \implies a \in \mathrm{fi}(A)$
3. $A \rightarrow \varepsilon \in P \implies \varepsilon \in \mathrm{fi}(A)$
4. $\mathrm{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \mathrm{fi}(A) \implies a \in \mathrm{fi}(A\alpha)$
6. $A \rightarrow \alpha B \in P, x \in \mathrm{fo}(A) \implies x \in \mathrm{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$
\begin{aligned}
G_{AE}: \quad E &\rightarrow E\texttt{+}T \mid T \\
T &\rightarrow T\texttt{*}F \mid F \\
F &\rightarrow \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b}
\end{aligned}
$$

- $F \rightarrow \texttt{a} \in P \implies \texttt{a} \in \mathrm{fi}(F)$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \mathrm{fi}(A)$
2. $A \to B\alpha \in P, a \in \mathrm{fi}(B) \implies a \in \mathrm{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \mathrm{fi}(A)$
4. $\mathrm{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \mathrm{fi}(A) \implies a \in \mathrm{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \mathrm{fo}(A) \implies x \in \mathrm{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$G_{AE}: \quad \begin{aligned} E &\to E\texttt{+}T \mid T \\ T &\to T\texttt{*}F \mid F \\ F &\to \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b} \end{aligned}$$

- $F \to \texttt{a} \in P \implies \texttt{a} \in \mathrm{fi}(F)$
- $T \to F \in P, \texttt{a} \in \mathrm{fi}(F) \implies \texttt{a} \in \mathrm{fi}(T)$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \text{fi}(A)$
2. $A \to B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
4. $\text{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$\begin{aligned} G_{AE}: \quad E &\to E\texttt{+}T \mid T \\ T &\to T\texttt{*}F \mid F \\ F &\to \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b} \end{aligned}$$

- $F \to \texttt{a} \in P \implies \texttt{a} \in \text{fi}(F)$
- $T \to F \in P, \texttt{a} \in \text{fi}(F) \implies \texttt{a} \in \text{fi}(T)$
- $\texttt{a} \in \text{fi}(T)$

  $\implies \text{la}(T \to T\texttt{*}F) = \text{fi}(T\texttt{*}F \cdot \text{fo}(T)) \ni \texttt{a}$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \text{fi}(A)$
2. $A \to B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
4. $\text{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$G_{AE}: \quad \begin{aligned} E &\to E\texttt{+}T \mid T \\ T &\to T\texttt{*}F \mid F \\ F &\to \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b} \end{aligned}$$

- $F \to \texttt{a} \in P \implies \texttt{a} \in \text{fi}(F)$
- $T \to F \in P, \texttt{a} \in \text{fi}(F) \implies \texttt{a} \in \text{fi}(T)$
- $\texttt{a} \in \text{fi}(T)$
  $\implies \text{la}(T \to T\texttt{*}F) = \text{fi}(T\texttt{*}F \cdot \text{fo}(T)) \ni \texttt{a}$
- $\texttt{a} \in \text{fi}(F)$
  $\implies \text{la}(T \to F) = \text{fi}(F \cdot \text{fo}(T)) \ni \texttt{a}$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \mathrm{fi}(A)$
2. $A \to B\alpha \in P, a \in \mathrm{fi}(B) \implies a \in \mathrm{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \mathrm{fi}(A)$
4. $\mathrm{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \mathrm{fi}(A) \implies a \in \mathrm{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \mathrm{fo}(A) \implies x \in \mathrm{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$G_{AE}: \quad \begin{array}{lcl} E & \to & E\texttt{+}T \mid T \\ T & \to & T\texttt{*}F \mid F \\ F & \to & \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b} \end{array}$$

- $F \to \texttt{a} \in P \implies \texttt{a} \in \mathrm{fi}(F)$
- $T \to F \in P, \texttt{a} \in \mathrm{fi}(F) \implies \texttt{a} \in \mathrm{fi}(T)$
- $\texttt{a} \in \mathrm{fi}(T)$
  $\implies \mathrm{la}(T \to T\texttt{*}F) = \mathrm{fi}(T\texttt{*}F \cdot \mathrm{fo}(T)) \ni \texttt{a}$
- $\texttt{a} \in \mathrm{fi}(F)$
  $\implies \mathrm{la}(T \to F) = \mathrm{fi}(F \cdot \mathrm{fo}(T)) \ni \texttt{a}$
- $\implies \texttt{a} \in \mathrm{la}(T \to T\texttt{*}F) \cap \mathrm{la}(T \to F) \neq \emptyset$

Computing Lookahead Sets II

Corollary 8.11

1. $A \to a\alpha \in P \implies a \in \text{fi}(A)$
2. $A \to B\alpha \in P, a \in \text{fi}(B) \implies a \in \text{fi}(A)$
3. $A \to \varepsilon \in P \implies \varepsilon \in \text{fi}(A)$
4. $\text{fi}(\varepsilon) = \{\varepsilon\}$
5. $a \in \text{fi}(A) \implies a \in \text{fi}(A\alpha)$
6. $A \to \alpha B \in P, x \in \text{fo}(A) \implies x \in \text{fo}(B)$

Example 8.12

Grammar for arithmetic expressions (cf. Example 7.3):

$$G_{AE}: \quad \begin{array}{rcl} E & \to & E\texttt{+}T \mid T \\ T & \to & T\texttt{*}F \mid F \\ F & \to & \texttt{(}E\texttt{)} \mid \texttt{a} \mid \texttt{b} \end{array}$$

- $F \to \texttt{a} \in P \implies \texttt{a} \in \text{fi}(F)$
- $T \to F \in P, \texttt{a} \in \text{fi}(F) \implies \texttt{a} \in \text{fi}(T)$
- $\texttt{a} \in \text{fi}(T)$
  $\implies \text{la}(T \to T\texttt{*}F) = \text{fi}(T\texttt{*}F \cdot \text{fo}(T)) \ni \texttt{a}$
- $\texttt{a} \in \text{fi}(F)$
  $\implies \text{la}(T \to F) = \text{fi}(F \cdot \text{fo}(T)) \ni \texttt{a}$
- $\implies \texttt{a} \in \text{la}(T \to T\texttt{*}F) \cap \text{la}(T \to F) \neq \emptyset$
- $\implies G_{AE} \notin LL(1)$