Script generated by TTT

Title: Simon: Compilerbau (19.05.2014)

Date: Mon May 19 14:27:34 CEST 2014

Duration: 93:26 min

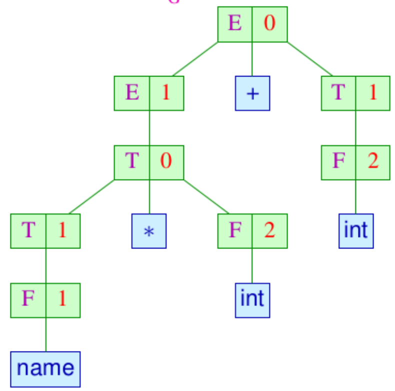
Pages: 69

Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
counter $* \; 2 + 40$

Pushdown:
$(\; q_0 \;)$

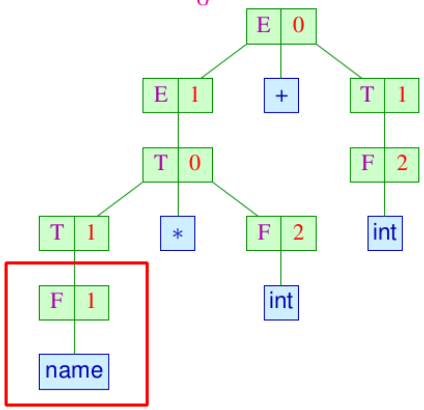


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
$* \; 2 + 40$

Pushdown:
$(\; q_0 \; \mathit{name} \;)$

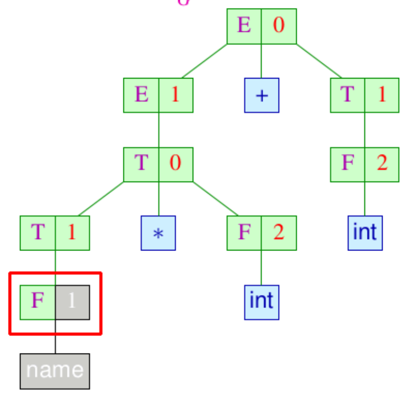


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
$* \; 2 + 40$

Pushdown:
$(\; q_0 \; F \;)$



Reverse Rightmost Derivations in Shift-Reduce-Parsers

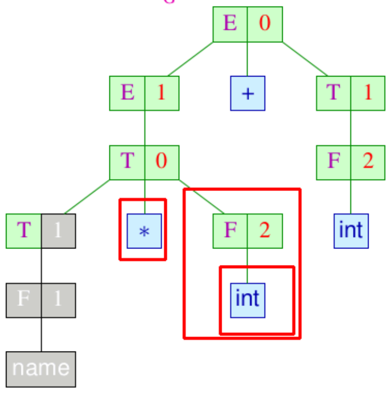
Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

$* \, 2 + 40$

Pushdown:

$(\, q_0 \, T \,)$



Reverse Rightmost Derivations in Shift-Reduce-Parsers

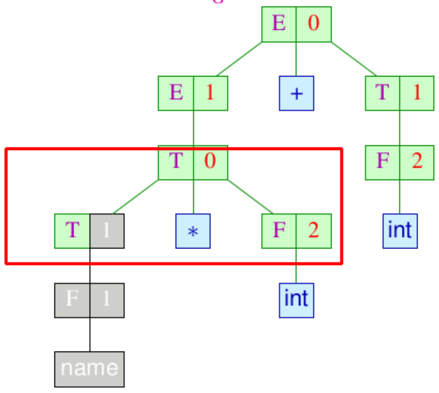
Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

$2 + 40$

Pushdown:

$(\, q_0 \, T \, * \,)$



Reverse Rightmost Derivations in Shift-Reduce-Parsers

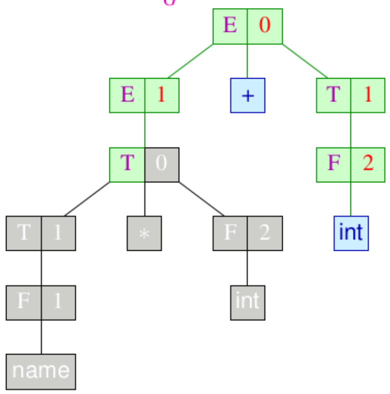
Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

$+ \, 40$

Pushdown:

$(\, q_0 \, T \,)$



Reverse Rightmost Derivations in Shift-Reduce-Parsers

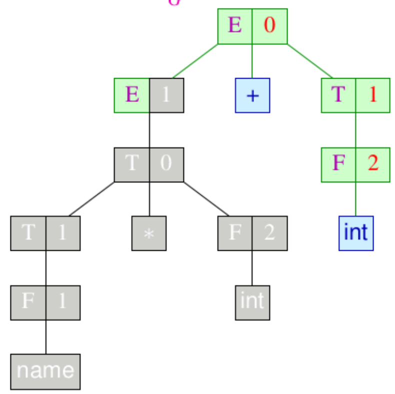
Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

$+ \, 40$

Pushdown:

$(\, q_0 \, E \,)$

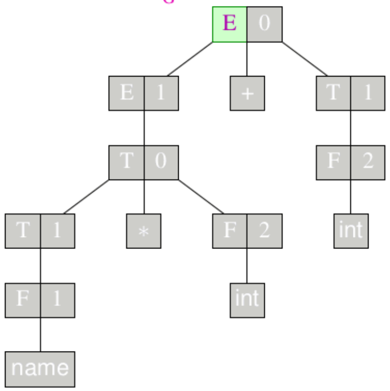


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

Pushdown:
(f)

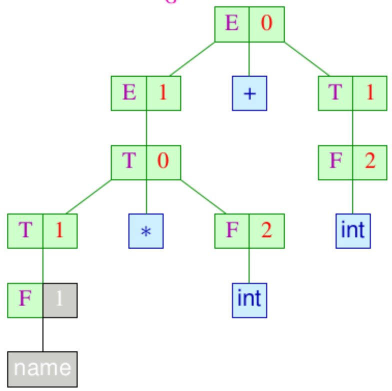


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
$* \; 2 + 40$

Pushdown:
$(q_0 \; F)$

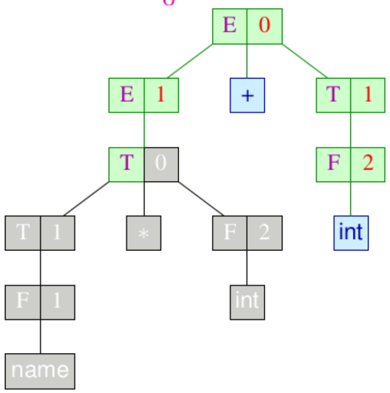


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
$+ \; 40$

Pushdown:
$(q_0 \; T)$

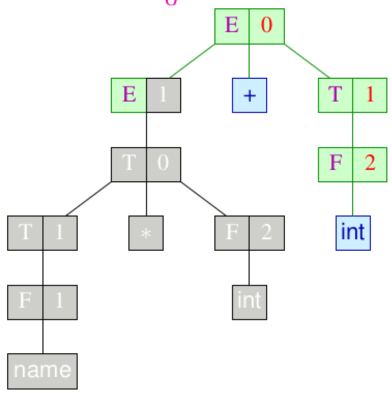


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
$+ \; 40$

Pushdown:
$(q_0 \; E)$

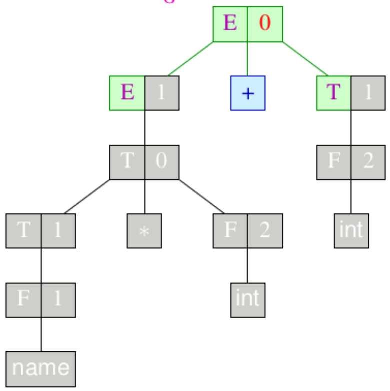


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

Pushdown:
$(q_0 \; E + T)$

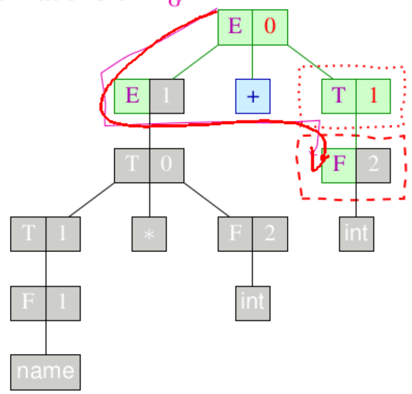


Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

Pushdown:
$(q_0 \; E + F)$



Generic Observation:
In a sequence of configurations of M_G^R

$$(q_0 \, \alpha \, \gamma, \; v) \vdash (q_0 \, \alpha \, B, \; v) \vdash^* (q_0 \, S, \; \epsilon)$$

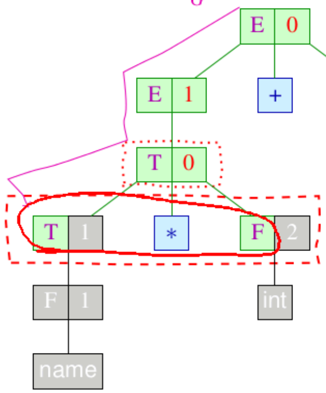
we call $\alpha \, \gamma$ a viable prefix for the complete item $[B \to \gamma \bullet]$.

Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:
$+ \; 40$

Pushdown:
$(q_0 \; T * F)$



Generic Observation:
In a sequence of configurations of M_G^R

$$(q_0 \, \alpha \, \gamma, \; v) \vdash (q_0 \, \alpha \, B, \; v) \vdash^* (q_0 \, S, \; \epsilon)$$

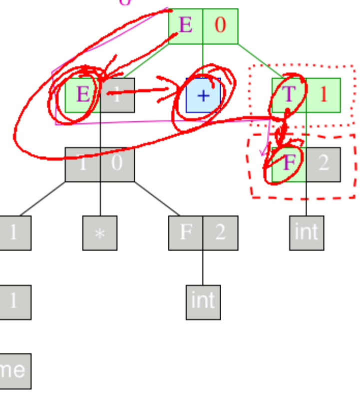
we call $\alpha \, \gamma$ a viable prefix for the complete item $[B \to \gamma \bullet]$.

Reverse Rightmost Derivations in Shift-Reduce-Parsers

Idea: Observe *reverse rightmost*-derivations of M_G^R!

Input:

Pushdown:
$(q_0 \; E + F)$



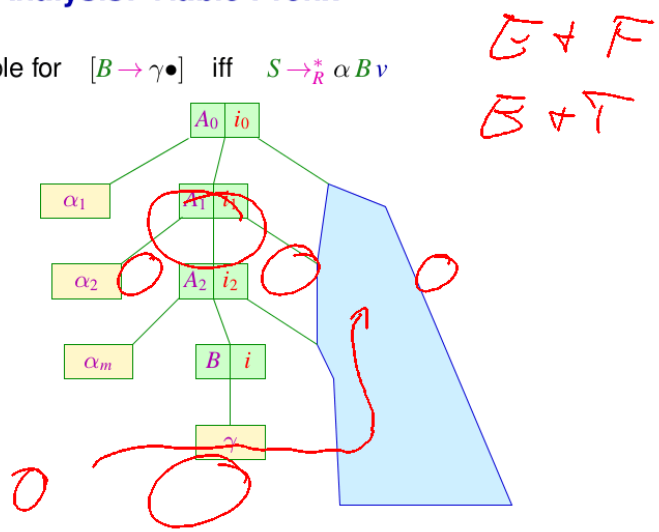
Generic Observation:
In a sequence of configurations of M_G^R

$$(q_0 \, \alpha \, \gamma, \; v) \vdash (q_0 \, \alpha \, B, \; v) \vdash^* (q_0 \, S, \; \epsilon)$$

we call $\alpha \, \gamma$ a viable prefix for the complete item $[B \to \gamma \bullet]$.

Bottom-up Analysis: Viable Prefix

$\alpha\,\gamma$ is viable for $[B \to \gamma\bullet]$ iff $S \to_R^* \alpha\, B\, v$



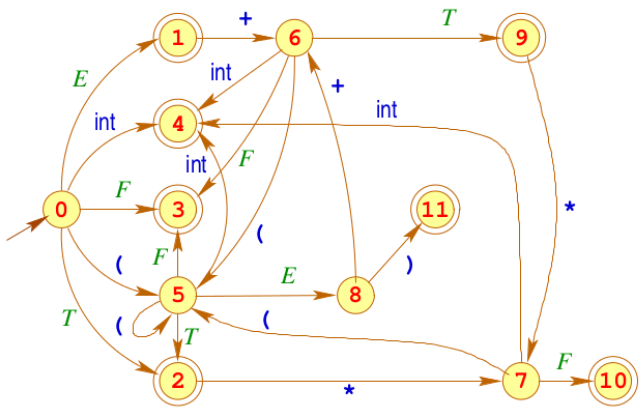
... with $\alpha = \alpha_1 \ldots \alpha_m$

Canonical LR(0)-Automaton

The canonical $LR(0)$-automaton $LR(G)$ is created from $c(G)$ by:

1. performing arbitrarily many ϵ-transitions after every consuming transition
2. performing the powerset construction

... for example:



Canonical LR(0)-Automaton

Example:

$$
\begin{array}{lcll}
E & \to & E+T & \mid\; T \\
T & \to & T*F & \mid\; F \\
F & \to & (\,E\,) & \mid\; \text{int}
\end{array}
$$

Therefore we determine:

LR(0)-Parser

... for example:

$q_1 = \{[S' \to E\,\bullet],$
$\quad [E \to E\,\bullet\,+T]\}$

$q_2 = \{[E \to T\,\bullet],$
$\quad [T \to T\,\bullet\,*F]\}$

$q_9 = \{[E \to E+T\,\bullet],$
$\quad [T \to T\,\bullet\,*F]\}$

$q_3 = \{[T \to F\,\bullet]\}$

$q_{10} = \{[T \to T*F\,\bullet]\}$

$q_4 = \{[F \to \text{int}\,\bullet]\}$

$q_{11} = \{[F \to (\,E\,)\,\bullet]\}$

The final states q_1, q_2, q_9 contain more then one admissible item

$\Rightarrow$ non deterministic!

LR(0)-Parser

... for example:

$$q_1 = \{[S' \to E\,\bullet],\ [E \to E\,\bullet\,+T]\}$$

$$q_2 = \{[E \to T\,\bullet],\ [T \to T\,\bullet * F]\} \qquad q_9 = \{[E \to E+T\,\bullet],\ [T \to T\,\bullet * F]\}$$

$$q_3 = \{[T \to F\,\bullet]\} \qquad q_{10} = \{[T \to T * F\,\bullet]\}$$

$$q_4 = \{[F \to \text{int}\,\bullet]\} \qquad q_{11} = \{[F \to (\,E\,)\,\bullet]\}$$

The final states q_1, q_2, q_9 contain more then one admissible item

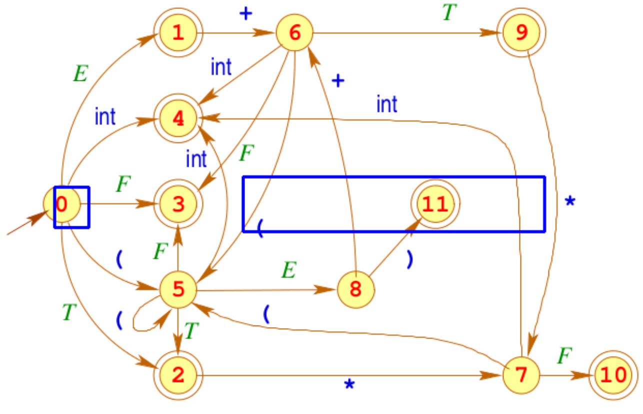
$\Rightarrow$ non deterministic!

Canonical LR(0)-Automaton

The canonical $LR(0)$-automaton $LR(G)$ is created from $c(G)$ by:

1. performing arbitrarily many ϵ-transitions after every consuming transition
2. performing the powerset construction

... for example:



LR(0)-Parser

Correctness:

we show:

The accepting computations of an $LR(0)$-parser are one-to-one related to those of a shift-reduce parser M_G^R.

we conclude:

- The accepted language is exactly $\mathcal{L}(G)$
- The sequence of reductions of an accepting computation for a word $w \in T$ yields a reverse rightmost derivation of G for w

Revisiting the Conflicts of the LR(0)-Automaton

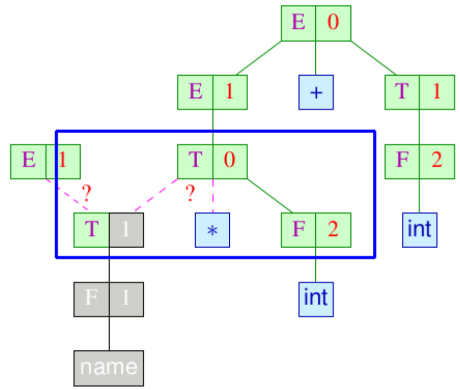
What differenciates the particular Reductions and Shifts?

Input:

$* \, 2 + 40$

Pushdown:

$(\, q_0 \; T \,)$



$$E \to E+T \mid T$$
$$T \to T*F \mid F$$
$$F \to (\,E\,) \mid \text{int}$$

Revisiting the Conflicts of the LR(0)-Automaton

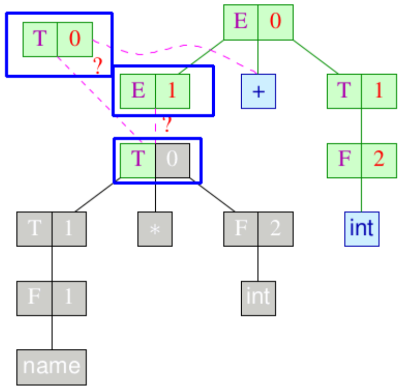
What differenciates the particular Reductions and Shifts?

Input:

$+\ 40$

Pushdown:

$(\ q_0\ T\ )$



$$
\begin{array}{lcl}
E & \to & E+T \quad | \quad T \\
T & \to & T*F \quad | \quad F \\
F & \to & (\ E\ ) \quad | \quad \text{int}
\end{array}
$$

Revisiting the Conflicts of the LR(0)-Automaton

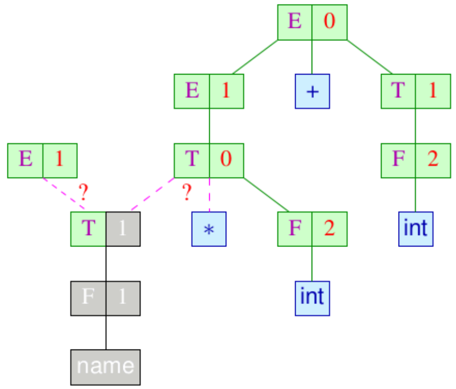
What differenciates the particular Reductions and Shifts?

Input:

$*\ 2 + 40$

Pushdown:

$(\ q_0\ T\ )$



$$
\begin{array}{lcl}
E & \to & E+T \quad | \quad T \\
T & \to & T*F \quad | \quad F \\
F & \to & (\ E\ ) \quad | \quad \text{int}
\end{array}
$$

Revisiting the Conflicts of the LR(0)-Automaton

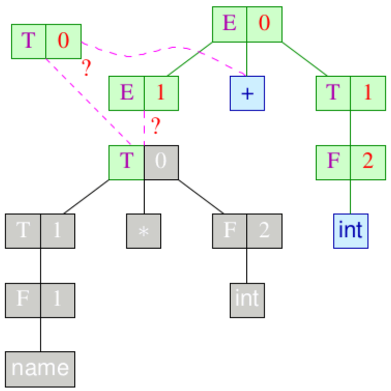
What differenciates the particular Reductions and Shifts?

Input:

$+\ 40$

Pushdown:

$(\ q_0\ T\ )$



$$
\begin{array}{lcl}
E & \to & E+T \quad | \quad T \\
T & \to & T*F \quad | \quad F \\
F & \to & (\ E\ ) \quad | \quad \text{int}
\end{array}
$$

Revisiting the Conflicts of the LR(0)-Automaton

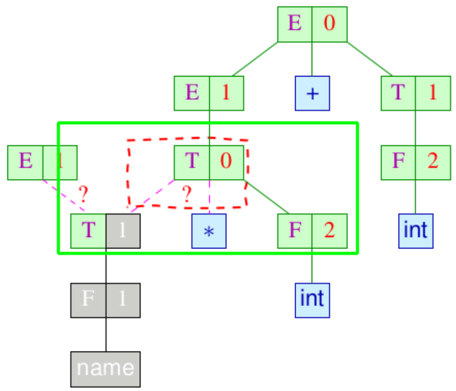
Idea: Matching lookahead with *right context* matters!

Input:

$*\ 2 + 40$

Pushdown:

$(\ q_0\ T\ )$



$$
\begin{array}{lcl}
E & \to & E+T \quad | \quad T \\
T & \to & T*F \quad | \quad F \\
F & \to & (\ E\ ) \quad | \quad \text{int}
\end{array}
$$

Revisiting the Conflicts of the LR(0)-Automaton

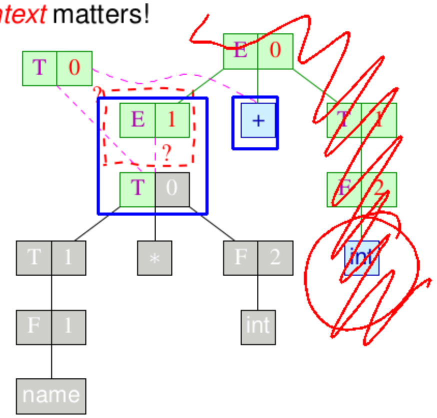
Idea: Matching lookahead with *right context* matters!

Input:

$+\ 40$

Pushdown:

$(\ q_0\ T\ )$



$$E \to E + T \mid T$$
$$T \to T * F \mid F$$
$$F \to (\ E\ ) \mid \text{int}$$

LR(k)-Grammars

Idea: Consider k-lookahead in conflict situations.

Definition:

The reduced contextfree grammar G is called $LR(k)$-grammar, if for $\text{First}_k(w) = \text{First}_k(x)$ with:

$$\left.\begin{array}{lll} S \to_R^* & \alpha\, A\, w & \to \alpha\, \beta\, w \\ S \to_R^* & \alpha'\, A'\, w' & \to \alpha\, \beta\, x \end{array}\right\} \text{follows: } \alpha = \alpha' \wedge A = A' \wedge w' = x$$

LR(k)-Grammars

Idea: Consider k-lookahead in conflict situations.

Definition:

The reduced contextfree grammar G is called $LR(k)$-grammar, if for $\text{First}_k(w) = \text{First}_k(x)$ with:

$$\left.\begin{array}{lll} S \to_R^* & \alpha\, A\, w & \to \alpha\, \beta\, w \\ S \to_R^* & \alpha'\, A'\, w' & \to \alpha\, \beta\, x \end{array}\right\} \text{follows: } \alpha = \alpha' \wedge A = A' \wedge w' = x$$

Strategy for testing Grammars for $LR(k)$-property

1. Focus iteratively on all rightmost derivations $S \to_R^* \alpha\, X\, w \to \alpha\, \beta\, w$
2. Identify handle $\alpha\, \beta$ in sentence forms $\alpha\, \beta\, w$
3. Determine minimal k, such that $\text{First}_k(w)$ associates β with a unique $X \to \beta$ for non-prefixfree $\alpha\, \beta$s

LR(k)-Grammars

for example:

(1) $S \to A \mid B \qquad A \to a\, A\, b \mid 0 \qquad B \to a\, B\, b\, b \mid 1$

LR(k)-Grammars

for example:

(1) $S \to A \mid B \qquad A \to a\,A\,b \mid 0 \qquad B \to a\,B\,b\,b \mid 1$

... is not $LL(k)$ for any k :

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$$\underline{A}\,,\ \underline{B}\,,\ a^n\,\underline{a\,A\,b}\,,\ a^n\,\underline{a\,B\,b\,b}\,,\ a^n\,\underline{0}\,,\ a^n\,\underline{1} \qquad (n \geq 0)$$

LR(k)-Grammars

for example:

(1) $S \to A \mid B \qquad A \to a\,A\,b \mid 0 \qquad B \to a\,B\,b\,b \mid 1$

... is not $LL(k)$ for any k — but $LR(0)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$$\underline{A}\,,\ \underline{B}\,,\ a^n\,\underline{a\,A\,b}\,,\ a^n\,\underline{a\,B\,b\,b}\,,\ a^n\,\underline{0}\,,\ a^n\,\underline{1} \qquad (n \geq 0)$$

LR(k)-Grammars

for example:

(1) $S \to A \mid B \qquad A \to a\,A\,b \mid 0 \qquad B \to a\,B\,b\,b \mid 1$

... is not $LL(k)$ for any k — but $LR(0)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$$\underline{A}\,,\ \underline{B}\,,\ a^n\,\underline{a\,A\,b}\,,\ a^n\,\underline{a\,B\,b\,b}\,,\ a^n\,\underline{0}\,,\ a^n\,\underline{1} \qquad (n \geq 0)$$

(2) $S \to a\,A\,c \qquad A \to A\,b\,b \mid b$

LR(k)-Grammars

for example:

(1) $S \to A \mid B \qquad A \to a\,A\,b \mid 0 \qquad B \to a\,B\,b\,b \mid 1$

... is not $LL(k)$ for any k — but $LR(0)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$$\underline{A}\,,\ \underline{B}\,,\ a^n\,\underline{a\,A\,b}\,,\ a^n\,\underline{a\,B\,b\,b}\,,\ a^n\,\underline{0}\,,\ a^n\,\underline{1} \qquad (n \geq 0)$$

(2) $S \to a\,A\,c \qquad A \to A\,b\,b \mid b$

... is also not $LL(k)$ for any k :

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$\underline{aA}c$, $\underline{aAbb}b^n c$, $\underline{ab}b^n c$

LR(k)-Grammars

for example:

(1) $S \to A \mid B \qquad A \to a\,A\,b \mid 0 \qquad B \to a\,B\,b\,b \mid 1$

... is not $LL(k)$ for any k — but $LR(0)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$$\underline{A}\,,\ \underline{B}\,,\ a^n\,\underline{a\,A\,b}\,,\ a^n\,\underline{a\,B\,b\,b}\,,\ a^n\,\underline{0}\,,\ a^n\,\underline{1} \qquad (n \geq 0)$$

(2) $S \to a\,A\,c \qquad A \to A\,b\,b \mid b$

... is also not $LL(k)$ for any k :

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Then $\alpha\,\underline{\beta}$ is of one of these forms:

$$a\,\underline{b}\,,\ a\,\underline{A\,b\,b}\,,\ \underline{a\,A\,c}$$

LR(k)-Grammars

for example:

(3) $S \to a\,A\,c \qquad A \to b\,b\,A \mid b$

LR(k)-Grammars

for example:

(3) $S \to a\,A\,c \qquad A \to b\,b\,A \mid b$

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$ with $\{y\} = \text{First}_k(w)$ then $\alpha\,\underline{\beta}\,y$ is of one of these forms:

LR(k)-Grammars

for example:

(3) $S \to a\,A\,c \qquad A \to b\,b\,A \mid b$

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$ with $\{y\} = \text{First}_k(w)$ then $\alpha\,\underline{\beta}\,y$ is of one of these forms:

$$a\,b^{2n}\,\underline{b\,c}\,,\ a\,b^{2n}\,\underline{b\,b\,A}\,c\,,\ \underline{a\,A\,c}$$

LR(k)-Grammars

for example:

(3) $S \to a\,A\,c \qquad A \to b\,b\,A \mid b$ ... is not $LR(0)$, but $LR(1)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$ with $\{y\} = \text{First}_k(w)$ then $\alpha\,\underline{\beta}\,y$ is of one of these forms:

$$a\,b^{2n}\,\underline{b}\,c\;,\;\; a\,b^{2n}\,\underline{b\,b\,A}\,c\;,\;\; \underline{a\,A\,c}$$

(4) $S \to a\,A\,c \qquad A \to b\,A\,b \mid b$

LR(k)-Grammars

for example:

(3) $S \to a\,A\,c \qquad A \to b\,b\,A \mid b$ ... is not $LR(0)$, but $LR(1)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$ with $\{y\} = \text{First}_k(w)$ then $\alpha\,\underline{\beta}\,y$ is of one of these forms:

$$a\,b^{2n}\,\underline{b}\,c\;,\;\; a\,b^{2n}\,\underline{b\,b\,A}\,c\;,\;\; \underline{a\,A\,c}$$

(4) $S \to a\,A\,c \qquad A \to b\,A\,b \mid b$

Consider the rightmost derivations:

$$S \to_R^* a\,b^n\,A\,b^n\,c \to a\,b^n\,\underline{b}\,b^n\,c$$

LR(k)-Grammars

for example:

(3) $S \to a\,A\,c \qquad A \to b\,b\,A \mid b$ ... is not $LR(0)$, but $LR(1)$:

Let $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$ with $\{y\} = \text{First}_k(w)$ then $\alpha\,\underline{\beta}\,y$ is of one of these forms:

$$a\,b^{2n}\,\underline{b}\,c\;,\;\; a\,b^{2n}\,\underline{b\,b\,A}\,c\;,\;\; \underline{a\,A\,c}$$

bAb

(4) $S \to a\,A\,c \qquad A \to b\,A\,b \mid b$ ... is not $LR(k)$ for any $k \geq 0$:

Consider the rightmost derivations:

$$S \to_R^* a\,b^n\,A\,b^n\,c \to a\,b^n\,\underline{b}\,b^n\,c$$

LR(1)-Parsing

Idea: Let's equip items with 1-lookahead

Definition LR(1)-Item

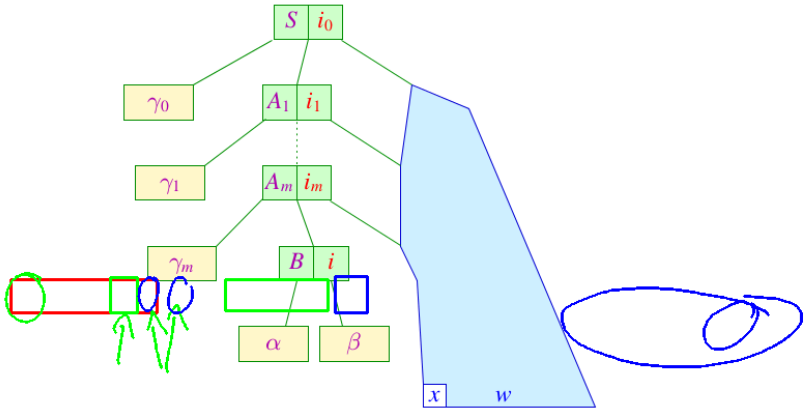
An $LR(1)$-item is a pair $[B \to \alpha \bullet \beta, x]$ with

$$x \in \text{Follow}_1(B) = \bigcup \{\text{First}_1(\nu) \mid S \to^* \mu\,B\,\nu\}$$

Admissible LR(1)-Items

The item $[B \to \alpha \bullet \beta,\ x]$ is *admissable* for $\gamma\,\alpha$ if:

$$S \to_R^* \gamma\, B\, w \qquad \text{with} \qquad \{x\} = \text{First}_1(w)$$



... with $\gamma_0 \dots \gamma_m = \gamma$

The Characteristic LR(1)-Automaton

The set of admissible $LR(1)$-items for viable prefixes is again computed with the help of the finite automaton $c(G, 1)$.

The automaton $c(G, 1)$:

States: $LR(1)$-items

Start state: $[S' \to \bullet\, S,\ \epsilon]$

Final states: $\{[B \to \gamma \bullet,\ x] \mid B \to \gamma \in P,\ x \in \text{Follow}_1(B)\}$

Transitions:

(1) $([A \to \alpha \bullet X\, \beta,\ x], X, [A \to \alpha\, X \bullet \beta,\ x]), \quad X \in (N \cup T)$

(2) $([A \to \alpha \bullet B\, \beta,\ x], \epsilon, [B \to \bullet\, \gamma,\ x']),$
$A \to \alpha\, B\, \beta\,,\ B \to \gamma \in P, x' \in \text{First}_1(\beta) \odot \{x\};$

This automaton works like $c(G)$ — but additionally manages a 1-prefix from Follow_1 of the left-hand sides.

The Canonical LR(1)-Automaton

The canonical $LR(1)$-automaton $LR(G, 1)$ is created from $c(G, 1)$, by performing arbitrarily many ϵ-transitions and then making the resulting automaton deterministic ...

But again, it can be constructed directly from the grammar; analoguously to $LR(0)$, we need the ϵ-closure δ_ϵ^* as a helper function:

$$\delta_\epsilon^*(q) = q \cup \{[C \to \bullet\, \gamma,\ x] \mid \exists\, [A \to \alpha \bullet B\, \beta',\ x'] \in q, \beta \in (N \cup T)^* : B \to^* C\, \beta \,\wedge\, x \in \text{First}_1(\beta\, \beta') \odot \{x'\}\}$$

Then, we define:

States: Sets of $LR(1)$-items;

Start state: $\delta_\epsilon^* \{[S' \to \bullet\, S,\ \epsilon]\}$

Final states: $\{q \mid \exists A \to \alpha \in P :\ [A \to \alpha \bullet,\ x] \in q\}$

Transitions: $\delta(q, X) = \delta_\epsilon^* \{[A \to \alpha\, X \bullet \beta,\ x] \mid [A \to \alpha \bullet X\, \beta,\ x] \in q\}$

The Canonical LR(1)-Automaton

For example:

$$\begin{array}{lcl} E & \to & E + T \mid T \\ T & \to & T * F \mid F \\ F & \to & (\, E\,) \mid \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

$$\begin{array}{lclcl} q_0 & = & & & \{[S' \to \bullet\, E,\ \{\epsilon\}], \\ & & & & [E \to \bullet\, E + T,\ \{\epsilon, +\}], \\ & & & & [E \to \bullet\, T,\ \{\epsilon, +\}], \\ & & & & [T \to \bullet\, T * F,\ \{\epsilon, +, *\}], \\ & & & & [T \to \bullet\, F,\ \{\epsilon, +, *\}], \\ & & & & [F \to \bullet\, (\, E\,),\ \{\epsilon, +, *\}], \\ & & & & [F \to \bullet\, \text{int},\ \{\epsilon, +, *\}]\} \\ q_1 & = & \delta(q_0, E) & = & \{[S' \to E \bullet \quad], \\ & & & & [E \to E \bullet + T \quad]\} \\ q_2 & = & \delta(q_0, T) & = & \{[E \to T \bullet \quad], \\ & & & & [T \to T \bullet * F \quad]\} \end{array}$$

$$\begin{array}{lclcl} q_3 & = & \delta(q_0, F) & = & \{[T \to F \bullet \quad]\} \\ q_4 & = & \delta(q_0, \text{int}) & & \{[F \to \text{int} \bullet \quad]\} \\ q_5 & = & \delta(q_0, (\,) & = & \{[F \to (\, \bullet\, E\,) \quad], \\ & & & & [E \to \bullet\, E + T \quad], \\ & & & & [E \to \bullet\, T \quad], \\ & & & & [T \to \bullet\, T * F \quad] \\ & & & & [T \to \bullet\, F \quad], \\ & & & & [F \to \bullet\, (\, E\,) \quad] \\ & & & & [F \to \bullet\, \text{int} \quad]\} \end{array}$$

The Canonical LR(1)-Automaton

For example:

$$\begin{array}{lcll} E & \to & E+T & \mid\ T \\ T & \to & T*F & \mid\ F \\ F & \to & (\,E\,) & \mid\ \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

$$\begin{array}{lclcl}
q_0 & = & & & \{[S' \to \bullet\, E,\ \{\epsilon\}],\ [E \to \bullet\, E+T,\ \{\epsilon,+\}],\ [E \to \bullet\, T,\ \{\epsilon,+\}],\ [T \to \bullet\, T*F,\ \{\epsilon,+,*\}],\ [T \to \bullet\, F,\ \{\epsilon,+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\epsilon,+,*\}],\ [F \to \bullet\, \text{int},\ \{\epsilon,+,*\}]\} \\
q_1 & = & \delta(q_0,E) & = & \{[S' \to E\,\bullet,\ \{\epsilon\}],\ [E \to E\,\bullet + T,\ \{\epsilon,+\}]\} \\
q_2 & = & \delta(q_0,T) & = & \{[E \to T\,\bullet,\ \{\epsilon,+\}],\ [T \to T\,\bullet * F,\ \{\epsilon,+,*\}]\} \\
q_3 & = & \delta(q_0,F) & = & \{[T \to F\,\bullet,\ \{\epsilon,+,*\}]\} \\
q_4 & = & \delta(q_0,\text{int}) & & \{[F \to \text{int}\,\bullet,\ \{\epsilon,+,*\}]\} \\
q_5 & = & \delta(q_0,\,(\,) & = & \{[F \to (\ \bullet\, E\,)\quad],\ [E \to \bullet\, E+T\quad],\ [E \to \bullet\, T\quad],\ [T \to \bullet\, T*F\quad],\ [T \to \bullet\, F\quad],\ [F \to \bullet\, (\,E\,)\quad],\ [F \to \bullet\, \text{int}\quad]\}
\end{array}$$

The Canonical LR(1)-Automaton

For example:

$$\begin{array}{lcll} E & \to & E+T & \mid\ T \\ T & \to & T*F & \mid\ F \\ F & \to & (\,E\,) & \mid\ \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

$$\begin{array}{lclcl}
q_0 & = & & & \{[S' \to \bullet\, E,\ \{\epsilon\}],\ [E \to \bullet\, E+T,\ \{\epsilon,+\}],\ [E \to \bullet\, T,\ \{\epsilon,+\}],\ [T \to \bullet\, T*F,\ \{\epsilon,+,*\}],\ [T \to \bullet\, F,\ \{\epsilon,+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\epsilon,+,*\}],\ [F \to \bullet\, \text{int},\ \{\epsilon,+,*\}]\} \\
q_1 & = & \delta(q_0,E) & = & \{[S' \to E\,\bullet,\ \{\epsilon\}],\ [E \to E\,\bullet + T,\ \{\epsilon,+\}]\} \\
q_2 & = & \delta(q_0,T) & = & \{[E \to T\,\bullet,\ \{\epsilon,+\}],\ [T \to T\,\bullet * F,\ \{\epsilon,+,*\}]\} \\
q_3 & = & \delta(q_0,F) & = & \{[T \to F\,\bullet,\ \{\epsilon,+,*\}]\} \\
q_4 & = & \delta(q_0,\text{int}) & & \{[F \to \text{int}\,\bullet,\ \{\epsilon,+,*\}]\} \\
q_5 & = & \delta(q_0,\,(\,) & = & \{[F \to (\ \bullet\, E\,),\ \{\epsilon,+,*\}],\ [E \to \bullet\, E+T,\ \{\,),+\}],\ [E \to \bullet\, T,\ \{\,),+\}],\ [T \to \bullet\, T*F,\ \{\,),+,*\}],\ [T \to \bullet\, F,\ \{\,),+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\,),+,*\}],\ [F \to \bullet\, \text{int},\ \{\,),+,*\}]\}
\end{array}$$

The Canonical LR(1)-Automaton

For example:

$$\begin{array}{lcll} E & \to & E+T & \mid\ T \\ T & \to & T*F & \mid\ F \\ F & \to & (\,E\,) & \mid\ \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

$$\begin{array}{lclcl}
q_5' & = & \delta(q_5,\,(\,) & = & \{[F \to (\ \bullet\, E\,),\ \{\,),+,*\}],\ [E \to \bullet\, E+T,\ \{\,),+\}],\ [E \to \bullet\, T,\ \{\,),+\}],\ [T \to \bullet\, T*F,\ \{\,),+,*\}],\ [T \to \bullet\, F,\ \{\,),+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\,),+,*\}],\ [F \to \bullet\, \text{int},\ \{\,),+,*\}]\} \\
q_6 & = & \delta(q_1,+) & = & \{[E \to E + \bullet\, T,\ \{\epsilon,+\}],\ [T \to \bullet\, T*F,\ \{\epsilon,+,*\}],\ [T \to \bullet\, F,\ \{\epsilon,+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\epsilon,+,*\}],\ [F \to \bullet\, \text{int},\ \{\epsilon,+,*\}]\} \\
q_7 & = & \delta(q_2,*) & = & \{[T \to T * \bullet\, F,\ \{\epsilon,+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\epsilon,+,*\}],\ [F \to \bullet\, \text{int},\ \{\epsilon,+,*\}]\} \\
q_8 & = & \delta(q_5,E) & = & \{[F \to (\,E\,\bullet\,),\ \{\epsilon,+,*\}]\}\ [E \to E\,\bullet + T,\ \{\,),+\}]\} \\
q_9 & = & \delta(q_6,T) & = & \{[E \to E+T\,\bullet,\ \{\epsilon,+\}],\ [T \to T\,\bullet * F,\ \{\epsilon,+,*\}]\} \\
q_{10} & = & \delta(q_7,F) & = & \{[T \to T*F\,\bullet,\ \{\epsilon,+,*\}]\} \\
q_{11} & = & \delta(q_8,\,)\,) & = & \{[F \to (\,E\,)\,\bullet,\ \{\epsilon,+,*\}]\}
\end{array}$$

The Canonical LR(1)-Automaton

For example:

$$\begin{array}{lcll} E & \to & E+T & \mid\ T \\ T & \to & T*F & \mid\ F \\ F & \to & (\,E\,) & \mid\ \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

$$\begin{array}{lclcl}
q_5' & = & \delta(q_5,\,(\,) & = & \{[F \to (\ \bullet\, E\,),\ \{\,),+,*\}],\ [E \to \bullet\, E+T,\ \{\,),+\}],\ [E \to \bullet\, T,\ \{\,),+\}],\ [T \to \bullet\, T*F,\ \{\,),+,*\}],\ [T \to \bullet\, F,\ \{\,),+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\,),+,*\}],\ [F \to \bullet\, \text{int},\ \{\,),+,*\}]\} \\
q_6 & = & \delta(q_1,+) & = & \{[E \to E + \bullet\, T,\ \{\epsilon,+\}],\ [T \to \bullet\, T*F,\ \{\epsilon,+,*\}],\ [T \to \bullet\, F,\ \{\epsilon,+,*\}],\ [F \to \bullet\, (\,E\,),\ \{\epsilon,+,*\}],\ [F \to \bullet\, \text{int},\ \{\epsilon,+,*\}]\} \\
q_7 & = & \delta(q_2,*) & = & \{[T \to T * \bullet\, F\quad],\ [F \to \bullet\, (\,E\,)\quad],\ [F \to \bullet\, \text{int}\quad]\} \\
q_8 & = & \delta(q_5,E) & = & \{[F \to (\,E\,\bullet\,)\quad]\}\ [E \to E\,\bullet + T\quad]\} \\
q_9 & = & \delta(q_6,T) & = & \{[E \to E+T\,\bullet\quad],\ [T \to T\,\bullet * F\quad]\} \\
q_{10} & = & \delta(q_7,F) & = & \{[T \to T*F\,\bullet\quad]\} \\
q_{11} & = & \delta(q_8,\,)\,) & = & \{[F \to (\,E\,)\,\bullet\quad]\}
\end{array}$$

The Canonical LR(1)-Automaton

For example:

$$\begin{array}{lcl} E & \to & E + T \quad | \quad T \\ T & \to & T * F \quad | \quad F \\ F & \to & (\, E \,) \quad | \quad \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

$$\begin{array}{lclcl} q_5' & = & \delta(q_5, () & = & \{[F \to (\, \bullet E\,) \quad], \\ & & & & [E \to \bullet E + T \quad], \\ & & & & [E \to \bullet T \quad], \\ & & & & [T \to \bullet T * F \quad], \\ & & & & [T \to \bullet F \quad], \\ & & & & [F \to \bullet (\, E\,) \quad], \\ & & & & [F \to \bullet \text{int} \quad]\} \\ q_6 & = & \delta(q_1, +) & = & \{[E \to E + \bullet T \quad], \\ & & & & [T \to \bullet T * F \quad], \\ & & & & [T \to \bullet F \quad], \\ & & & & [F \to \bullet (\, E\,) \quad], \\ & & & & [F \to \bullet \text{int} \quad]\} \end{array}$$

$$\begin{array}{lclcl} q_7 & = & \delta(q_2, *) & = & \{[T \to T * \bullet F \quad], \\ & & & & [F \to \bullet (\, E\,) \quad], \\ & & & & [F \to \bullet \text{int} \quad]\} \\ q_8 & = & \delta(q_5, E) & = & \{[F \to (\, E \bullet\,) \quad]\} \\ & & & & [E \to E \bullet + T \quad]\} \\ q_9 & = & \delta(q_6, T) & = & \{[E \to E + T \bullet \quad], \\ & & & & [T \to T \bullet * F \quad]\} \\ q_{10} & = & \delta(q_7, F) & = & \{[T \to T * F \bullet \quad]\} \\ q_{11} & = & \delta(q_8,)) & = & \{[F \to (\, E\,) \bullet \quad]\} \end{array}$$

The Canonical LR(1)-Automaton

For example:

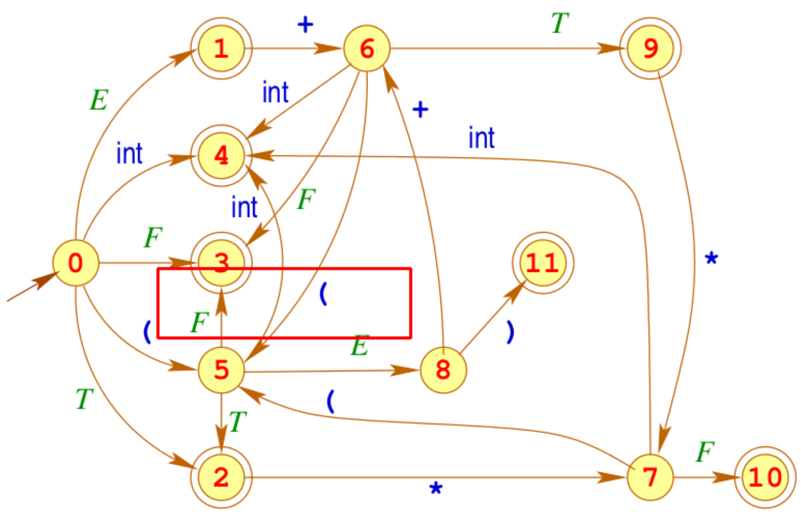
$$\begin{array}{lcl} E & \to & E + T \quad | \quad T \\ T & \to & T * F \quad | \quad F \\ F & \to & (\, E \,) \quad | \quad \text{int} \end{array}$$

$\text{First}_1(S') = \text{First}_1(E) = \text{First}_1(T) = \text{First}_1(F) = \text{name}, \text{int}, ($

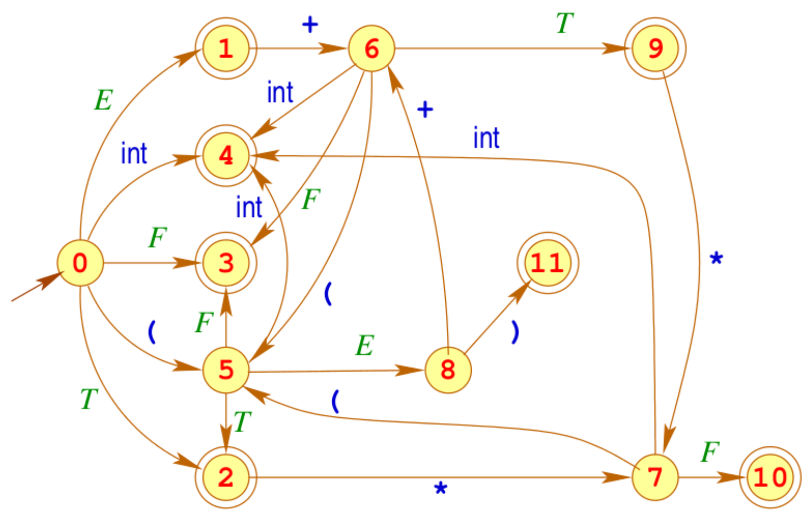
$$\begin{array}{lclcl} q_5' & = & \delta(q_5, () & = & \{[F \to (\, \bullet E\,),\ \{\,)\,, +, *\}], \\ & & & & [E \to \bullet E + T,\ \{\,)\,, +\}], \\ & & & & [E \to \bullet T,\ \{\,)\,, +\}], \\ & & & & [T \to \bullet T * F,\ \{\,)\,, +, *\}], \\ & & & & [T \to \bullet F,\ \{\,)\,, +, *\}], \\ & & & & [F \to \bullet (\, E\,),\ \{\,)\,, +, *\}], \\ & & & & [F \to \bullet \text{int},\ \{\,)\,, +, *\}]\} \\ q_6 & = & \delta(q_1, +) & = & \{[E \to E + \bullet T,\ \{\epsilon, +\}], \\ & & & & [T \to \bullet T * F,\ \{\epsilon, +, *\}], \\ & & & & [T \to \bullet F,\ \{\epsilon, +, *\}], \\ & & & & [F \to \bullet (\, E\,),\ \{\epsilon, +, *\}], \\ & & & & [F \to \bullet \text{int},\ \{\epsilon, +, *\}]\} \end{array}$$

$$\begin{array}{lclcl} q_7 & = & \delta(q_2, *) & = & \{[T \to T * \bullet F \quad], \\ & & & & [F \to \bullet (\, E\,) \quad], \\ & & & & [F \to \bullet \text{int} \quad]\} \\ q_8 & = & \delta(q_5, E) & = & \{[F \to (\, E \bullet\,) \quad]\} \\ & & & & [E \to E \bullet + T \quad]\} \\ q_9 & = & \delta(q_6, T) & = & \{[E \to E + T \bullet \quad], \\ & & & & [T \to T \bullet * F \quad]\} \\ q_{10} & = & \delta(q_7, F) & = & \{[T \to T * F \bullet \quad]\} \\ q_{11} & = & \delta(q_8,)) & = & \{[F \to (\, E\,) \bullet \quad]\} \end{array}$$

The Canonical LR(1)-Automaton



The Canonical LR(1)-Automaton



The Canonical LR(1)-Automaton

Discussion:

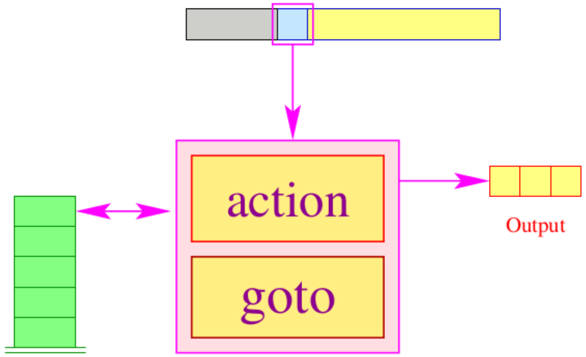
- In the example, the number of states was almost doubled
  ... and it can become even worse
- The conflicts in states q_1, q_2, q_9 are now resolved !
  e.g. we have for:

$$q_9 = \{[E \to E + T\bullet, \{\epsilon, +\}], \\ [T \to T \bullet * F, \{\epsilon, +, *\}]\}$$

with:

$$\{\epsilon, +\} \cap (\text{First}_1(*F) \odot \{\epsilon, +, *\}) = \{\epsilon, +\} \cap \{*\} = \emptyset$$

The LR(1)-Parser:



- The goto-table encodes the transitions:

$$\text{goto}[q, X] = \delta(q, X) \in Q$$

- The action-table describes for every state q and possible lookahead w the necessary action.

The LR(1)-Parser

The construction of the $LR(1)$-parser:

States: $Q \cup \{f\}$ (f fresh)
Start state: q_0
Final state: f
Transitions:

Shift: $(p, a, p\, q)$ if $q = \text{goto}[q, a]$, $s = \text{action}[p, w]$

Reduce: $(p\, q_1 \ldots q_{|\beta|}, \epsilon, p\, q)$ if $[A \to \beta\,\bullet] \in q_{|\beta|}$, $q = \text{goto}(p, A)$, $[A \to \beta\,\bullet] = \text{action}[q_{|\beta|}, w]$

Finish: $(q_0\, p, \epsilon, f)$ if $[S' \to S\bullet] \in p$

with $LR(G, 1) = (Q, T, \delta, q_0, F)$.

The LR(1)-Parser:

Possible actions are:

shift	//	Shift-operation
reduce $(A \to \gamma)$	//	Reduction with callback/output
error	//	Error

... for example:

$$E \to E + T^{\,0} \mid T^{\,1}$$
$$T \to T * F^{\,0} \mid F^{\,1}$$
$$F \to (\,E\,)^{\,0} \mid \text{int}^{\,1}$$

action	ϵ	int	(	)	+	*
q_1	$S', 0$					s
q_2	$E, 1$					s
q_2'				$E, 1$		s
q_3	$T, 1$				$T, 1$	$T, 1$
q_3'				$T, 1$	$T, 1$	$T, 1$
q_4	$F, 1$				$F, 1$	$F, 1$
q_4'				$F, 1$	$F, 1$	$F, 1$
q_9	$E, 0$				$E, 0$	s
q_9'				$E, 0$	$E, 0$	s
q_{10}	$T, 0$				$T, 0$	$T, 0$
q_{10}'				$T, 0$	$T, 0$	$T, 0$
q_{11}	$F, 0$				$F, 0$	$F, 0$
q_{11}'				$F, 0$	$F, 0$	$F, 0$

The LR(1)-Parser

The construction of the $LR(1)$-parser:

States: $Q \cup \{f\}$ (f fresh)
Start state: q_0
Final state: f
Transitions:

Shift: $(p, a, p\,q)$ if $q = \text{goto}[q, a]$, $s = \text{action}[p, w]$

Reduce: $(p\,q_1 \ldots q_{|\beta|}, \epsilon, p\,q)$ if $[A \to \beta \bullet] \in q_{|\beta|}$, $q = \text{goto}(p, A)$, $[A \to \beta \bullet] = \text{action}[q_{|\beta|}, w]$

Finish: $(q_0\,p, \epsilon, f)$ if $[S' \to S \bullet] \in p$

with $LR(G, 1) = (Q, T, \delta, q_0, F)$.

The Canonical LR(1)-Automaton

In general: We identify two conflicts:

Reduce-Reduce-Conflict:
$[A \to \gamma \bullet, x], \; [A' \to \gamma' \bullet, x] \in q$ with $A \neq A' \vee \gamma \neq \gamma'$

Shift-Reduce-Conflict:
$[A \to \gamma \bullet, x], \; [A' \to \alpha \bullet a\,\beta, y] \in q$
with $a \in T$ und $x \in \{a\} \odot_k \text{First}_k(\beta) \odot_k \{y\}$.

for a state $q \in Q$.

Such states are now called $LR(k)$-unsuited

Special LR(k)-Subclasses

Theorem:
A reduced contextfree grammar G is called $LR(k)$ iff the canonical $LR(k)$-automaton $LR(G, k)$ has no $LR(k)$-unsuited states.

Special LR(k)-Subclasses

Theorem:
A reduced contextfree grammar G is called $LR(k)$ iff the canonical $LR(k)$-automaton $LR(G, k)$ has no $LR(k)$-unsuited states.

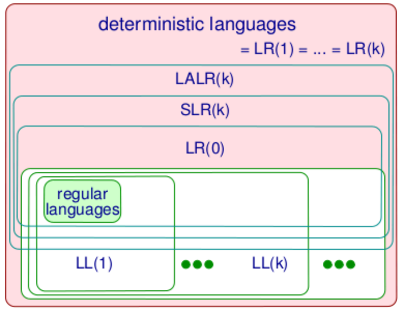
Discussion:

- Our example apparently is $LR(1)$
- In general, the canonical $LR(k)$-automaton has much more states then $LR(G) = LR(G, 0)$
- Therefore in practice, subclasses of $LR(k)$-grammars are often considered, which only use $LR(G)$ ...

Chapter 5:

Summary

Parsing Methods

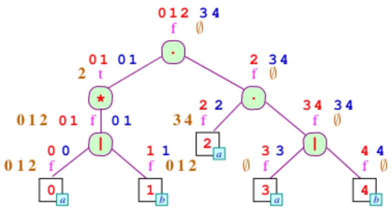


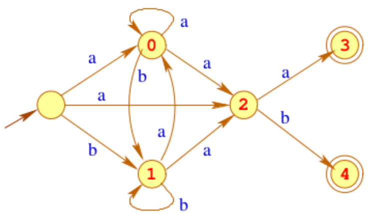
Discussion:

- All contextfree languages, that can be parsed with a deterministic pushdown automaton, can be characterized with an LR(1)-grammar.
- LR(0)-grammars describe all prefixfree deterministic contextfree languages
- The language-classes of LL(k)-grammars form a hierarchy within the deterministic contextfree languages.

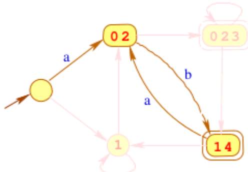
Lexical and Syntactical Analysis:

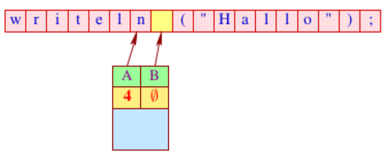
From Regular Expressions to Finite Automata





From Finite Automata to Scanners

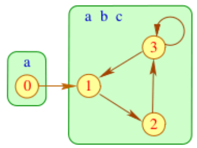


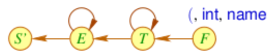


Lexical and Syntactical Analysis:

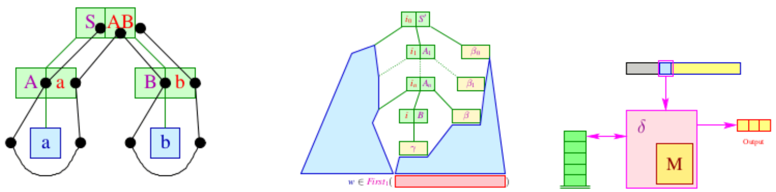
Computation of lookahead sets:

$$\begin{array}{lll} F_\epsilon(S') \supseteq F_\epsilon(E) & F_\epsilon(E) \supseteq F_\epsilon(E) \\ F_\epsilon(E) \supseteq F_\epsilon(T) & F_\epsilon(T) \supseteq F_\epsilon(T) \\ F_\epsilon(T) \supseteq F_\epsilon(F) & F_\epsilon(F) \supseteq \{(\,, \text{name}, \text{int}\} \end{array}$$



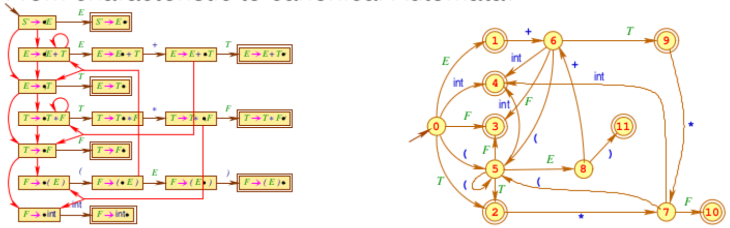


From Item-Pushdown Automata to LL(1)-Parsers:

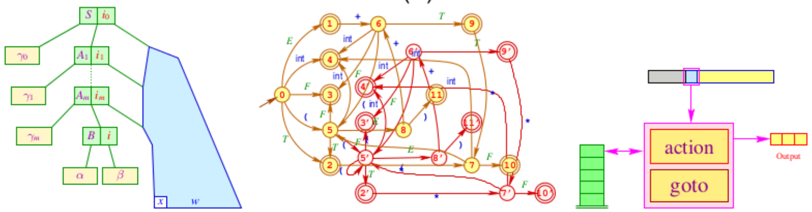


Lexical and Syntactical Analysis:

From characteristic to canonical Automata:

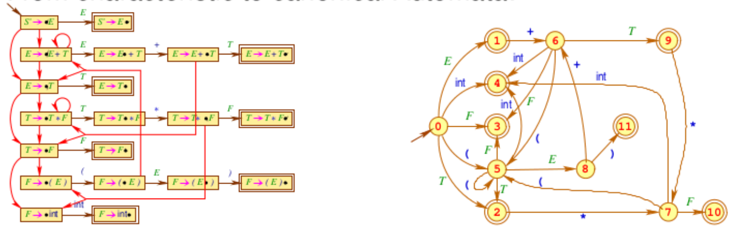


From Shift-Reduce-Parsers to LR(1)-Parsers:



Lexical and Syntactical Analysis:

From characteristic to canonical Automata:



From Shift-Reduce-Parsers to LR(1)-Parsers: