



AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

Parsery LR(1) – część 1

Teoria kompilacji

Dr inż. Janusz Majewski
Katedra Informatyki

Nazwa gramatyki: LR(k)

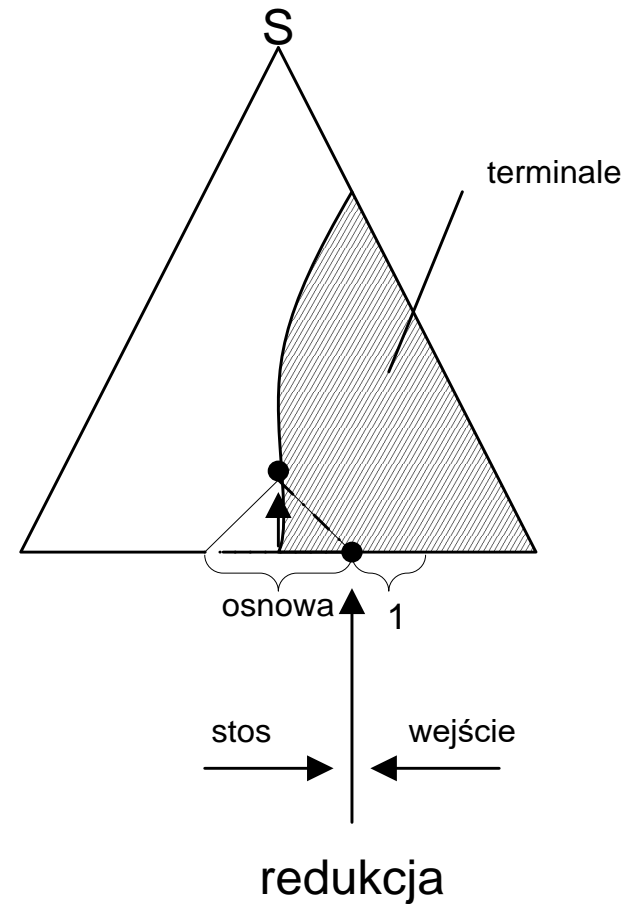
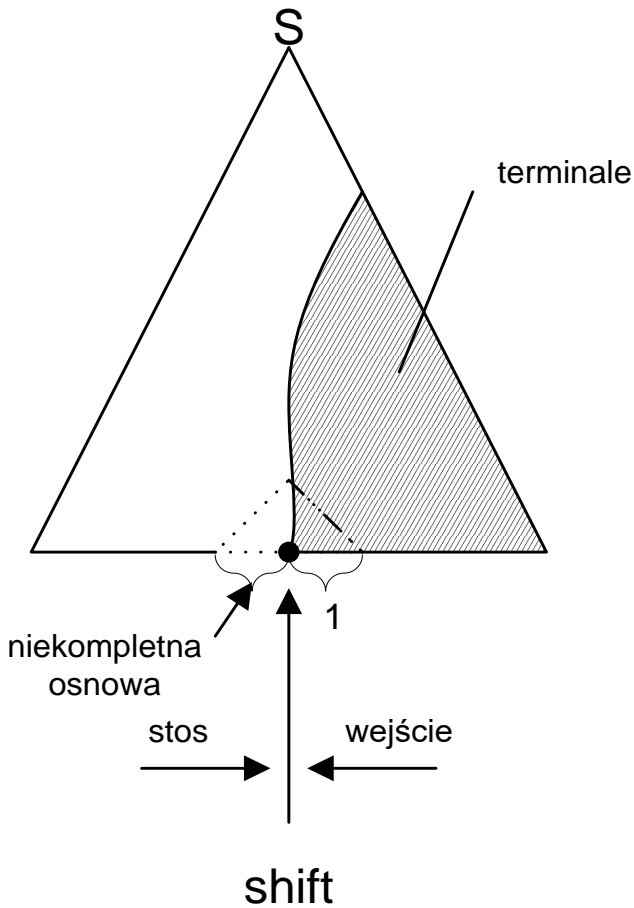
L R (k)

Przeglądanie
wejścia od
lewej strony
do prawej

Odtwarzanie
wyvodu
prawostronnego

Wystarcza znajomość
"k" następnych symboli
łańcucha wejściowego i
historii
dotychczasowych
redukcji, aby wyznaczyć
jednoznacznie osnowę i
dokonać jej redukcji

Odtwarzanie wywodu prawostronnego metodą bottom-up



Gramatyka uzupełniona

Dla gramatyki $G = \langle V, \Sigma, P, S \rangle \in \mathbf{G}_{BK}$

definiuje się gramatykę uzupełnioną G'

$$G' = \langle V \cup \{S'\}, \Sigma, P \cup \{S' \rightarrow S\}, S' \rangle$$

nowy symbol początkowy i dodatkowa
produkcja $S' \rightarrow S$ mają wyłącznie znaczenie
techniczne

Definicja gramatyki LR(k)

Gramatyka G jest gramatyką LR(k) dla $k \geq 0 \Leftrightarrow$ jeżeli z:

$$(i) \quad S' \xRightarrow[G'R]{*} \alpha Aw \xRightarrow[G'R]{} \alpha \beta w$$

$$(ii) \quad S' \xRightarrow[G'R]{*} \gamma Bx \xRightarrow[G'R]{} \alpha \beta y$$

$$(iii) \quad FIRST_k(w) = FIRST_k(y)$$

wynika, że

$$\alpha Ay = \gamma Bx$$

$$\text{tzn. } \alpha = \gamma; \quad A = B; \quad y = x$$

$$A, B \in V$$

$$w, x, y \in \Sigma^*$$

$$\alpha, \beta, \gamma \in (V \cup \Sigma)^*$$

\Rightarrow wyprowadzenie prawostronne
 $G'R$

w gramatyce uzupełnionej G'

Definicja gramatyki LR(k)

Gramatyka G jest gramatyką LR(k) dla $k \geq 0 \Leftrightarrow$ jeżeli z:

$$(i) S' \xRightarrow[G'R]{*} \alpha Aw \xRightarrow[G'R]{*} \alpha \beta w$$

$$(ii) S' \xRightarrow[G'R]{*} \gamma Bx \xRightarrow[G'R]{*} \alpha \beta y$$

$$(iii) FIRST_k(w) = FIRST_k(y)$$

wynika, że

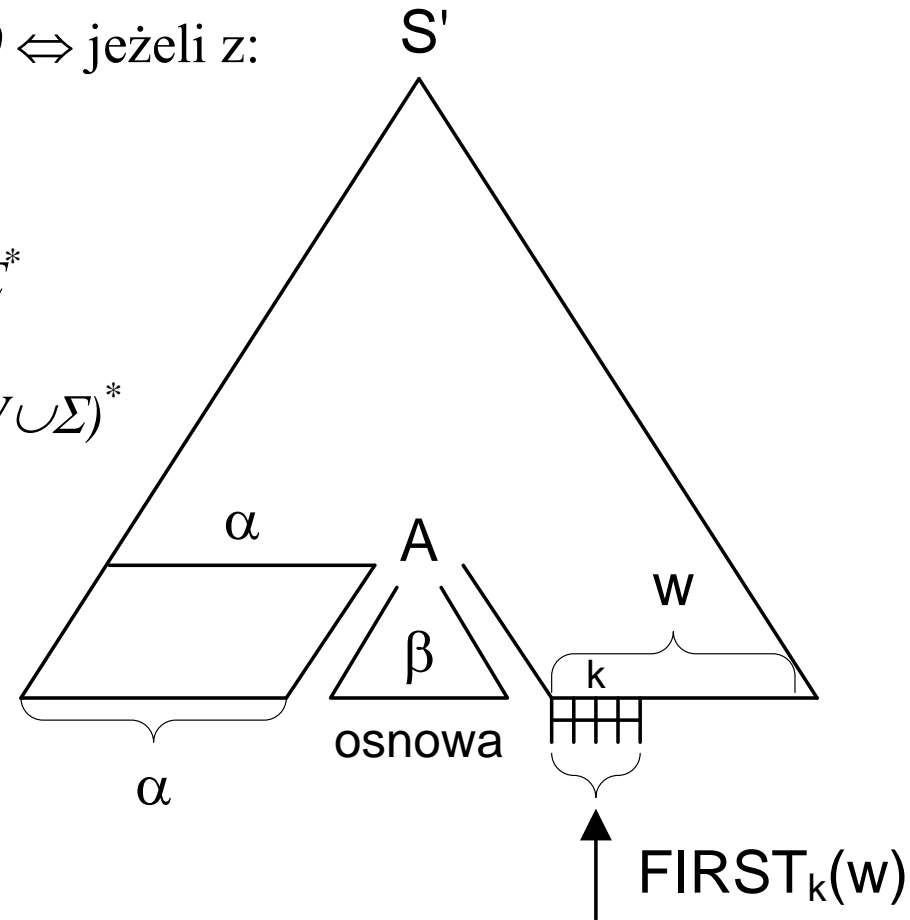
$$\alpha Ay = \gamma Bx$$

$$\text{tzn. } \alpha = \gamma; A = B; y = x$$

$$A, B \in V$$

$$w, x, y \in \Sigma^*$$

$$\alpha, \beta, \gamma \in (V \cup \Sigma)^*$$



Przykład: $S \rightarrow Sa \mid a$

Gramatyka uzupełniona:

$$S' \rightarrow S$$

$$S \rightarrow Sa$$

$$S \rightarrow a$$

Gramatyka ta nie jest gramatyką LR(0), bo z tego, że

$$(i) S' \xRightarrow{*} S' \Rightarrow S \quad (ii) S' \xRightarrow{*} S \Rightarrow Sa$$

$$S' \quad \alpha A w \quad \alpha \beta w \quad S' \quad \gamma B x \quad \gamma \beta y$$

$$(iii) FIRST_0(\varepsilon) = FIRST_0(a) = \{\varepsilon\}$$

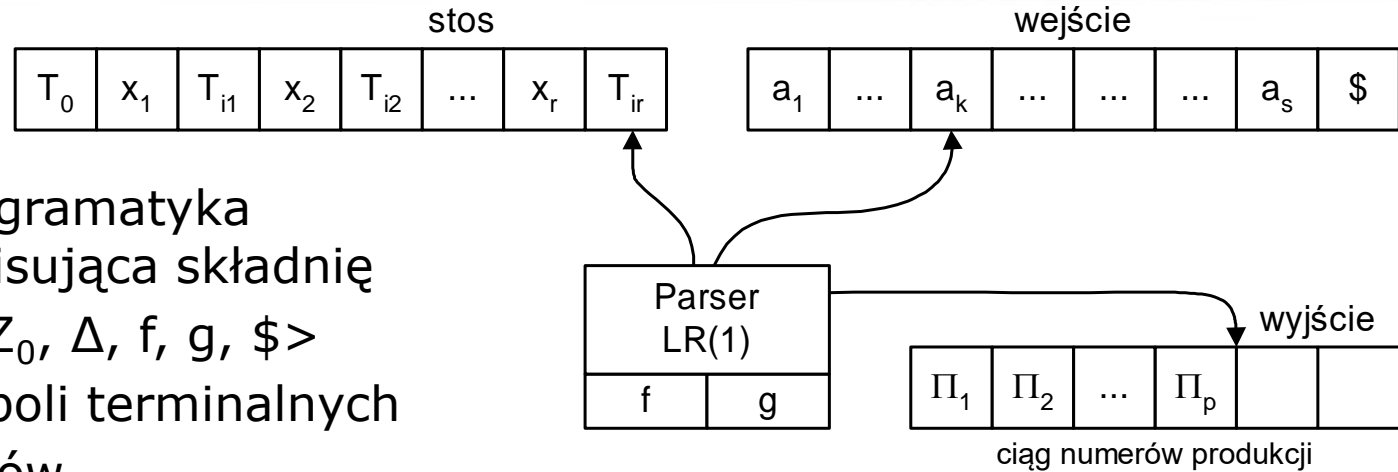
$$FIRST_0(w) \quad FIRST_0(y)$$

nie wynika, że $S'a = S$

$$\alpha A y \quad \gamma B x$$

Nie można stwierdzić, czy S jest osnową nie znając następnego symbolu (którym może być $\$$ – wtedy S będzie osnową, lub a – wtedy osnową nie będzie S lecz Sa)

Parser LR(1)



$G = \langle V, \Sigma, P, S \rangle$ – gramatyka
uzupełniona opisująca składnię

$A = \langle \Sigma, \mathfrak{S}, T_0, \Gamma, Z_0, \Delta, f, g, \$ \rangle$

Σ – zbiór symboli terminalnych

\mathfrak{S} – zbiór stanów

$T_0 \in \mathfrak{S}$ – stan początkowy

Γ – zbiór symboli stosowych

$$\Gamma = V \cup \Sigma \cup \mathfrak{S}$$

$Z_0 = T_0$ – symbol początkowy stosu oraz stan początkowy

Δ – alfabet wyjściowy: zbiór numerów produkcji

f – tablica działania parsera

g – tablica przejść parsera

$\$$ – ogranicznik końca słowa wejściowego

Automat parsera LR(1)

Analizator (parser) LR(1) jest zdeterminowanym automatem ze stosem, którego sterowanie określają dwie funkcje f i g :

f – funkcja działania

$$g: \mathcal{S} \times (\Sigma \cup \{\$\}) \alpha \{ \underline{shift} \ j, \underline{red} \ i, \underline{err}, \underline{acc} \}$$

↑
↑
 numer stanu numer produkcji

g – funkcja przejścia

$$g: \mathcal{S} \times V \alpha \mathcal{S} \cup \{ \underline{err} \}$$

gdzie:

\mathcal{S} - zbiór stanów

$$\mathcal{S} = \{ T_j : j = 0, 1, \dots, n \}$$

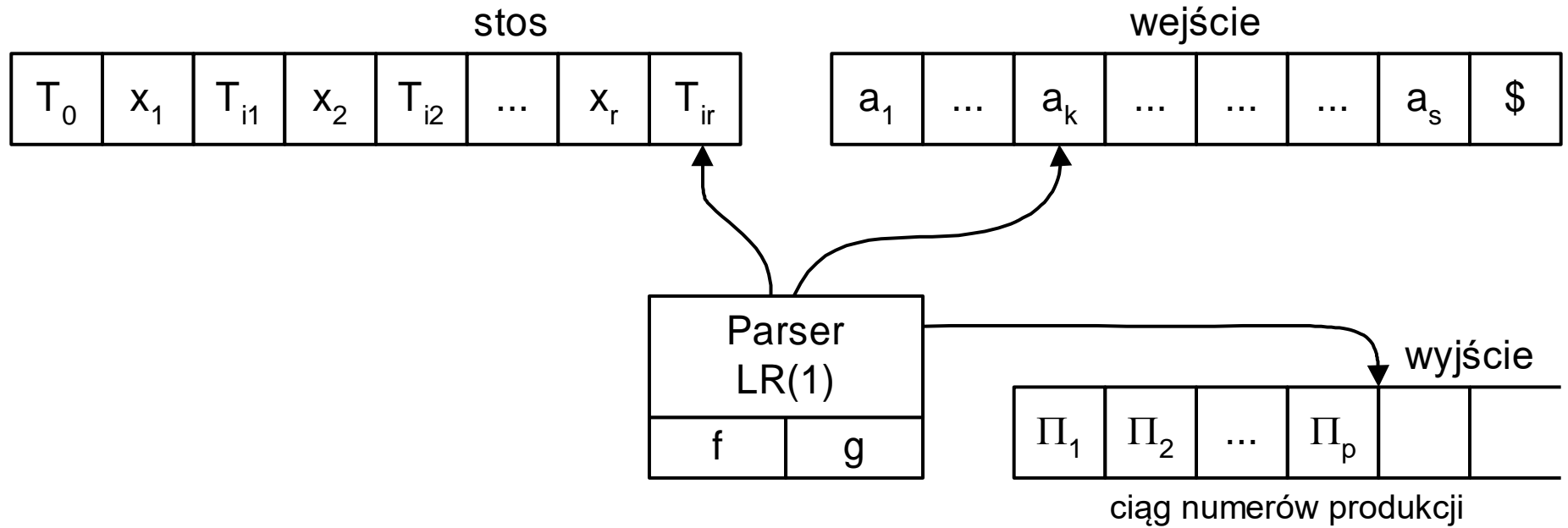
$i = 1, 2, \dots, m$

$$m = \#P$$

m – liczba produkcji gramatyki G

T_0 – stan początkowy, a zarazem początkowy symbol stosu

Automat parsera LR(1)



Konfiguracja parsera:

$$(T_0 x_1 T_{i_1} \dots x_r T_{i_r}, a_k \dots a_s \$, \pi_1 \dots \pi_p)$$

przy czym : x_i są symbolami gramatyki (nieterminalami lub terminalami)

Konfiguracja początkowa : $(T_0, a_1 a_2 \dots a_s \$, \varepsilon)$

Działanie parsera LR(1)

Niech parser znajduje się w konfiguracji:

$$(T_0 x_1 T_{i_1} \dots x_r T_{i_r}, a_k a_{k+1} \dots a_s \$, \pi)$$

(i) Jeśli $f(T_{i_r}, a_k) = \underline{red}$ i

oraz $A \rightarrow \gamma$ jest produkcją o numerze i

$$|\gamma| = d; d \leq r$$

to:

$$(T_0 x_1 T_{i_1} \dots x_{r-d} T_{i_{r-d}} \underbrace{\dots x_r T_{i_r}}_{\text{zdejmie się } 2d \text{ symboli}}, a_k a_{k+1} \dots a_s \$, \pi) \Rightarrow$$

zdejmie się $2d$ symboli

$$\Rightarrow (T_0 x_1 T_{i_1} \dots x_{r-d} T_{i_{r-d}} A T_j, a_k a_{k+1} \dots a_s \$, \pi)$$

$$\text{gdzie } T_j = g(T_{i_{r-d}}, A)$$

Działanie parsera LR(1)

Niech parser znajduje się w konfiguracji:

$$(T_0 x_1 T_{i_1} \dots x_r T_{i_r}, a_k a_{k+1} \dots a_s \$, \pi)$$

(ii) Jeśli $f(T_{i_r}, a_k) = \underline{shift\ j}$

to:

$$(T_0 x_1 T_{i_1} \dots x_r T_{i_r}, a_k a_{k+1} \dots a_s \$, \pi) \Rightarrow$$

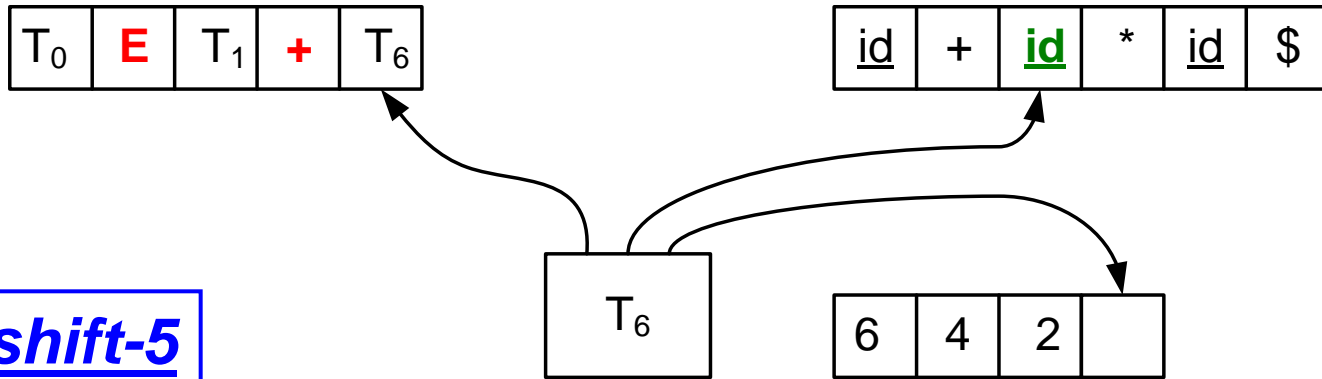
$$\Rightarrow (T_0 x_1 T_{i_1} \dots x_r T_{i_r} r a_k T_j, a_{k+1} \dots a_s \$, \pi)$$

(iii) Jeśli $f(T_{i_r}, a_k) = \underline{acc}$ — akceptacja,

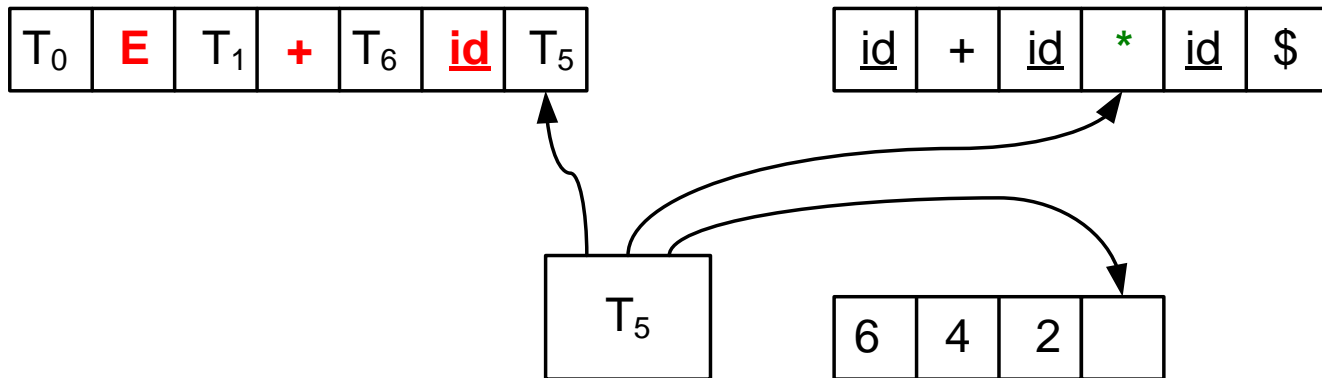
parsing zakończony poprawnie

(iv) Jeśli $f(T_{i_r}, a_k) = \underline{err}$ — błąd syntaktyczny

Krok „shift”



$f(T_6, \underline{id}) = \underline{shift-5}$

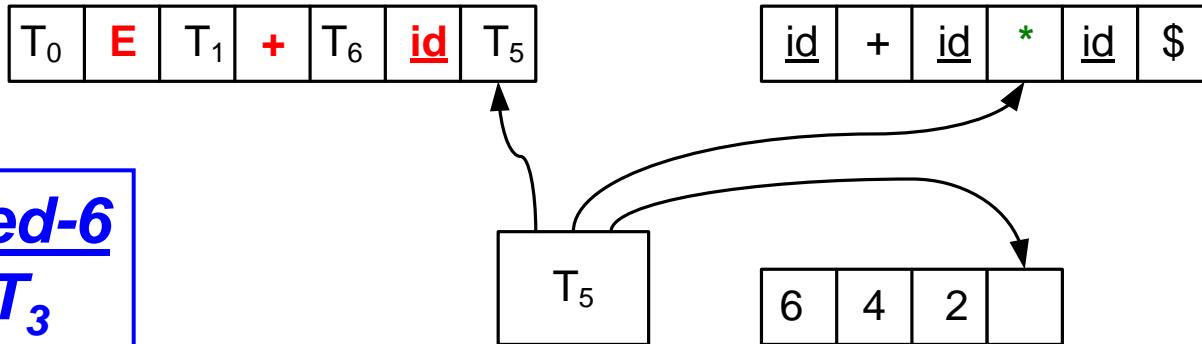


Przykład – gramatyka jednoznaczna

- (0) $E' \rightarrow E$
 (1) $E \rightarrow E+T$
 (2) $E \rightarrow T$
 (3) $T \rightarrow T * F$
 (4) $T \rightarrow F$
 (5) $F \rightarrow (E)$
 (6) $F \rightarrow id$

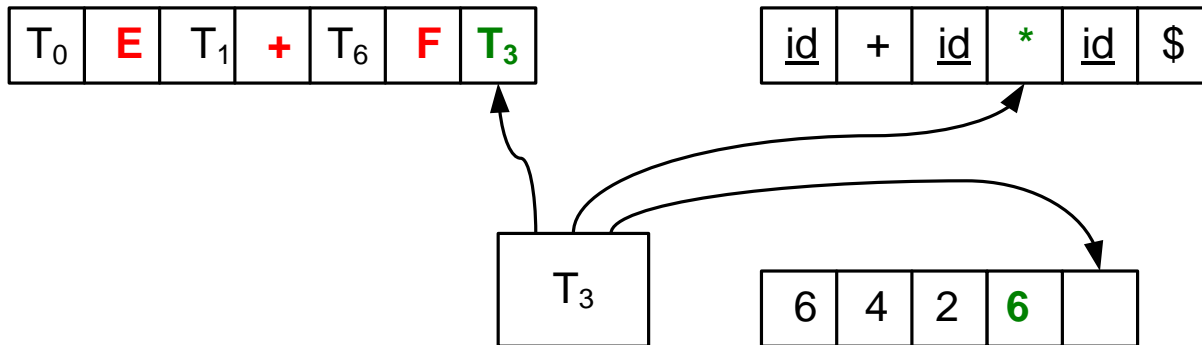
stan	f						g		
	\$	+	*	()	id	E	T	F
T ₀				<u>shift4</u>		<u>shift5</u>	T ₁	T ₂	T ₃
T ₁	<u>acc</u>	<u>shift 6</u>							
T ₂	<u>red-2</u>	<u>red-2</u>	<u>shift</u>		<u>red-2</u>				
T ₃	<u>red-4</u>	<u>red-4</u>	<u>red-4</u>		<u>red-4</u>				
T ₄				<u>shift4</u>		<u>shift5</u>	T ₈	T ₂	T ₃
T ₅	<u>red-6</u>	<u>red-6</u>	<u>red-6</u>		<u>red-6</u>				
T ₆				<u>shift4</u>		<u>shift5</u>		T ₉	T ₃
T ₇				<u>shift4</u>		<u>shift5</u>			T ₁₀
T ₈		<u>shift6</u>			<u>shift11</u>				
T ₉	<u>red-1</u>	<u>red-1</u>	<u>shift7</u>		<u>red-1</u>				
T ₁₀	<u>red-3</u>	<u>red-3</u>	<u>red-3</u>		<u>red-3</u>				
T ₁₁	<u>red-5</u>	<u>red-5</u>	<u>red-5</u>		<u>red-5</u>				

Krok „redukcji”



$f(T_5, *) = \underline{red-6}$
 $g(T_6, F) = T_3$

(6) **F** → id



Przykład – gramatyka jednoznaczna

- (0) $E' \rightarrow E$
 (1) $E \rightarrow E + T$
 (2) $E \rightarrow T$
 (3) $T \rightarrow T * F$
 (4) $T \rightarrow F$
 (5) $F \rightarrow (E)$
 (6) $F \rightarrow id$

stan	f						g		
	\$	+	*	()	id	E	T	F
T ₀				<u>shift4</u>		<u>shift5</u>	T ₁	T ₂	T ₃
T ₁	<u>acc</u>	<u>shift 6</u>							
T ₂	<u>red-2</u>	<u>red-2</u>	<u>shift</u>		<u>red-2</u>				
T ₃	<u>red-4</u>	<u>red-4</u>	<u>red-4</u>		<u>red-4</u>				
T ₄				<u>shift4</u>		<u>shift5</u>	T ₈	T ₂	T ₃
T ₅	<u>red-6</u>	<u>red-6</u>	<u>red-6</u>		<u>red-6</u>				
T ₆				<u>shift4</u>		<u>shift5</u>		T ₉	T₃
T ₇				<u>shift4</u>		<u>shift5</u>			T ₁₀
T ₈		<u>shift6</u>			<u>shift11</u>				
T ₉	<u>red-1</u>	<u>red-1</u>	<u>shift7</u>		<u>red-1</u>				
T ₁₀	<u>red-3</u>	<u>red-3</u>	<u>red-3</u>		<u>red-3</u>				
T ₁₁	<u>red-5</u>	<u>red-5</u>	<u>red-5</u>		<u>red-5</u>				

Przykład

Gramatyka uzupełniona:

(0) $S' \rightarrow S$

(1) $S \rightarrow SaSb$

(2) $S \rightarrow \varepsilon$

W tablicy parsera
nie umieszczamy
pozycji err

Stany	f			g
	a	b	$\$$	S
T_0	<u>red 2</u>	<u>err</u>	<u>red 2</u>	T_1
T_1	<u>shift 2</u>	<u>err</u>	<u>acc</u>	<u>err</u>
T_2	<u>red 2</u>	<u>red 2</u>	<u>err</u>	T_3
T_3	<u>shift 4</u>	<u>shift 5</u>	<u>err</u>	<u>err</u>
T_4	<u>red 2</u>	<u>red 2</u>	<u>err</u>	T_6
T_5	<u>red 1</u>	<u>err</u>	<u>red 1</u>	<u>err</u>
T_6	<u>shift 4</u>	<u>shift 7</u>	<u>err</u>	<u>err</u>
T_7	<u>red 1</u>	<u>red 1</u>	<u>err</u>	<u>err</u>

Przykład

Gramatyka uzupełniona:

(0) $S' \rightarrow S$

(1) $S \rightarrow SaSb$

(2) $S \rightarrow \varepsilon$

Stany	f			g
	a	b	$\$$	S
T_0	<u>red 2</u>		<u>red 2</u>	T_1
T_1	<u>shift 2</u>		<u>acc</u>	
T_2	<u>red 2</u>	<u>red 2</u>		T_3
T_3	<u>shift 4</u>	<u>shift 5</u>		
T_4	<u>red 2</u>	<u>red 2</u>		T_6
T_5	<u>red 1</u>		<u>red 1</u>	
T_6	<u>shift 4</u>	<u>shift 7</u>		
T_7	<u>red 1</u>	<u>red 1</u>		

Przykład parsingu

$(T_0, aabb\$, \varepsilon)$	ϕ	$(T_0ST_1, aabb\$, 2)$	(red-2)
	ϕ	$(T_0ST_1aT_2, abb\$, 2)$	(shift)
	ϕ	$(T_0ST_1aT_2ST_3, abb\$, 22)$	(red-2)
	ϕ	$(T_0ST_1aT_2ST_3aT_4, bb\$, 22)$	(shift)
	ϕ	$(T_0ST_1aT_2ST_3aT_4ST_6, bb\$, 222)$	(red-2)
	ϕ	$(T_0ST_1aT_2\boxed{ST_3aT_4ST_6bT_7}, b\$, 222)$	(shift)
	ϕ	$(T_0ST_1aT_2ST_3, b\$, 2221)$	(red-1)
	ϕ	$(T_0\boxed{ST_1aT_2ST_3bT_5}, \$, 2221)$	(shift)
	ϕ	$(T_0ST_1, \$, 22211)$	(red-1)
	(acc)		

Przykład c.d.

Sprawdzenie (zgodnie z wywodem prawostronnym dokonywanym przez parser):

$$aabb \xleftarrow[R]{2} Saabb \xleftarrow[R]{2} SaSabb \xleftarrow[R]{2} SaSaSbb \xleftarrow[R]{1} SaSb \xleftarrow[R]{1} S \quad \text{o.k.}$$

Formy zdaniowe z zaznaczonymi osnowami

