AKADEMIA GÓRNICZO-HUTNICZA
IM. STANISŁAWA STASZICA W KRAKOWIE

Parsery LALR(1)

Teoria kompilacji

Dr inż. Janusz Majewski
Modyfikacje Marcin Kuta

Gramatyki LALR(k) (Look Ahead LR(k))

Duża klasa gramatyk mających praktyczne zastosowanie nie należy do SLR(1), ale dla tych gramatyk daje się skonstruować parser o rozmiarze tablicy sterującej (funkcje ”f” i ”g”) identycznym z SLR(1), tyle że przy pomocy trochę bardziej skomplikowanego algorytmu. Klasa ta nazywa się LALR(1) i z praktycznego punktu widzenia jest najważniejszą podklasą LR(1), gdyż:

1) Jest dostatecznie szeroka, żeby objąć znaczną większość języków programowania.

2) Rozmiar tablicy sterującej parsera jest jeszcze do przyjęcia (w odróżnieniu od kanonicznego LR(1)).

Rdzeń zbioru LR(1)-sytuacji

CORE – rdzeń zbioru LR(1) – sytuacji

$\mathcal{A}$ - zbiór LR(1) – sytuacji

$CORE(\mathcal{A}) = \{ [A \rightarrow \alpha \bullet \beta] \mid [A \rightarrow \alpha \bullet \beta, u] \in \mathcal{A} \}$

Twierdzenie

$\mathcal{J}_0$ – kanoniczny system zbiorów LR(0) – sytuacji w G

$\mathcal{J}_1$ – kanoniczny system zbiorów LR(1) – sytuacji w G

$$\mathcal{J}_0 = \{ CORE(\mathcal{A}) : \mathcal{A} \in \mathcal{J}_1 \}$$

CORE(GOTO(…))

Twierdzenie

$$CORE(GOTO_1(\mathcal{A}, x)) = GOTO_0(CORE(\mathcal{A}), x)$$

$$x \in (V \cup \Sigma)$$

gdzie:

GOTO$_1$ – funkcja *GOTO* dla zbioru LR(1)-sytuacji

GOTO$_0$ – funkcja *GOTO* dla zbioru LR(0)-sytuacji

$\mathcal{e}$ – relacja równości rdzeni

$$\mathcal{e} \subset \mathcal{J}_1 \times \mathcal{J}_1 \ : \mathcal{A}_1 \, \mathcal{e} \, \mathcal{A}_2 \overset{!}{\Leftrightarrow} CORE(\mathcal{A}_1) = CORE(\mathcal{A}_2)$$

Gramatyka LALR(1)

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

$\mathcal{J}_1$ – kanoniczny system zbiorów LR(1) – sytuacji dla G

$\mathcal{J} = \{ \mathcal{B}([\mathcal{A}]_e) : \mathcal{A} \in \mathcal{J}_1 \};$

gdzie:

$$\mathcal{B}([\mathcal{A}]_e) = \bigcup_{C \in [\mathcal{A}]_e} \quad (C \in \mathcal{J}_1)$$

G – gramatyka LALR(1) $\Leftrightarrow (\forall \mathcal{B} \in \mathcal{J})$ ($\mathcal{B}$ - zgodny)

Przykład (łączenie zbiorów LR(1)-sytuacji)

$\mathcal{J}_1 = \{ \mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4, \mathcal{A}_5, \mathcal{A}_6 \}$

$\mathcal{e} = \{ (\mathcal{A}_1, \mathcal{A}_1), (\mathcal{A}_2, \mathcal{A}_2), (\mathcal{A}_3, \mathcal{A}_3), (\mathcal{A}_4, \mathcal{A}_4), (\mathcal{A}_2, \mathcal{A}_3),$
$(\mathcal{A}_3, \mathcal{A}_2), (\mathcal{A}_2, \mathcal{A}_4), (\mathcal{A}_4, \mathcal{A}_2), (\mathcal{A}_3, \mathcal{A}_4), (\mathcal{A}_4, \mathcal{A}_3),$
$(\mathcal{A}_5, \mathcal{A}_5), (\mathcal{A}_6, \mathcal{A}_5), (\mathcal{A}_5, \mathcal{A}_6), (\mathcal{A}_6, \mathcal{A}_6) \}$

$\mathcal{B}_1 = \mathcal{A}_1$

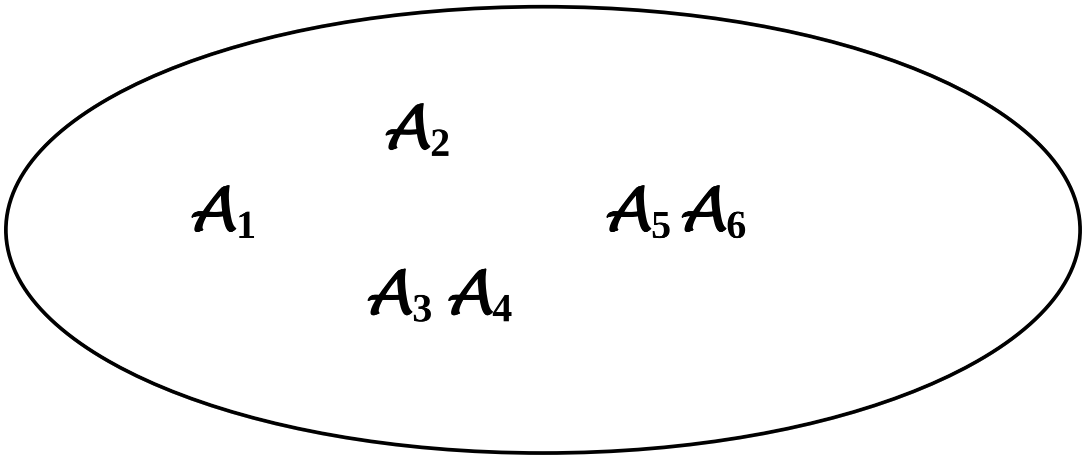
$\mathcal{B}_2 = \mathcal{A}_2 \cup \mathcal{A}_3 \cup \mathcal{A}_4$

$\mathcal{B}_3 = \mathcal{A}_5 \cup \mathcal{A}_6$

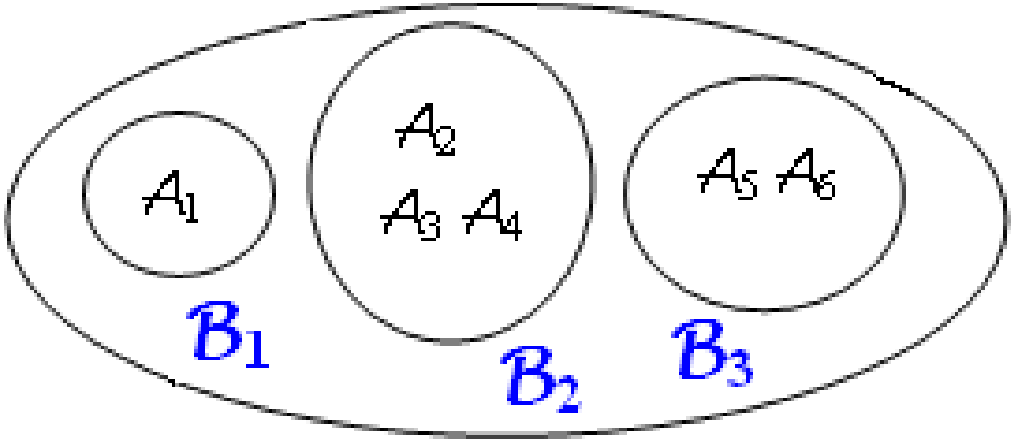
$\mathcal{J} = \{\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3\}$

Przykład c.d.

$\mathcal{J}_1 =$



$\mathcal{J} =$



Konstrukcja tablicy parsera LALR(1)

WEJŚCIE: $G = \langle V, T, P, S \rangle \in \mathcal{G}_{BK}$
WYJŚCIE: Tablica LALR(1)
ALGORYTM:

1. Wyznaczamy $\mathcal{J}_1$ – kanoniczny system zbiorów LR(1)-sytuacji dopuszczalnych
2. Konstruujemy zbiór klas abstrakcji dla relacji $\mathcal{e}$:

   $\mathcal{J} := \{ \mathcal{B}([\mathcal{A}]_{\mathcal{e}}) : \mathcal{A} \in \mathcal{J}_1 \};$

   /* $\mathcal{J} = \{ \mathcal{B}_0, \mathcal{B}_1, \ldots , \mathcal{B}_n \}$

   gdzie $\mathcal{B}_i = \mathcal{A}_{(i1)} \cup \mathcal{A}_{(i2)} \cup \ldots \cup \mathcal{A}_{(i\ mi)}$

   $i = 0, 1, \ldots , n$

   $[\mathcal{A}_{(i1)}]_{\mathcal{e}} = [\mathcal{A}_{(i2)}]_{\mathcal{e}} = \ldots = [\mathcal{A}_{(i\ mi)}]_{\mathcal{e}}$ */
3. Określamy funkcje f i g dla $\mathcal{J} = \{ \mathcal{B}_0, \mathcal{B}_1, \ldots , \mathcal{B}_n \}$ identycznie, jak to miało miejsce w przypadku kanonicznego LR(1);

Przykład

$G' = \langle V' = \{ S', S, L, R \}, T = \{ =, *, \underline{id} \}, P', S' \rangle$

$P' = \{$

(0) $S' \rightarrow S$

(1) $S \rightarrow L = R$

(2) $S \rightarrow R$

(3) $L \rightarrow *R$

(4) $L \rightarrow \underline{id}$

(5) $R \rightarrow L$

$\}$

$FIRST_1(S) = \{*, \underline{id}\}$

$FIRST_1(L) = \{*, \underline{id}\}$

$FIRST_1(R) = \{*, \underline{id}\}$

$FOLLOW_1(S) = \{\$\}$

$FOLLOW_1(L) = \{\$, =\}$

$FOLLOW_1(R) = \{\$, =\}$

Przykład

Sprawdzenie czy poniższa gramatyka G' jest SLR(1)?

Próbujemy skonstruować $\mathcal{J}_0$ – kanoniczny system LR(0) – sytuacji

$$\mathcal{A}_0 = \{ \quad [S' \to \bullet S],\ [S \to \bullet L = R],\ [S \to \bullet R],\ [L \to \bullet *R],\ [L \to \bullet \underline{id}],\ [R \to \bullet L] \}$$

$$\mathcal{A}_1 = GOTO(\mathcal{A}_0, S) = \{ [S' \to S\bullet] \}$$

$$\mathcal{A}_2 = GOTO(\mathcal{A}_0, L) = \{ [S \to L\bullet = R],\ [R \to L\bullet] \}$$

1) $S' \to S$
2) $S \to L = R$
3) $S \to R$
4) $L \to *R$
5) $L \to \underline{id}$
6) $R \to L$

$FOLLOW_1(S) = \{\$\}$
$FOLLOW_1(L) = \{\$, =\}$
$FOLLOW_1(R) = \{\$, =\}$

Przykład

$$\mathcal{A}_2 = GOTO(\mathcal{A}_0, L) = \{\ [\ S \to L\bullet = R]\ ,$$
$$[\ R \to L\bullet]$$
$$\}$$

Ponieważ:

$$FOLLOW_1(R) = \{\$, =\}$$

Więc:

$$\left.\begin{array}{l} f(T_2, =) = \underline{red\text{–}6} \\ f(T_2, =) = \underline{shift} \end{array}\right\} \quad \underline{\text{konflikt!}}$$

gramatyka nie jest SLR(1)!

Przykład

Konstruujemy $\mathcal{J}_1$ – kanoniczny system zbiorów LR(1)–sytuacji.

1) $S' \to S$
2) $S \to L = R$
3) $S \to R$
4) $L \to {*R}$
5) $L \to \underline{id}$
6) $R \to L$

$$\mathcal{A}_0 = \{ \; [\, S' \to \bullet S,\ \$\,],$$
$$[\, S \to \bullet L{=}R,\ \$\,],$$
$$[\, S \to \bullet R,\ \$\,],$$
$$[\, L \to \bullet {*R},\ \$|{=}\,],$$
$$[\, L \to \bullet \underline{id},\ \$|{=}\,],$$
$$[\, R \to \bullet L,\ \$\,]\}$$

$$\mathcal{A}_1 = \{ [S' \to S\bullet,\ \$] \} \qquad \mathcal{A}_1 = GOTO(\mathcal{A}_0, S)$$

$$\mathcal{A}_2 = \{ [S \to L\bullet{=}R,\ \$] \qquad \mathcal{A}_2 = GOTO(\mathcal{A}_0, L)$$
$$[R \to L\bullet,\ \$] \}$$

Przykład

$\mathcal{A}_3 = \{\ [S \to R\bullet, \$]\ \}$ $\qquad \mathcal{A}_3 = GOTO(\mathcal{A}_0, R)$

$\mathcal{A}_4 = \{\ [L \to *\bullet R, \$|=]\ ,$ $\qquad \mathcal{A}_4 = GOTO(\mathcal{A}_0, *)$

$[R \to \bullet L, \$|=],$

$[L \to \bullet *R, \$|=],$

$[L \to \bullet \underline{id}, \$|=]\}$

$\mathcal{A}_5 = \{\ [L \to \underline{id}\bullet, \$|=]\ \}$ $\qquad \mathcal{A}_5 = GOTO(\mathcal{A}_0, \underline{id})$

$\mathcal{A}_6 = \{\ [S \to L{=}\bullet R, \$],$ $\qquad \mathcal{A}_6 = GOTO(\mathcal{A}_2, =)$

$[R \to \bullet L, \$],$

$[L \to \bullet *R, \$],$

$[L \to \bullet \underline{id}, \$]\ \}$

1) $S' \to S$
2) $S \to L = R$
3) $S \to R$
4) $L \to *R$
5) $L \to \underline{id}$
6) $R \to L$

Przykład

$\mathcal{A}_7 = \{ [L \rightarrow *R\bullet, \$|=] \}$

$\mathcal{A}_8 = \{ [R \rightarrow L\bullet, \$|=] \}$

$\mathcal{A}_9 = \{[S \rightarrow L=R\bullet, \$]\}$

$\mathcal{A}_{10} = \{[L \rightarrow *\bullet R, \$],$
$[R \rightarrow \bullet L, \$],$
$[L \rightarrow \bullet *R, \$],$
$[L \rightarrow \bullet \underline{id}, \$]\}$

$\mathcal{A}_7 = GOTO(\mathcal{A}_4, R)$

$\mathcal{A}_8 = GOTO(\mathcal{A}_4, L)$

$\mathcal{A}_4 = GOTO(\mathcal{A}_4, *)$

$\mathcal{A}_5 = GOTO(\mathcal{A}_4, \underline{id})$

$\mathcal{A}_9 = GOTO(\mathcal{A}_6, R)$

$\mathcal{A}_{10} = GOTO(\mathcal{A}_6, *)$

1) $S' \rightarrow S$
2) $S \rightarrow L = R$
3) $S \rightarrow R$
4) $L \rightarrow *R$
5) $L \rightarrow \underline{id}$
6) $R \rightarrow L$

Przykład

$A_{11}=\{[L\rightarrow \underline{id}\bullet, \$]\}$

$A_{12}=\{[R\rightarrow L\bullet, \$]\}$

$A_{13}=\{[L\rightarrow *R\bullet, \$]\}$

$A_{11}=GOTO(A_6, \underline{id})$

$A_{12}=GOTO(A_6,L)$

$A_{13}=GOTO(A_{10},R)$

$A_{12}=GOTO(A_{10},L)$

$A_{10}=GOTO(A_{10},*)$

$A_{11}=GOTO(A_{10},\underline{id})$

1) $S' \rightarrow S$
2) $S \rightarrow L = R$
3) $S \rightarrow R$
4) $L \rightarrow *R$
5) $L \rightarrow \underline{id}$
6) $R \rightarrow L$

Przykład

$\mathcal{B}_0 = \mathcal{A}_0$	
$\mathcal{B}_1 = \mathcal{A}_1$	$\mathcal{B}_1 = \mathrm{GOTO}(\mathcal{B}_0, S)$
$\mathcal{B}_2 = \mathcal{A}_2$	$\mathcal{B}_2 = \mathrm{GOTO}(\mathcal{B}_0, L)$
$\mathcal{B}_3 = \mathcal{A}_3$	$\mathcal{B}_3 = \mathrm{GOTO}(\mathcal{B}_0, R)$
$\underline{\mathcal{B}_4 = \mathcal{A}_4 \cup \mathcal{A}_{10}}$	$\mathcal{B}_4 = \mathrm{GOTO}(\mathcal{B}_0, *)$
$\underline{\mathcal{B}_5 = \mathcal{A}_5 \cup \mathcal{A}_{11}}$	$\mathcal{B}_5 = \mathrm{GOTO}(\mathcal{B}_0, \underline{id})$
$\mathcal{B}_6 = \mathcal{A}_6$	$\mathcal{B}_6 = \mathrm{GOTO}(\mathcal{B}_2, =)$
$\underline{\mathcal{B}_7 = \mathcal{A}_7 \cup \mathcal{A}_{13}}$	$\mathcal{B}_7 = \mathrm{GOTO}(\mathcal{B}_4, R)$
$\underline{\mathcal{B}_8 = \mathcal{A}_8 \cup \mathcal{A}_{12}}$	$\mathcal{B}_8 = \mathrm{GOTO}(\mathcal{B}_4, L)$
	$\mathcal{B}_4 = \mathrm{GOTO}(\mathcal{B}_4, *)$
	$\mathcal{B}_5 = \mathrm{GOTO}(\mathcal{B}_4, \underline{id})$
$\mathcal{B}_9 = \mathcal{A}_9$	$\mathcal{B}_9 = \mathrm{GOTO}(\mathcal{B}_6, R)$
	$\mathcal{B}_4 = \mathrm{GOTO}(\mathcal{B}_6, *)$
	$\mathcal{B}_5 = \mathrm{GOTO}(\mathcal{B}_6, \underline{id})$
	$\mathcal{B}_8 = \mathrm{GOTO}(\mathcal{B}_6, L)$

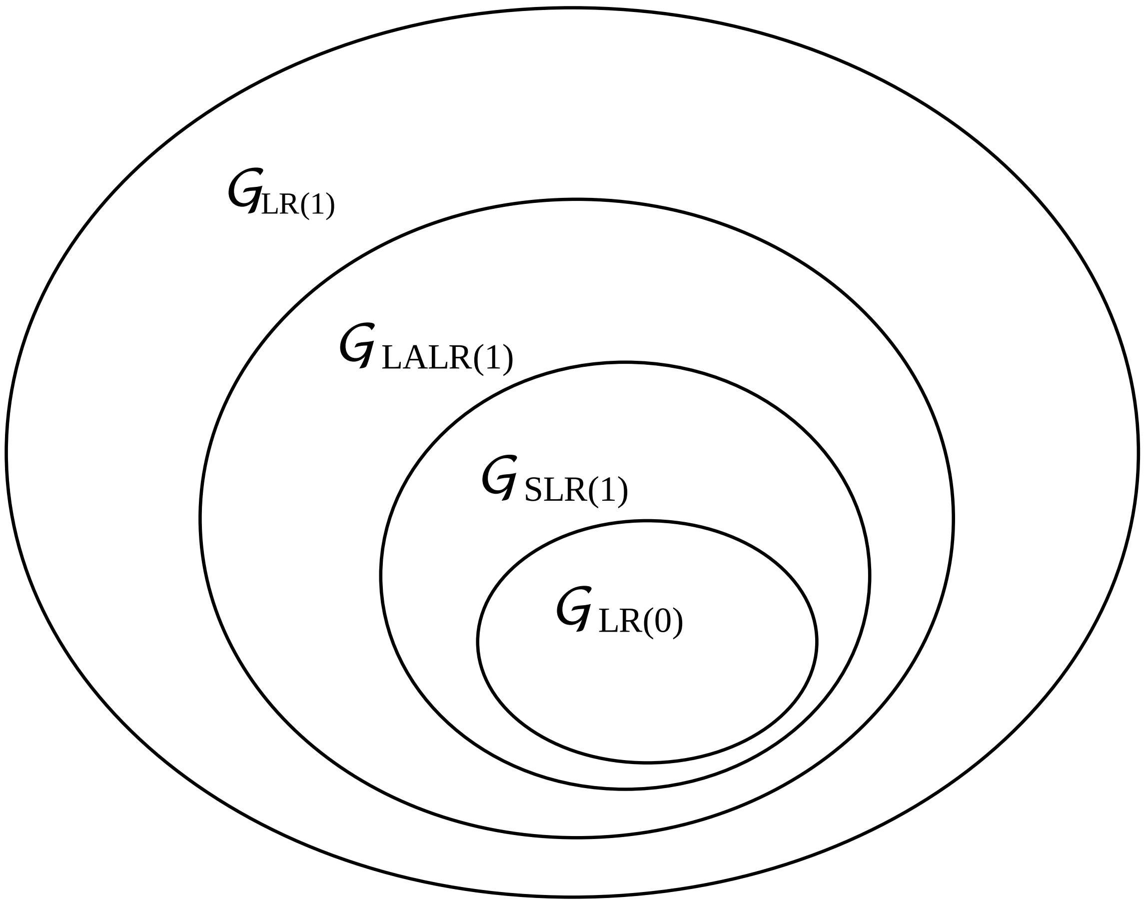
Przykład

TABLICA LALR(1)

stan	f				g		
	$	=	*	id	S	L	R
T_0			shift-4	shift-5	T_1	T_2	T_3
T_1	acc						
T_2	red-5	shift-6					
T_3	red-2						
T_4			shift-4	shift-5		T_8	T_7
T_5	red-4	red-4					
T_6			shift-4	shift-5		T_8	T_9
T_7	red-3	red-3					
T_8	red-5	red-5					
T_9	red-1						

(Kanoniczny LR(1) ma o 4 stany więcej)

Zależności pomiędzy klasami gramatyk



Każda gramatyka LR(0) jest gramatyką SLR(1).
Każda gramatyka SLR(1) jest gramatyką LALR(1).
Każda gramatyka LALR(1) jest gramatyką LR(1).

$$\mathcal{G}_{LR(0)} \subset \mathcal{G}_{SLR(1)} \subset \mathcal{G}_{LALR(1)} \subset \mathcal{G}_{LR(1)}$$

$$\mathcal{G}_{LR(0)} \neq \mathcal{G}_{SLR(1)} \neq \mathcal{G}_{LALR(1)} \neq \mathcal{G}_{LR(1)}$$

Języki bezkontekstowe deterministyczne a języki LR

Język bezkontekstowy nazywamy deterministycznym, gdy jest on akceptowany przez deterministyczny automat ze stosem.

$$L_{\text{CFL DET}} = L_{\text{LR(1)}}$$

Każdy język bezkontekstowy deterministyczny ma swoją gramatykę LR(1).

Każdy język bezkontekstowy deterministyczny posiadający własność przedrostkową ma swoją gramatykę LR(0).

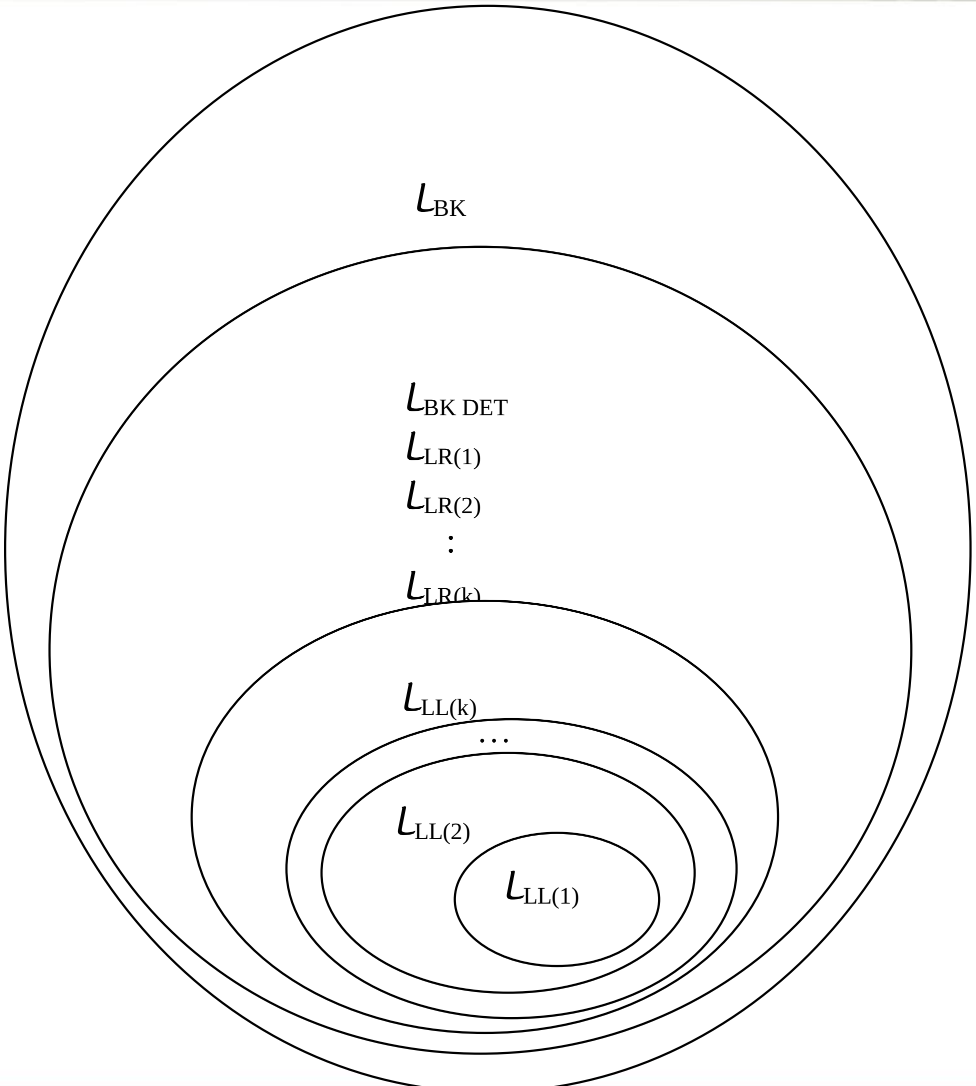
Języki LL a języki LR

$$L_{\mathrm{LL(K)}} \subset L_{\mathrm{LR(K)}}$$
$$L_{\mathrm{LL(K)}} \neq L_{\mathrm{LR(K)}}$$

Każda gramatyka LL(k) jest gramatyką LR(k). Istnieją języki LR(k), które nie są generowane przez żadną gramatykę LL(k).

Języki LL(0) są jednoelementowe.

Podklasy języków bezkontekstowych



Konstrukcja parserów bottom-up

Typ gramatyki	Automat
LR(0)	LR(0)-sytuacje
SLR(1)	LR(0)-sytuacje
LALR(1)	LR(0)-sytuacje lub LR(1)-sytuacje
LR(1)	LR(1)-sytuacje

Konstrukcja parserów bottom-up

Typ gramatyki	Redukcje
LR(0)	reduce(q,a) = { $A\to\alpha$: $A\to\alpha.\in q$ }
SLR(1)	reduce(q,a) = {$A\to\alpha$: $A\to\alpha.\in q$ i $a\epsilon Follow(A)$ }
LALR(1)	reduce(q,a) = {$A\to\alpha$: $A\to\alpha.\in q$ i $a\epsilon LA(q,A\to\alpha)$ }
LR(1)	reduce(q,a) = { $A\to\alpha$: ($A\to\alpha.$, a) $\in q$ }

Złożoność algorytmów konstrukcji parserów bottom-up

- N – suma długości prawych stron produkcji

Typ gramatyki	Złożoność
LL(1)	N^2
SLR(1)	$2^{N+\log N}$
LALR(1)	$2^{2N+\log N}$
LR(1)	$2^{N^2+\log N}$
SLR(k)	$2^{N+k\cdot\log N}$
LR(k)	$2^{N^{(k+1)}+\log N}$

- Konstrukcja LALR(1)-automatów najbardziej efektywna
- Najczęściej stosowana do konstrukcji analizatorów składniowych języków programowania