

Построение SLR-анализатора для грамматики

1. Исходная грамматика

$$\begin{array}{l} S \rightarrow aB \mid c \\ A \rightarrow cB \mid c \\ B \rightarrow Ba \mid Aa \mid \varepsilon \end{array}$$

2. FIRST и FOLLOW для грамматики

Множества FIRST:

$$\begin{aligned} \text{FIRST}(S) &= \{a, c\} \\ \text{FIRST}(A) &= \{c\} \\ \text{FIRST}(B) &= \{a, c, \varepsilon\} \end{aligned}$$

Множества FOLLOW:

$$\begin{aligned} \text{FOLLOW}(S) &= \{\$\} \\ \text{FOLLOW}(A) &= \{a\} \\ \text{FOLLOW}(B) &= \{\$, a\} \end{aligned}$$

3. Построение множества пунктов (LR(0)-пунктов)

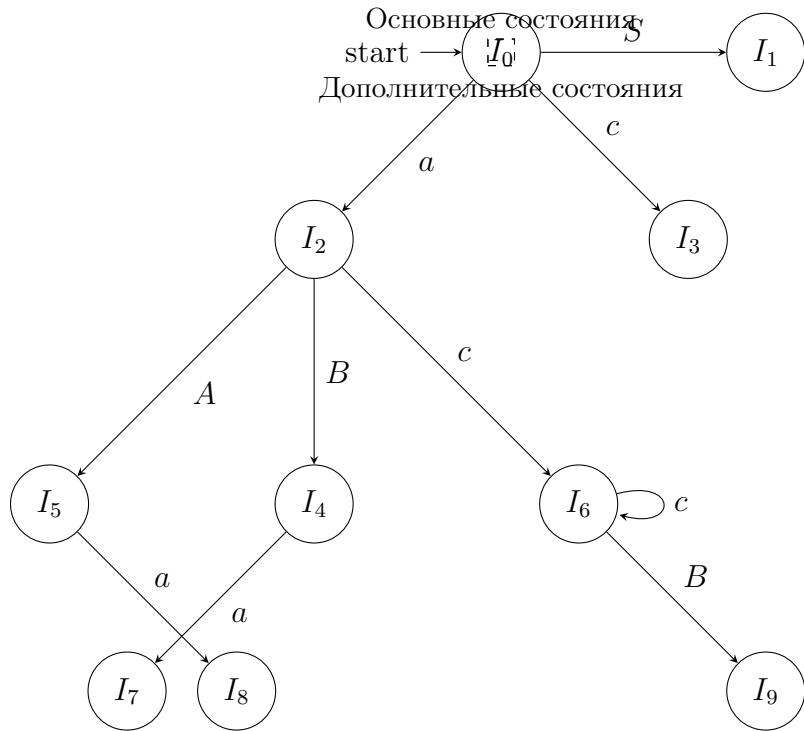
Расширим грамматику добавлением нового начального символа S' :

$$\begin{array}{ll} 0: & S' \rightarrow S \\ 1: & S \rightarrow aB \\ 2: & S \rightarrow c \\ 3: & A \rightarrow cB \\ 4: & A \rightarrow c \\ 5: & B \rightarrow Ba \\ 6: & B \rightarrow Aa \\ 7: & B \rightarrow \varepsilon \end{array}$$

Каноническое множество LR(0)-пунктов:

1. $I_0: \{S' \rightarrow \bullet S, S \rightarrow \bullet aB, S \rightarrow \bullet c\}$
2. $I_1: goto(I_0, S) = \{S' \rightarrow S\bullet\}$
3. $I_2: goto(I_0, a) = \{S \rightarrow a\bullet B, B \rightarrow \bullet Ba, B \rightarrow \bullet Aa, B \rightarrow \bullet, A \rightarrow \bullet cB, A \rightarrow \bullet c\}$
4. $I_3: goto(I_0, c) = \{S \rightarrow c\bullet\}$
5. $I_4: goto(I_2, B) = \{S \rightarrow aB\bullet, B \rightarrow B \bullet a\}$
6. $I_5: goto(I_2, A) = \{B \rightarrow A \bullet a\}$
7. $I_6: goto(I_2, c) = \{A \rightarrow c \bullet B, A \rightarrow c\bullet, B \rightarrow \bullet Ba, B \rightarrow \bullet Aa, B \rightarrow \bullet, A \rightarrow \bullet cB, A \rightarrow \bullet c\}$
8. $I_7: goto(I_4, a) = \{B \rightarrow Ba\bullet\}$
9. $I_8: goto(I_5, a) = \{B \rightarrow Aa\bullet\}$
10. $I_9: goto(I_6, B) = \{A \rightarrow cB\bullet, B \rightarrow B \bullet a\}$

4. Диаграмма переходов LR(0)-автомата



5. Таблица SLR-анализатора

Состояние	ACTION				GOTO				Восстановление
	a	c	$\$$	Ошибка	S	A	B	Ошибка	
0	s2	s3	-	-	1	-	-	-	-
1	-	-	acc	-	-	-	-	-	-
2	r7	s6	r7	-	-	5	4	-	При a или $\$$: свёртка $B \rightarrow$
3	-	-	r2	-	-	-	-	-	-
4	s7	-	r1	-	-	-	-	-	-
5	s8	-	-	-	-	-	-	-	-
6	r4/r7	s6	r7	конфликт	-	5	9	-	При a : выбрать r4 ($A \rightarrow c$)
7	r5	-	r5	-	-	-	-	-	-
8	r6	-	r6	-	-	-	-	-	-
9	s7	-	-	-	-	-	-	-	-

Обозначения:

- sn - сдвиг и переход в состояние n
- rn - свёртка по правилу n
- acc - допуск (успешный разбор)
- $r4/r7$ - конфликт свёртка/свёртка

Конфликты и их разрешение:

- В состоянии 6 при входе a : конфликт между $r4$ ($A \rightarrow c$) и $r7$ ($B \rightarrow$).
Разрешение: выбираем $r4$ (свёртка по $A \rightarrow c$), так как это правило имеет более высокий приоритет.
- В состоянии 9 при входе a : возможен конфликт сдвиг/свёртка ($s7$ и $r3$), но в SLR его нет, так как $\text{FOLLOW}(A) = \{a\}$.

6. Проверка на примерах

Пример 1 (правильный): а а \$

Стек	Вход	Действие	Комментарий
0	а а \$	$s2$	Сдвиг
0 а 2	а \$	$r7$ ($B \rightarrow$)	Свёртка
0 а 2 В 4	а \$	$s7$	Сдвиг
0 а 2 В 4 а 7	\$	$r5$ ($B \rightarrow Ba$)	Свёртка
0 а 2 В 4	\$	$r1$ ($S \rightarrow aB$)	Свёртка
0 S 1	\$	acc	Допуск

Вывод: Цепочка а а \$ принимается.

Пример 2 (неправильный): с с \$

Стек	Вход	Действие	Комментарий
0	с с \$	$s3$	Сдвиг
0 с 3	с \$	ошибка	Нет действия для 'с' в состоянии 3

Восстановление:

1. Удалить ошибочный символ 'с'
2. Продолжить разбор: состояние 3, вход \$
3. Выполнить $r2$ ($S \rightarrow c$)
4. Получить 0 S 1, вход \$
5. Допуск

Пример 3 (неправильный): а с \$

Стек	Вход	Действие	Комментарий
0	а с \$	s2	Сдвиг
0 а 2	с \$	s6	Сдвиг
0 а 2 с 6	\$	r7 (B→)	Свёртка
0 а 2 с 6 В 9	\$	ошибка	Нет действия для '\$' в состоянии 9

Восстановление:

1. Вставить символ 'а' (ожидался по FOLLOW(A) и FOLLOW(B))
2. Продолжить: состояние 9, вход а\$
3. Выполнить s7
4. Состояние 7, вход \$
5. Выполнить r5 (B→Ba)
6. Состояние 4, вход \$
7. Выполнить r1 (S→aB)
8. Получить 0 S 1, вход \$
9. Допуск

7. Алгоритм восстановления после ошибок

Для данного SLR-анализатора реализованы следующие стратегии восстановления:

1. **Режим паники:** при обнаружении ошибки пропускать входные символы до тех пор, пока не будет найден символ из множества FOLLOW текущего нетерминала.
2. **Вставка недостающих символов:** если в таблице ACTION нет записи для текущего состояния и входного символа, но есть ожидаемые символы из FOLLOW, можно вставить один из них.
3. **Удаление ошибочных символов:** если входной символ не может быть обработан в текущем состоянии, его можно удалить.
4. **Использование свёрток по умолчанию:** при конфликтах свёртка/свёртка выбирается свёртка с меньшим номером правила.