

§ 1. LR(0)-автомат. LR(0)-грамматика. Анализатор на основе LR(0)-автомата

Ключевые определения:

Опр. Пусть $S \Rightarrow \gamma_1 \Rightarrow \dots \Rightarrow \gamma_m = \omega$ правосторонний вывод цепочки ω из аксиомы S .

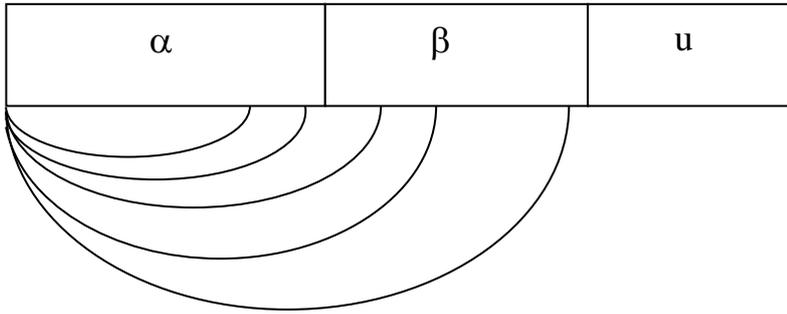
r-формой (right-form) называется каждая γ_i .

Опр. Пусть в правостороннем выводе $\alpha A u \Rightarrow \alpha \beta u$ было использовано правило $A \rightarrow \beta$.

Основой r -формы $\alpha \beta u$ называется β .

Комментарий: при работе анализатора цепочка β на вершшке стека свернулась в A .

Опр. Активным префиксом r -формы $\alpha\beta u$ называется любой префикс цепочки $\alpha\beta$ (т.е. префикс, не выходящий за правый конец основы).



Опр. LR(0)-пунктом (пункт) называется набор (A, β_1, β_2) , где $A \rightarrow \beta_1\beta_2$ – правило грамматики.

Обозначение пункта: $[A \rightarrow \beta_1 \bullet \beta_2]$ («Правило с точкой»).

Все LR(0)-пункты – допустимые. Определение допустимости будет позже.

Комментарий: очевидно для любой грамматики количество всех возможных пунктов конечно.

Опр. Пусть $G = (\Sigma, \Gamma, P, S)$ – грамматика. Расширенной грамматикой называется $G' = (\Sigma, \Gamma \cup \{S'\}, P \cup \{S' \rightarrow S\}, S')$, где $S' \notin \Gamma$ (новый нетерминал = новая аксиома).

Опр. Автоматом пунктов расширенной грамматики $G = (\Sigma, \Gamma, P, S')$ называется ε -НКА $(Q, \Sigma \cup \Gamma, \delta, q_0, Q)$, где

Q – множество пунктов (один пункт = одно состояние);

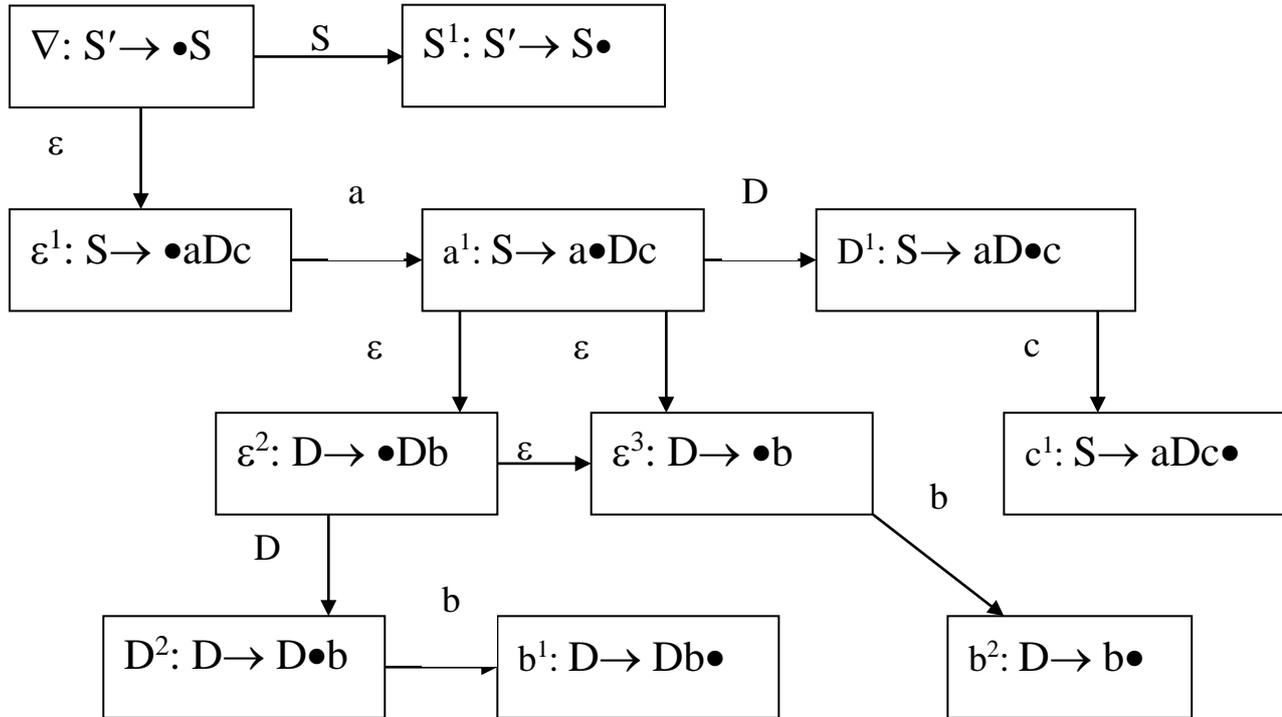
$q_0 = [S' \rightarrow \bullet S]$;

$$\delta = \bigcup_{\forall X \in \Sigma \cup \Gamma} (\delta([A \rightarrow \beta_1 \bullet X \beta_3], X) = [A \rightarrow \beta_1 X \bullet \beta_3]) \cup$$

$$\cup \bigcup_{\forall B \in \Gamma} (\delta([A \rightarrow \beta_1 \bullet B \beta_3], \varepsilon) = [B \rightarrow \bullet \gamma]).$$

Пример. $G = \{S' \rightarrow S, S \rightarrow aDc, D \rightarrow Db \mid b\}$.
(стр. 161-162 книжки Шура и Замятина)

$G = \{S' \rightarrow S, S \rightarrow aDc, D \rightarrow Db \mid b\}$.

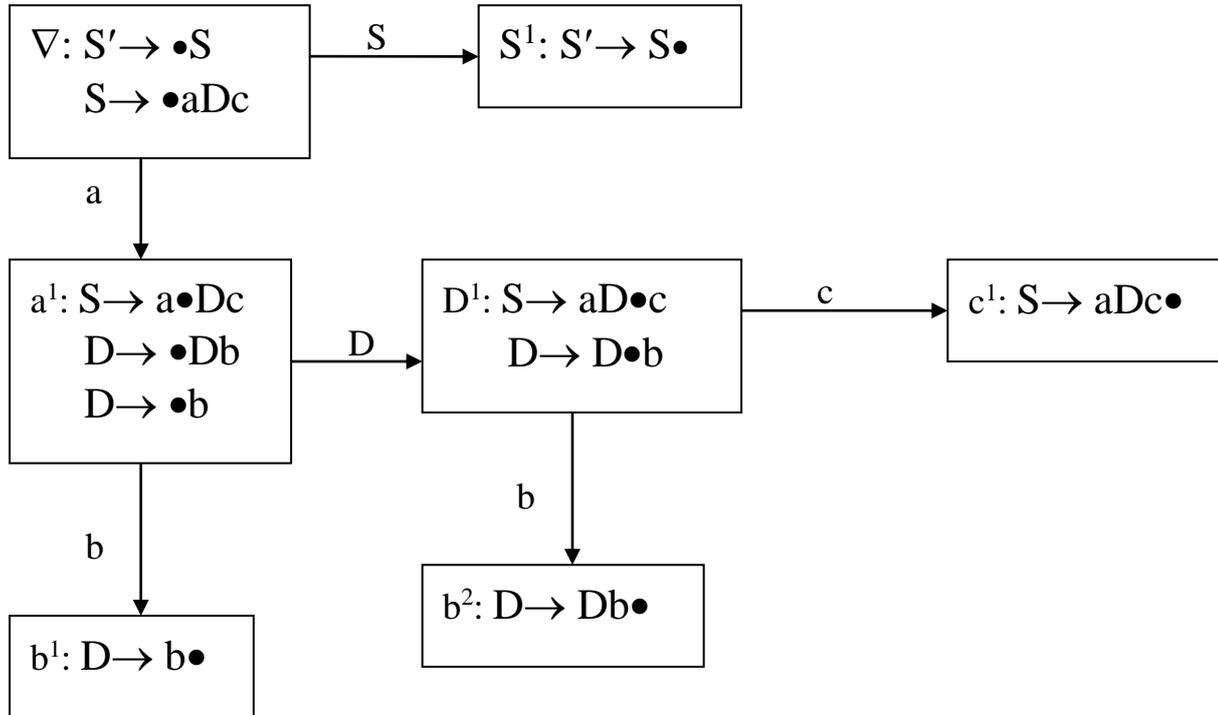


Опр. LR(0)-автоматом грамматики G называется ε -замыкание автомата пунктов.

Комментарий: в задачах на практических занятиях будем строить ε -замыкание без поиска ε -НКА пунктов.

Пример. $G = \{S' \rightarrow S, S \rightarrow aDc, D \rightarrow Db \mid b\}$.

- 1) Каждое состояние LR(0)-автомата – набор пунктов, где $A \rightarrow \alpha \bullet B \beta$ влечет за собой $B \rightarrow \bullet \gamma$.
- 2) Один пункт может содержаться в нескольких состояниях.
- 3) Состояния равны, только если множества пунктов совпадают.



Опр. LR-анализатором называется таблица ACTION/GOTO, в которой строки соответствуют состояниям автомата, столбцы – символам $\Sigma \cup \Gamma \cup \{\mid\}$:

	ACTION		GOTO
	Σ	\mid	Γ
состояние			

В части ACTION применяются пометки вида:

\surd = «допуск» (цепочка принадлежит языку);

$\leftarrow q$ = «перенос» (где q – название состояния в автомате, по которому строится анализатор);

$\otimes n$ = «свертка» по правилу с номером n .

Опр. Таблица анализатора бесконфликтная, если в каждой ячейке не более одного действия.

Существует два вида конфликтов:
перенос-свертка,
свертка-свертка.

Опр. (Один хороший случай бесконфликтного анализатора).
Грамматика называется LR(0)-грамматикой, если в LR(0)-автомате
каждое состояние, содержащее пункт с точкой на конце ($[A \rightarrow \alpha \bullet]$),
состоит из единственного пункта.

Пример, рассмотренный ранее. $G = \{S' \rightarrow S, S \rightarrow aDc, D \rightarrow Db \mid b\}$.

Вопрос: является ли LR(0)-грамматикой?

Алгоритм построения LR(0)-анализатора
(бесконфликтный анализатор для LR(0)-грамматики).

Пусть $(Q, \Sigma \cup \Gamma, \delta, \nabla, Q)$ – LR(0)-автомат.

Заполнение ACTION:

- 1) В строке, соответствующей $[S' \rightarrow S \bullet]$ в столбце \mid заносим \surd (допуск).
- 2) В строках, соотв. $[A \rightarrow \alpha \bullet]$, кроме заполненной в 1), в каждом столбце, включая \mid заносим \otimes_n (свертка).
- 3) В строках q , не заполненных в 1) и 2), для $a \in \Sigma \cup \{\mid\}$ заносим $\leftarrow \delta(q, a)$, если $\delta(q, a) \neq \emptyset$ (перенос).

Заполнение GOTO:

В каждой строке q для $A \in \Gamma$ заносим $\delta(q, A)$, если $\delta(q, A) \neq \emptyset$.

Пример, рассмотренный ранее. $G = \{S' \xrightarrow{1} S, S \xrightarrow{2} aDc, D \xrightarrow{3} Db \mid b\}$.

	ACTION				GOTO	
	<i>a</i>	<i>b</i>	<i>c</i>	—	S	D
∇						
a^1						
b^1						
b^2						
c^1						
S^1						
D^1						

	ACTION				GOTO	
	a	b	c	\neg	S	D
∇	$\leftarrow a^1$				S^1	
a^1		$\leftarrow b^1$				D^1
b^1	$\otimes 4$	$\otimes 4$	$\otimes 4$	$\otimes 4$		
b^2	$\otimes 3$	$\otimes 3$	$\otimes 3$	$\otimes 3$		
c^1	$\otimes 2$	$\otimes 2$	$\otimes 2$	$\otimes 2$		
S^1				$\sqrt{\quad}$		
D^1		$\leftarrow b^2$	$\leftarrow c^1$			

Комментарий: пустые ячейки в анализаторе означают, что входная цепочка не соответствует правилам языка.

В таблице ACTION можно добавить действия, называемые «обработчиком ошибок».

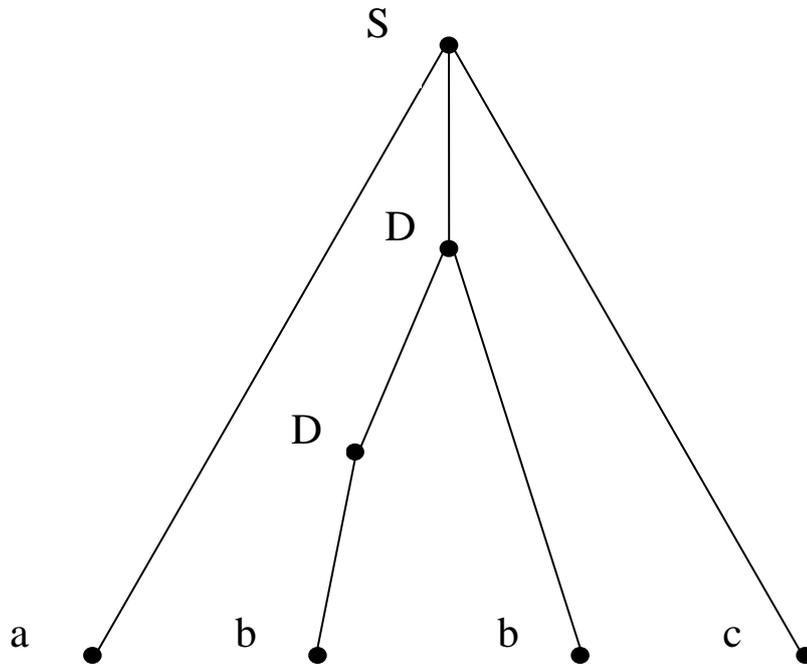
В таблице GOTO обращение к пустым ячейкам невозможно.

Пример. Рассмотрим анализ цепочки $\omega = abbc$.

состояние	№ такта	стек	указатель	действие
∇	1	∇	$\diamond abbc \mid$	

состояние	№ такта	стек	указатель	действие
∇	1	∇	$\diamond abbc \text{---} \vdash$	
				перенос
a^1	2	∇a^1	$a \diamond bbc \text{---} \vdash$	
				перенос
b^1	3	$\nabla a^1 \underline{b^1}$	$ab \diamond bc \text{---} \vdash$	
				свертка по 4
$D^1 = \text{GOTO}(a^1, D)$	4	$\nabla a^1 D^1$	$ab \diamond bc \text{---} \vdash$	
				перенос
b^2	5	$\nabla a^1 \underline{D^1} \underline{b^2}$	$abb \diamond c \text{---} \vdash$	
				свертка по 3
$D^1 = \text{GOTO}(a^1, D)$	6	$\nabla a^1 D^1$	$abb \diamond c \text{---} \vdash$	
				перенос
c^1	7	$\nabla \underline{a^1} \underline{D^1} \underline{c^1}$	$abc \diamond \text{---} \vdash$	
				свертка по 2
$S = \text{GOTO}(\nabla, S)$	8	∇S	$abc \diamond \text{---} \vdash$	
				допуск

Восстановленное дерево вывода:



Восстановленный вывод:

$S \Rightarrow aDc \Rightarrow aDbc \Rightarrow abbc.$