Лекции по теории формальных языков

Лекция 2.
Недетерминированные конечные автоматы.
Операции над автоматными языками.
Регулярные языки.
Автоматные фрагменты языков программирования

Александр Сергеевич Герасимов
`http://gas-teach.narod.ru`

Кафедра математических и информационных технологий
Санкт-Петербургского академического университета
Российской академии наук.
Весенний семестр 2010/11 учебного года

18 февраля 2011 г.

План

План

Определение недетерминированного конечного автомата

Недетерминированным конечным автоматом (*НКА*) называется пятёрка $\mathcal{B} = (Q, \Sigma, \delta, Q_0, F)$, где

- Q — непустое конечное множество *состояний*,
- Σ — алфавит,
- $\delta : Q \times \Sigma \to 2^Q$ — *функция переходов*,
- $Q_0 \subseteq Q$ — множество *начальных* состояний,
- $F \subseteq Q$ — множество *заключительных* (или *допускающих*) состояний.

(δ может быть определено и как отношение $\delta \subseteq Q \times \Sigma \times Q$.)

НКА, находящийся в состоянии $\mathtt{q}$ и обозревающий ячейку с символом $\mathtt{a}$, может перейти в любое состояние из множества $\delta(\mathtt{q}, \mathtt{a})$.

Язык, распознаваемый НКА

- Доопределим функцию переходов δ на $Q \times \Sigma^*$:
$$\delta(\mathtt{q}, \varepsilon) = \{\mathtt{q}\}, \qquad \delta(\mathtt{q}, u\mathtt{a}) = \bigcup_{\mathtt{r} \in \delta(\mathtt{q}, u)} \delta(\mathtt{r}, \mathtt{a}).$$
- НКА $\mathcal{B} = (Q, \Sigma, \delta, Q_0, F)$ *распознает* (или *допускает*) цепочку $w \in \Sigma^*$, если
$$\left(\bigcup_{\mathtt{q} \in Q_0} \delta(\mathtt{q}, w)\right) \bigcap F \neq \emptyset.$$
- Множество всех цепочек, допускаемых автоматом $\mathcal{B}$, называется *языком, распознаваемым автоматом* $\mathcal{B}$, и обозначается $L(\mathcal{B})$.

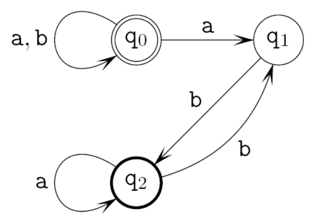
Задание НКА расширенной таблицей переходов

НКА $\mathcal{B}$:

	a	b	Q_0	F
$\mathtt{q}_0$	$\mathtt{q}_0, \mathtt{q}_1$	$\mathtt{q}_0$	1	0
$\mathtt{q}_1$		$\mathtt{q}_2$	0	0
$\mathtt{q}_2$	$\mathtt{q}_2$	$\mathtt{q}_1$	0	1

Задание НКА диаграммой переходов

НКА $\mathcal{B}$ тот же, что и на предыдущем слайде.



- $\mathtt{aab} \in L(\mathcal{B})$: $\delta(\mathtt{q}_0, \mathtt{a}) = \{\mathtt{q}_0, \mathtt{q}_1\} = \delta(\mathtt{q}_0, \mathtt{aa})$, $\delta(\mathtt{q}_0, \mathtt{aab}) = \{\mathtt{q}_0, \mathtt{q}_2\} \cap F \neq \emptyset$.
- $\mathtt{abba} \notin L(\mathcal{B})$: $\delta(\mathtt{q}_0, \mathtt{abba}) = \{\mathtt{q}_0, \mathtt{q}_1\} \cap F = \emptyset$.

Теорема Рабина-Скотта

Теорема (Рабина-Скотта)

Класс языков, распознаваемых НКА, совпадает с классом языков, распознаваемых ДКА.

Д о к а з а т е л ь с т в о. ДКА является частным случаем НКА, поэтому класс языков, распознаваемых ДКА, содержится в классе языков, распознаваемых НКА. Докажем обратное включение.

- Пусть $\mathcal{B} = (Q, \Sigma, \delta, Q_0, F)$ — произвольный НКА. Возьмём ДКА $\mathcal{A} = (2^Q, \Sigma, \delta', Q_0, F')$, где
$$\delta'(P, \mathtt{a}) = \bigcup_{\mathtt{q} \in P} \delta(\mathtt{q}, \mathtt{a}), \qquad F' = \{P \in 2^Q \mid P \cap F \neq \emptyset\},$$
и докажем, что $L(\mathcal{A}) = L(\mathcal{B})$.

Теорема Рабина-Скотта: продолжение доказательства

- Индукцией по длине произвольной $w \in \Sigma^*$ установим, что для любого $P \in 2^Q$
$$\delta'(P, w) = \bigcup_{\mathtt{q} \in P} \delta(\mathtt{q}, w). \qquad (\star)$$
- База индукции ($|w| = 0$) верна, поскольку
$$\delta'(P, \varepsilon) = P = \bigcup_{\mathtt{q} \in P} \delta(\mathtt{q}, \varepsilon).$$
- Индукционный переход. Рассмотрим $w = u\mathtt{a}$.
$$\delta'(P, u\mathtt{a}) = \delta'(\delta'(P, u), \mathtt{a}) =_{\text{инд. предп.}} \delta'\left(\bigcup_{\mathtt{q} \in P} \delta(\mathtt{q}, u),\ \mathtt{a}\right) =_{\text{опр. } \delta'}$$
$$\bigcup_{\mathtt{r} \in \bigcup_{\mathtt{q} \in P} \delta(\mathtt{q}, u)} \delta(\mathtt{r}, \mathtt{a}) = \bigcup_{\mathtt{q} \in P} \bigcup_{\mathtt{r} \in \delta(\mathtt{q}, u)} \delta(\mathtt{r}, \mathtt{a}) = \bigcup_{\mathtt{q} \in P} \delta(\mathtt{q}, u\mathtt{a}).$$

Теорема Рабина-Скотта: окончание доказательства

- Наконец, пользуясь определениями и беря Q_0 в качестве P в $(\star)$, получаем

$$L(\mathcal{A}) = \{w \in \Sigma^* \mid \delta'(Q_0, w) \in F'\} = \{w \in \Sigma^* \mid \delta'(Q_0, w) \cap F \neq \emptyset\} = \{w \in \Sigma^* \mid \bigcup_{q \in Q_0} \delta(q, w) \bigcap F \neq \emptyset\} = L(\mathcal{B}).$$

□

Применённый в доказательстве теоремы способ получения ДКА по НКА называется «построением подмножеств».

Алгоритм нахождения ДКА по НКА (построением подмножеств).

Вход. НКА $\mathcal{B} = (Q, \Sigma, \delta, Q_0, F)$.
Выход. Эквивалентный ДКА $\mathcal{A} = (Q', \Sigma, \delta', Q_0, F')$ без недостижимых состояний.

1. **для каждого** $P \subseteq Q$ $label(P) := 0$;
2. $Q' := \{Q_0\}$;
3. **пока** $(\exists P \in Q' : label(P) = 0)$ **повторять**
4. **для каждого** $a \in \Sigma$
5. $\delta'(P, a) := \bigcup_{q \in P} \delta(q, a)$;
6. $Q' := Q' \cup \{\delta'(P, a)\}$;
7. $label(P) := 1$;
8. $F' := \{P \in Q' \mid P \cap F \neq \emptyset\}$

Все состояния автомата $\mathcal{A}$ достижимы, поскольку каждое состояние в Q', кроме начального, получено переходом в него из ранее построенного состояния.

Пример построения ДКА по НКА

	a	b	Q_0	F
q_0	q_0, q_1	q_0	1	0
q_1		q_2	0	0
q_2	q_2	q_1	0	1

	a	b	F
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$	0
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$	0
$\{q_0, q_2\}$	$\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	1
$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$	1

Определение недетерминированного конечного автомата с ε-переходами

Определяемые здесь автоматы будет удобно строить по заданным языкам.

Недетерминированным конечным автоматом с ε-переходами (ε-*НКА*) называется пятёрка $\mathcal{B} = (Q, \Sigma, \delta, Q_0, F)$, где

- Q, Σ, Q_0, F — те же, что и в случае НКА,
- $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \to 2^Q$ — *функция переходов*.

ε-НКА может не сдвигаться по входной цепочке по окончании некоторых тактов (на таком такте автомат прочитывает ε).

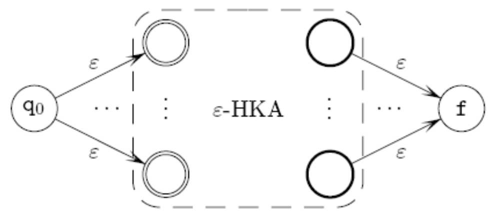
Диаграмма переходов ε-НКА может содержать дуги, помеченные ε.

ε-НКА распознает цепочку w $\Leftrightarrow$ в диаграмме переходов существует путь из начального состояния в заключительное, помеченный w.

(Определение распознаваемой цепочки через обобщение функции переходов будет дано ниже.)

Нормальные ε-НКА

- Произвольный ε-НКА эквивалентен ε-НКА с единственным начальным и единственным заключительным состоянием: добавим следующим образом к исходному автомату состояния q_0 и f



и объявим $\{q_0\}$ множеством начальных состояний, а $\{f\}$ — множеством заключительных состояний нового автомата.
- ε-НКА вида $(Q, \Sigma, \delta, \{q_0\}, \{f\})$ будем называть *нормальными*.
- В дальнейшем мы рассматриваем только нормальные ε-НКА.

Цепочки, распознаваемые (нормальным) ε-НКА

Пусть $\mathcal{B} = (Q, \Sigma, \delta, \{q_0\}, \{f\})$ — (нормальный) ε-НКА.

- Определим отношение $\rho \subseteq Q^2$: $(q, p) \in \rho \Leftrightarrow p \in \delta(q, \varepsilon)$.
- $(q, p) \in \rho^*$ (рефлексивно-транзитивное замыкание отношения ρ) $\Leftrightarrow$ автомат может перейти из состояния q в состояние p, не сдвигаясь по входной цепочке.
- Замыкание состояния q: $Clo(q) = \{p \in Q \mid (q, p) \in \rho^*\}$.
- Замыкание множества состояний $P \subseteq Q$: $Clo(P) = \bigcup_{q \in P} Clo(q)$.
- Обобщённая функция переходов $\bar{\delta} : Q \times \Sigma^* \to 2^Q$,

$$\bar{\delta}(q, \varepsilon) = Clo(q), \qquad \bar{\delta}(q, ua) = \bigcup_{r \in \bar{\delta}(q,u)} Clo(\delta(r, a)).$$

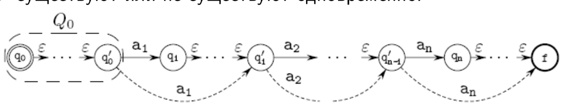
- $r \in \bar{\delta}(q, w)$ $\Leftrightarrow$ в диаграмме переходов автомата $\mathcal{B}$ существует путь из q в r, помеченный w.
- Цепочка w *распознаётся* автоматом $\mathcal{B}$, если $f \in \bar{\delta}(q_0, w)$.

Теорема

Класс языков, распознаваемых ε-НКА, совпадает с классом языков, распознаваемых НКА.

Доказательство.

- Пусть $\mathcal{B} = (Q, \Sigma, \delta, \{q_0\}, \{f\})$ — (нормальный) ε-НКА. Определим НКА $\mathcal{B}' = (Q, \Sigma, \delta', Q_0, \{f\})$, где $Q_0 = Clo(q_0)$, $\delta'(q, a) = \bar{\delta}(q, a)$.
- Пути, помеченные цепочкой $w \in \Sigma^*$, в диаграммах переходов $\mathcal{B}$ и $\mathcal{B}'$ существуют или не существуют одновременно:



- Таким образом, множества цепочек, распознаваемых $\mathcal{B}$ и $\mathcal{B}'$, совпадают.

□

Алгоритм нахождения ДКА по ε-НКА

Вход. ε-НКА $\mathcal{B} = (Q, \Sigma, \delta, \{q_0\}, F)$.
Выход. Эквивалентный ДКА $\mathcal{A} = (Q', \Sigma, \delta', Q_0, F')$ без недостижимых состояний.

1. **для каждого** $P \subseteq Q$ $label(P) := 0$;
2. $Q_0 := Clo(q_0)$; $Q' := \{Q_0\}$;
3. **пока** $(\exists P \in Q' : label(P) = 0)$ **повторять**
4. **для каждого** $a \in \Sigma$
5. $\delta'(P, a) := \bigcup_{q \in P} \bar{\delta}(q, a)$;
6. $Q' := Q' \cup \{\delta'(P, a)\}$;
7. $label(P) := 1$;
8. $F' := \{P \in Q' \mid P \cap F \neq \emptyset\}$

Это видоизменение алгоритма, описанного на слайде 11. Замыкание $Clo(q)$ вычисляется поиском из состояния q.

План

Определение автоматного языка. Теорема о замкнутости класса автоматных языков относительно некоторых операций

Назовём язык *автоматным*, если он распознаётся некоторым конечным автоматом (каким?). Обозначим через $\mathbb{A}$ класс всех автоматных языков.

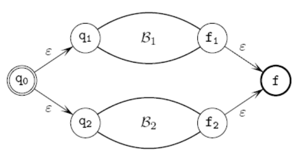
Теорема

Класс $\mathbb{A}$ замкнут относительно объединения, пересечения, дополнения, произведения и итерации.

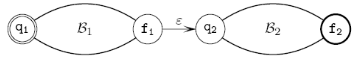
Доказательство.

- *Замкнутость относительно объединения.* Пусть $\mathcal{B}_1 = (Q_1, \Sigma, \delta_1, \{q_1\}, \{f_1\})$ и $\mathcal{B}_2 = (Q_2, \Sigma, \delta_2, \{q_2\}, \{f_2\})$ — нормальные ε-НКА, $Q_1 \cap Q_2 = \emptyset$. Тогда язык $L(\mathcal{B}_1) \cup L(\mathcal{B}_2)$ распознаётся следующим нормальным ε-НКА $(Q_1 \cup Q_2 \cup \{q_0, f\}, \Sigma, \delta_3, \{q_0\}, \{f\})$:

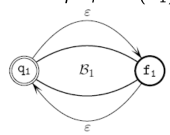
Теорема: продолжение доказательства



- *Замкнутость относительно произведения.* $L(\mathcal{B}_1)L(\mathcal{B}_2)$



- *Замкнутость относительно итерации.* $L(\mathcal{B}_1)^*$



Теорема: окончание доказательства

- *Замкнутость относительно дополнения.* Пусть $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ — (полный) ДКА. Тогда язык $\overline{L(\mathcal{A})}$ распознаётся ДКА $(Q, \Sigma, \delta, q_0, Q \setminus F)$.
- *Замкнутость относительно пересечения.* $K \cap L = \overline{\overline{K} \cup \overline{L}}$

□

Автоматность конечных языков

Предложение

Любой конечный язык является автоматным.

Доказательство.

- Языки $\emptyset$, $\{\varepsilon\}$, $\{a\}$ (где a — символ) распознаются следующими автоматами:

- Любой язык, состоящий из одной цепочки, либо совпадает с $\{\varepsilon\}$, либо является произведением конечного числа языков из одного символа, поэтому по предыдущей теореме такой язык является автоматным.
- Любой конечный язык либо совпадает с $\emptyset$, либо является объединением конечного числа языков из одной цепочки, поэтому по предыдущей теореме такой язык является автоматным.

□

План

Определение регулярного языка. Теорема Клини

Язык называется *регулярным*, если он может быть получен из конечных языков (или языков вида $\emptyset$, $\{\varepsilon\}$, $\{a\}$, где a — символ) в результате конечного числа операций объединения, произведения и итерации. Обозначим через $\mathbb{R}$ класс всех регулярных языков.

Теорема (теорема Клини)

$\mathbb{R} = \mathbb{A}$.

Доказательство. $\mathbb{R} \subseteq \mathbb{A}$ по предыдущим теореме и предложению. Покажем, что $\mathbb{A} \subseteq \mathbb{R}$.

Пусть $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ — неполный ДКА. Индукцией по числу переходов автомата $\mathcal{A}$ докажем, что язык $L(\mathcal{A})$ регулярен.

Теорема Клини: продолжение доказательства

База индукции.

- Функция переходов δ нигде не определена.
Тогда $L(\mathcal{A}) = \{\varepsilon\}$, если $q_0 \in F$, и $L(\mathcal{A}) = \emptyset$ иначе.
- Существует ровно один переход $\delta(q, a) = r$.
Если $q \neq q_0$, то имеем: $L(\mathcal{A}) = \{\varepsilon\}$ при $q_0 \in F$, а иначе $L(\mathcal{A}) = \emptyset$.
Разберём все возможные случаи, если $q = q_0$:

$r = q_0$	$q \in F$	$r \in F$	$L(\mathcal{A})$
нет	нет	нет	$\emptyset$
нет	нет	да	$\{a\}$
нет	да	нет	$\{\varepsilon\}$
нет	да	да	$\{\varepsilon, a\}$
да	нет	нет	$\emptyset$
да	да	да	$\{a\}^*$

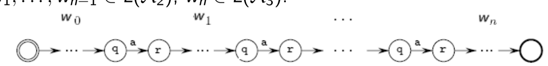
Теорема Клини: продолжение доказательства

Индукционый переход. Пусть автомат $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ имеет $k > 1$ переходов.

- Зафиксируем один переход $\delta(q, a) = r$. Через δ' обозначим (частичную) функцию переходов на $Q \times \Sigma$ такую, что значение δ' не определено на паре (q, a) и совпадает со значением δ на всех остальных парах.
- Рассмотрим автоматы
$\mathcal{A}_0 = (Q, \Sigma, \delta', q_0, F)$,
$\mathcal{A}_1 = (Q, \Sigma, \delta', q_0, \{q\})$,
$\mathcal{A}_2 = (Q, \Sigma, \delta', r, \{q\})$,
$\mathcal{A}_3 = (Q, \Sigma, \delta', r, F)$,
каждый из которых имеет $(k-1)$ переход. По индукционному предположению языки $L(\mathcal{A}_0)$, $L(\mathcal{A}_1)$, $L(\mathcal{A}_2)$, $L(\mathcal{A}_3)$ регулярны.

Теорема Клини: окончание доказательства

- Для завершения доказательства достаточно показать, что
$$L(\mathcal{A}) = L(\mathcal{A}_0) \cup L(\mathcal{A}_1)\{a\}(L(\mathcal{A}_2)\{a\})^* L(\mathcal{A}_3). \qquad (\star)$$
- Пусть $w \in L(\mathcal{A})$. Если автомат $\mathcal{A}$ распознал w, не совершив перехода $\delta(q, a) = r$, то $w \in L(\mathcal{A}_0)$.
- Если же переход $\delta(q, a) = r$ был совершён $n \geqslant 1$ раз, то $w = w_0 a w_1 a \ldots w_{n-1} a w_n$ для некоторых $w_0 \in L(\mathcal{A}_1)$, $w_1, \ldots, w_{n-1} \in L(\mathcal{A}_2)$, $w_n \in L(\mathcal{A}_3)$:



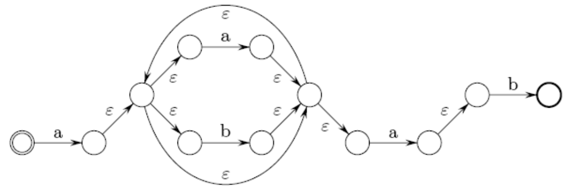
Так что $w \in L(\mathcal{A}_1)\{a\}(L(\mathcal{A}_2)\{a\})^* L(\mathcal{A}_3)$.
- Пусть w принадлежит правой части $(\star)$. Если $w \in L(\mathcal{A}_0)$, то по определению автомата $\mathcal{A}_0$ верно $w \in L(\mathcal{A})$.
- Если $w \in L(\mathcal{A}_1)\{a\}(L(\mathcal{A}_2)\{a\})^* L(\mathcal{A}_3)$, то $w = w_0 a w_1 a \ldots w_{n-1} a w_n$ для некоторых $n \geqslant 1$, $w_0 \in L(\mathcal{A}_1)$, $w_1, \ldots, w_{n-1} \in L(\mathcal{A}_2)$, $w_n \in L(\mathcal{A}_3)$. Отсюда $w \in L(\mathcal{A})$.

Определение регулярных выражений

- Регулярные языки можно задавать так называемыми *регулярными выражениями*.
- *Регулярные выражения* в алфавите (над алфавитом) Σ и языки, которые они обозначают, определяются следующим образом:
  - символы $\emptyset$, ε и a (где $\mathrm{a} \in \Sigma$) являются регулярными выражениями в Σ, обозначающими языки $\emptyset$, $\{\varepsilon\}$ и $\{\mathrm{a}\}$ соответственно;
  - если r и s — регулярные выражения в Σ, обозначающие языки R и S соответственно, то $(r|s)$, (rs) и r^* являются регулярными выражениями в Σ, обозначающими языки $R \cup S$, RS и R^* соответственно.
- Класс $\mathbb{R}$ совпадает с классом всех языков, определяемых регулярными выражениями.
- Переносим соглашение о приоритетах операций над языками.
- Пример: $\Sigma = \{\mathrm{a}, \mathrm{b}\}$,
  $\mathrm{a}^*\mathrm{b}^2|\mathrm{bb}^*\mathrm{a} = \{\mathrm{a}^m\mathrm{b}^2 \mid m \geqslant 0\} \cup \{\mathrm{b}^n\mathrm{a} \mid n \geqslant 1\}$,
  $(\mathrm{a}|\mathrm{b})^*\mathrm{a} = \{w\mathrm{a} \mid w \in \Sigma^*\}$.

Построение конечного автомата по регулярному выражению

Нормальный ε-НКА, распознающий язык $\mathrm{a}(\mathrm{a}|\mathrm{b})^*\mathrm{ab}$:



План

1. Недетерминированные конечные автоматы
2. Операции над автоматными языками
3. Регулярные языки
4. Автоматные (регулярные) фрагменты языков программирования
5. Пример нерегулярного языка

Регулярные выражения, задающие идентификаторы и числа

- Имя (или идентификатор):

$$\langle\text{имя}\rangle = \langle\text{буква}\rangle(\langle\text{буква}\rangle|\langle\text{цифра}\rangle)^*, \text{ где}$$
$$\langle\text{буква}\rangle = \mathrm{a}|\mathrm{b}|\ldots|\mathrm{z}\,,$$
$$\langle\text{цифра}\rangle = 0|1|\ldots|9$$

- Целое без знака:

$$\langle\text{целое\_без\_знака}\rangle = \langle\text{цифра}\rangle(\langle\text{цифра}\rangle)^*$$

- Число без знака:

$$\langle\text{число\_без\_знака}\rangle = \langle\text{целое\_без\_знака}\rangle(.\langle\text{целое\_без\_знака}\rangle|\varepsilon)$$
$$(\mathrm{E}(+|-|\varepsilon)\langle\text{целое\_без\_знака}\rangle|\varepsilon)$$

Регулярные выражения, задающие константы и простые типы

- Константа (в языке Паскаль):

$$\langle\text{константа}\rangle =$$
$$((+|-|\varepsilon)\langle\text{число\_без\_знака}\rangle|\langle\text{имя}\rangle) \mid {}'(\langle\text{буква}\rangle|\langle\text{цифра}\rangle|'')^*{}'$$

- Простой тип (в языке Паскаль):

$$\langle\text{простой\_тип}\rangle =$$
$$\langle\text{имя}\rangle \mid \text{"("}\langle\text{имя}\rangle(,\langle\text{имя}\rangle)^*\text{")"} \mid \langle\text{константа}\rangle..\langle\text{константа}\rangle$$

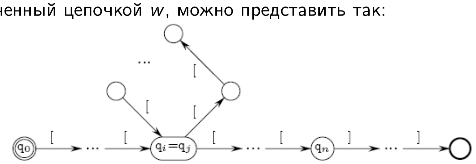
- Арифметические выражения (со сбалансированными скобками) не задаются регулярными выражениями, иначе говоря, не распознаются конечными автоматами.

План

1. Недетерминированные конечные автоматы
2. Операции над автоматными языками
3. Регулярные языки
4. Автоматные (регулярные) фрагменты языков программирования
5. Пример нерегулярного языка

Нерегулярность скобочного языка LB

- Предположим, что язык LB распознаётся некоторым ДКА $\mathcal{A}$.
- Пусть n — число состояний этого автомата. Автомат $\mathcal{A}$ распознаёт цепочку $w = \underbrace{[[\ldots[}_{n}\ \underbrace{]\ldots]]}_{n}$.
- Из начального состояния q_0 автомат $\mathcal{A}$ при чтении n скобок [последовательно переходит в состояния, обозначаемые $\mathrm{q}_1, \ldots, \mathrm{q}_n$.
- Найдутся такие $i < j$, что $\mathrm{q}_i = \mathrm{q}_j$. Путь в автомате $\mathcal{A}$, помеченный цепочкой w, можно представить так:



- Но тогда $\mathcal{A}$ распознаёт и цепочку $\underbrace{[[\ldots[}_{n-(j-i)}\ \underbrace{]\ldots]]}_{n}$, не принадлежащую LB. Противоречие.

Литература

Основная литература

- Замятин А. П., Шур А. М. Языки, грамматики, распознаватели: Учебное пособие. Екатеринбург : Изд-во Урал. ун-та, 2007 (электронный вариант книги — на `http://elar.usu.ru`, поиск)

Дополнительная литература

- Ахо А., Лам М., Сети Р., Ульман Дж. Компиляторы: принципы, технологии и инструментарий. М.: ООО "И.Д. Вильямс", 2008
- Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции. М.: Мир, 1978
- Мартыненко Б. К. Языки и трансляции: Учеб. пособие. СПб.: Издательство С.-Петербургского университета, 2004 (электронный вариант книги — на `http://www.math.spbu.ru/user/mbk`)