3 АВТОМАТНЫЕ ЯЗЫКИ

Напомним, что *языком* в алфавите Σ называется подмножество множества всех слов в этом алфавите. Язык называется *автоматным*, если он может быть задан некоторым конечным автоматом, другими словами, если существует конечный автомат, входной алфавит которого есть Σ и который воспринимает те и только те слова, которые принадлежат рассматриваемому языку.

Рассмотрим теперь языки в алфавите Σ. На множестве всех языков в этом алфавите рассматривают операции, которые порождают из заданных языков новые языки в том же алфавите.

Нам пока понадобится несколько операций.

Первая операция — теоретико-множественное *объединение*. Так как язык, по определению, является множеством слов, можно рассмотреть объединение двух языков как объединение двух множеств. Объединение языков L_1 и L_2 обозначается $L_1 \cup L_2$ и состоит из тех и только тех слов в алфавите Σ, которые входят либо в L_1, либо в L_2. Это записывается так:

$$L_1 \cup L_2 = \{w \in \Sigma^* \mid \quad w \in L_1 \ \text{ или } \ w \in L_2\}.$$

Аналогично мы вводим теоретико-множественные операции *разности* $L_1 \setminus L_2$ и *пересечения* $L_1 \cap L_2$:

$$L_1 \setminus L_2 = \{w \in \Sigma^* \mid \quad w \in L_1 \ \text{ и } \ w \notin L_2\};$$

$$L_1 \cap L_2 = \{w \in \Sigma^* \mid \quad w \in L_1 \ \text{ и } \ w \in L_2\}.$$

Частным случаем разности является *дополнение*. Именно:

$$\overline{L} = \{w \in \Sigma^* \mid \quad w \notin L\}.$$

Ясно, что $\overline{L} = \Sigma^* \setminus L$.

Теорема 3.1. *Операции объединения, пересечения и разности, примененные к автоматным языкам, дают снова автоматные языки. Дополнение автоматного языка является автоматным языком.*

Доказательство. Пусть для $i = 0, 1$

$$M_i =< \Sigma, Q_i, q_0^i, Q_{\text{закл}}^i, \Phi_i >$$

и конечный автомат M_i задаёт язык L_i.

Рассмотрим конечный автомат M:

$$M =< \Sigma,\ (Q_1 \times Q_2), (q_0^1, q_0^2), Q_{\text{закл}}, \Phi >,$$

состояниями которого являются пары состояний заданных конечных автоматов, начальным состоянием является пара, составленная из начальных состояний заданных автоматов, а правила составлены таким образом, что следующим состоянием нового автомата является пара следующих для заданных автоматов состояний:

$$\Phi((q', q''), a) = (\Phi_1(q', a), \Phi_2(q'', a)).$$

Очевидная индукция по длине слова показывает, что, получив на вход слово w, автомат перейдёт в состояние (q', q'') тогда и только тогда, когда, получив на вход это же слово w, первый из заданных автоматов перейдёт в состояние q', а второй перейдёт в состояние q''.

Действительно, если это верно для слова w и $\Phi_1(q', a) = s'$, а $\Phi_2(q'', a) = s''$, то $\Phi((q', q''), a) = (s', s'')$ и, прочитав слово wa, новый автомат перейдёт в состояние (s', s''), а старые автоматы перейдут при этом в состояния s' и s''.

Осталось задать заключительные состояния нового автомата. Если заключительными называть пары, одна из координат которых заключительна, то новый автомат задаёт объединение заданных множеств. Если заключительными называть пары, у которых обе координаты заключительны, то новый автомат задаёт пресечение заданных множеств. Если заключительными объявить пары, у которых первая координата заключительна, а вторая — незаключительна, то новый автомат задаёт разность заданных множеств. ∎

Прежде, чем переходить к изучению замкнутости автоматных языков относительно других операций над языками, введём понятие недетерминированного конечного автомата.

Определение 3.0.1 (недетерминированного конечного автомата). *Недетерминированным конечным автоматом (сокращенно, НКА) называется упорядоченная совокупность пяти объектов* $\langle \Sigma, Q, q_0, Q_{\text{закл}}, \Phi \rangle$*, где:*

- Σ *— конечный алфавит, в котором пишутся слова распознаваемого языка;*
- Q *— конечный алфавит состояний автомата;*
- $q_0 \in Q$ *— начальное состояние;*
- $Q_{\text{закл}}$ *— подмножество множества* Q*, элементы из* $Q_{\text{закл}}$ *— это так называемые заключительные состояния;*
- Φ *— конечное множество правил.*

Правилом называется тройка $\langle q, a, q' \rangle$*, составленная из состояния, обозреваемой входной буквы и нового состояния, следовательно, являющаяся элементом декартова произведения* $(Q \times \Sigma \times Q)$.

Итак, пусть на ленте недетерминированного конечного автомата $M = \langle \Sigma, Q, q_0, Q_{\text{закл}}, \Phi \rangle$ записано слово w из Σ^*. Головка автомата в начальный момент времени находится на самой левой букве слова w. Сам автомат находится в состоянии q_0.

Такт работы автомата M состоит в том, что он переходит в новое состояние, и затем сдвигает головку на одну букву вправо. Если сдвинуть головку уже нельзя, то автомат останавливается, в противном случае он делает следующий такт, и т.д.

Пусть головка автомата M обозревает в данный момент на ленте букву a из Σ, а сам автомат находится в состоянии q из Q. Тогда новое состояние на этом такте, q', выбирается произвольно с условием, что тройка $\langle q, a, q' \rangle$ принадлежит множеству Φ правил автомата M. В частности, может случиться так, что ни для какого q' тройка $\langle q, a, q' \rangle$ не принадлежит множеству Φ. Тогда нового состояния не существует, а автомат M не может прочитать входное слово.

Может быть и так, что для данных q, a существует несколько

$$q_1, \dots, q_k$$

таких, что

$$\langle q, a, q_1 \rangle, \dots, \langle q, a, q_k \rangle$$

все принадлежат Φ. В таком случае автомат M обладает свободой выбора конкретного q_i в качестве следующего состояния.

Легко видеть, что на слове w, которое автомат может прочитать, автомат M совершит ровно $|w|$ тактов. Состояние, в котором оказался автомат после последнего из этих тактов и на котором он остановился, будем называть последним состоянием.

Будем говорить, что автомат M воспринимает слово w в алфавите Σ, если при некотором способе работы последним состоянием M на слове w будет состояние, принадлежащее множеству $Q_{\text{закл}}$.

Более формально, *допустимой последовательностью состояний* автомата M для входного слова w называется одноэлементная последовательность q_0, если это входное слово пусто, и sq', если s является допустимой последовательностью состояний для w_1, s имеет вид $s'q$, w имеет вид w_1a и правило $< q, a, q' >$ входит в Φ. Таким образом, допустимая последовательность состояний начинается начальным состоянием, а каждое следующее состояние получается из предыдущего и обозреваемой буквы по одному из имеющихся для этого случая правил.

Слово w воспринимается недетерминированным автоматом M, если для этого слова существует такая допустимая последовательность состояний автомата M, которая заканчивается заключительным состоянием.

Приведём ещё более формальное определение. Пусть автомат M находится в состоянии q, обозревает букву a на входной ленте, и справа от его головки на ленте записано слово w.

Определение 3.0.2 (работы конечного недетерминированного автомата). *Если $q' \in Q$ такое, что $\langle q, a, q' \rangle \in \Phi$, то будем писать*

$$(q, aw) \mid\Rightarrow_M (q', w)$$

и

$$(q, aw) \mid\Rightarrow_M^1 (q', w).$$

Пусть

$$(q, aw'w) \mid\Rightarrow_M^{n+1} (q', w)$$

означает, что существует такое состояние q'' автомата M, что

$$(q, aw'w) \mid\Rightarrow_M (q'', w'w)$$

и

$$(q'', w'w) \mid\Rightarrow_M^n (q', w).$$

Отношение

$$(q, w'w) \mid\Rightarrow_M^+ (q', w)$$

означает, что

$$(q, w'w) \mid\Rightarrow_M^n (q', w)$$

для некоторого целого положительного n.

Автомат M воспринимает слово $w \in \Sigma^*$ тогда и только тогда, когда существует такое $q \in Q_{закл}$, что

$$(q_0, w) \mid\Rightarrow_M^+ (q, \lambda),$$

где λ — пустое слово. Автомат M воспринимает пустое слово тогда и только тогда, когда $q_0 \in Q_{закл}$.

Определение 3.0.3 (языка конечного недетерминированного автомата). *Языком конечного недетерминированного автомата $M = \langle \Sigma,\ Q,\ q_0,\ Q_{закл},\ \Phi \rangle$ называется множество всех слов в алфавите Σ, которые воспринимаются автоматом M. Язык автомата M мы будем обозначать через $L(M)$. Мы будем также говорить, что язык $L(M)$ распознается (или задается) автоматом M.*

Приведём пример.

Пример 3.0.1.

	0	1
начальное $q1$	$q8$	$q2$
$q2$	$q3$	$q8$
$q3$	$q8$	$q4$, $s0$
$q4$	$q6$	$q5$
$q5$	$q6$	$q5$
$q6$	$q7$	$q4$, $s0$
$q7$	$q7$	$q5$
$q8$	$q8$	$q8$
заключительное $s0$	$s1$	$s0$
$s1$	$s0$	$s1$

В этом примере рассматривается недетерминированный конечный автомат. Этот автомат недетерминированный, так как условие, чтобы для каждой пары $< q, a >$ из $(Q \times \Sigma)$ имелось одно и только одно такое состояние q', что тройка $< q, a, q' >$ является правилом рассматриваемого конечного автомата, не выполняется.

Например, для слова "10100101"получаем следующие последовательности состояний:

	1	0	1	0	0	1	0	1
$q1$	$q2$	$q3$	$s0$	$s1$	$s0$	$s0$	$s1$	$s1$
$q1$	$q2$	$q3$	$q4$	$q6$	$q7$	$q5$	$q6$	$s0$
$q1$	$q2$	$q3$	$q4$	$q6$	$q7$	$q5$	$q6$	$q4$

Из этого примера видно, что для одного и того же слова одни допустимые последовательности состояний заканчиваются заключительным состоянием, а другие заканчиваются незаключительным состоянием.

Подчеркнём ещё раз, что для того, чтобы слово воспринималось требуется, чтобы хотя бы одна допустимая последовательность состояний для этого слова заканчивалась заключительным состоянием. При этом некоторые другие допустимые последовательности состояний могут заканчиваться и незаключительными состояниями. По этой причине слово 10100101 воспринимается новым недетерминированным автоматом.

Покажем, что этот автомат воспринимает те и только те слова, которые являются конкатенацией слова, начинающегося и оканчивающегося на 101, со словом с чётным числом нулей. В самом деле, сначала автомат работает как автомат из примера 2.0.1. Если этот предыдущий автомат не приходит в заключительное состояние, новый автомат тоже не приходит. Если же новый автомат приходит в состояние $s0$, то предыдущий автомат должен был воспринять слово. Значит, слово начинается и оканчивается на 101. После этого новый автомат проверяет число нулей на чётность.

Важнейшим свойством конечных автоматов является возможность устранения недетерминированности.

Теорема 3.2. *Пусть* $M = \langle \Sigma, Q, q_0, Q_{\text{закл}}, \Phi \rangle$ *— недетерминированный конечный автомат. Существует детерминированный конечный автомат* $M' = \langle \Sigma, Q', q'_0, Q'_{\text{закл}}, \Psi \rangle$*, распознающий тот же язык.*

Доказательство. Рассмотрим автомат M. Нам необходимо построить детерминированный конечный автомат M', задающий тот же язык.

В качестве множества Q' состояний M' рассмотрим множество

$$\{T \mid \ T \subseteq Q\}.$$

Таким образом, для каждого подмножества множества Q у нас в Q' есть свое собственное состояние.

В качестве начального состояния q'_0 автомата M' рассмотрим состояние $\{q_0\}$.

В автомате M' мы хотим как бы промоделировать совместное исполнение всех возможных вариантов работы автомата M. Начнем с содержательных пояснений. В начальный момент времени автомат M находится в состоянии q_0 и обозревает первую букву слова, скажем, букву a_0. Поскольку M — недетерминированный автомат, он может сработать в этих условиях по разному. Пусть

$$\langle q_0, a_0, q_{i_1}\rangle, \langle q_0, a_0, q_{i_2}\rangle, \ldots, \langle q_0, a_0, q_{i_r}\rangle$$

будут все элементы Φ вида $\langle q_0, a_0, q\rangle$. Тогда M может на этом такте перейти в одно из состояний $q_{i_1}, q_{i_2}, \ldots, q_{i_r}$.

Этот первый такт работы мы смоделируем в M', включив в Ψ одно правило:

$$\langle \{q_0\}, a_0, \{q_{i_1}, q_{i_2}, \ldots, q_{i_r}\}\rangle. \qquad (1)$$

По определению, $\{q_{i_1}, q_{i_2}, \ldots, q_{i_r}\}$ является состоянием автомата M'. Если же выше $r = 0$, то есть ни для какого $q \in Q$ тройка $\langle q_0, a_0, q\rangle$ не принадлежит Φ, то в Ψ вместо (1) включим правило $\langle \{q_0\}, a_0, \emptyset\rangle$. Фактически, находясь в состоянии $\{q_{i_1}, q_{i_2}, \ldots, q_{i_r}\}$, мы понимаем, что M' мог бы находиться в одном из состояний $q_{i_1}, q_{i_2}, \ldots, q_{i_r}$.

Пусть, в некоторый момент времени, автомат M' обработал буквы $a_0, a_1, \ldots, a_{m-1}$ входного слова, при этом M' находится в состоянии

$$\{q_{i_1}, q_{i_2}, \ldots, q_{i_r}\}$$

и обозревает букву a_m входного слова. Ясно из построения, что автомат M, обработав буквы $a_0, a_1, ...a_{m-1}$ входного слова, мог бы находится в любом из состояний $q_{i_1}, q_{i_2}, \ldots, q_{i_r}$.

Поочередно рассмотрим все эти возможности. Если M находится в состоянии q_{i_n} , то, обозревая букву a_m, он может перейти в одно из состояний $q_{k_1}, q_{k_2}, \ldots, q_{k_s} \in Q$, где

$$\langle q_{i_n}, a_m, q_{k_1}\rangle, \langle q_{i_n}, a_m, q_{k_2}\rangle, \ldots, \langle q_{i_n}, a_m, q_{k_s}\rangle$$

представляют все правила из Φ вида $\langle q_{i_n}, a_m, q\rangle$. Если же в Φ нет правил вида $\langle q_{i_n}, a_m, q\rangle$, то M не может перейти в следующее состояние. Таким образом, единственное правило из Ψ для состояния $\{q_{i_1}, q_{i_2}, \ldots, q_{i_r}\}$ и буквы a_m должно учитывать все эти возможности для всех входящих в него состояний первоначального автомата, то есть иметь вид:

$$\langle \{q_{i_1}, q_{i_2}, \ldots, q_{i_r}\}, a_m, X\rangle,$$

где X содержит все такие q, что $\langle q_{i_j}, a_m, q\rangle \in \Phi$ для некоторого $j \in \{1, \ldots, r\}$, и не содержит никаких других элементов.

Формально, построим множество Ψ следующим образом.

Пусть a — буква из Σ и $T = \{q_1, \ldots, q_n\}$ — некоторое непустое подмножество Q. Пусть T' содержит все такие q, что $\langle q_j, a, q\rangle \in \Phi$ для некоторого $j \in \{1, \ldots, n\}$, и не содержит никаких других элементов. Тогда включим тройку $\langle T, a, T'\rangle$ в Ψ. Кроме того, включим тройку $< \emptyset, a, \emptyset >$ в Ψ.

Теперь положим

$$Q'_{\text{закл}} = \{T \in Q' \mid T \cap Q_{\text{закл}} \neq \emptyset\}.$$

Докажем утверждение теоремы формально.

1. Пусть $w = a_m a_{m+1} ... \ a_k \in \Sigma^*$ — некоторое слово, $T \subseteq Q$, $q \in T$. Индукцией по длине слова w покажем, что если

$$(q, w) \mid\Rightarrow^+_M (q', \lambda),$$

то для некоторого $T' \subseteq Q$ такого, что $q' \in T'$ имеет место

$$(T, w) \mid\Rightarrow^+_{M'} (T', \lambda),$$

где λ — пустое слово.

Если $m = k$, то есть $w = a_m$, то q' таково, что $\langle q, a_m, q'\rangle \in \Phi$. Ясно, что если рассмотреть то единственное T', что $\langle T, a_m, T'\rangle \in \Psi$, то $q' \in T'$. Тогда $(T, a_m) \mid\Rightarrow_{M'} (T', \lambda)$, то есть $(T, w) \mid\Rightarrow^+_{M'} (T', \lambda)$, и утверждение доказано.

Пусть $m < k$. Так как $(q, w) \mid\Rightarrow^+_M (q', \lambda)$, то существует в Q такое состояние q_{m+1}, что

$$(q, w) \mid\Rightarrow_M (q_{m+1}, a_{m+1} a_{m+2} ... a_k),$$

$$(q_{m+1}, a_{m+1} a_{m+2} ... a_k) \mid\Rightarrow^+_M (q', \lambda).$$

Рассуждая как и в предыдущем случае, можно предполагать, что существует $T'' \subseteq Q$ такое, что $q_{m+1} \in T''$ и

$$(T, w) \mid\Rightarrow_{M'} (T'', a_{m+1} a_{m+2} ... a_k).$$

По индукции, существуют $T' \subseteq Q$ и $q' \in T'$, такие что

$$(T'', a_{m+1}...a_k) \mid\Rightarrow^+_{M'} (T', \lambda).$$

Но тогда

$$(T, w) \mid\Rightarrow^+_{M'} (T', \lambda).$$

Утверждение доказано.

В частности, для произвольного $w \in L(M)$ существует такое заключительное состояние $q' \in Q_{\text{закл}}$, что $(q_0, w) \mid\Rightarrow^+_M (q', \lambda)$. Тогда, как доказано, существует такое $T' \subseteq Q$, что $(\{q_0\}, w) \mid\Rightarrow^+_{M'} (T', \lambda)$ и $q' \in T'$. Так как $q' \in Q_{\text{закл}}$, то $T' \in Q'_{\text{закл}}$, по определению. Значит, $w \in L(M')$.

Итак, $L(M) \subseteq L(M')$.

2. Теперь докажем обратное, то есть что $L(M') \subseteq L(M)$.

Пусть $w = a_m a_{m+1}...a_k \in \Sigma^*$ — некоторое слово, $T, T' \subseteq Q$, $q' \in T'$. Пусть

$$(T, w) \mid\Rightarrow^+_{M'} (T', \lambda).$$

Докажем, что существует такое $q \in T$, что

$$(q, w) \mid\Rightarrow^+_M (q', \lambda).$$

Пусть $m = k$. Тогда $(T, a_m) \mid\Rightarrow_{M'} (T', \lambda)$ и по определению, существует такое $q \in T$, что $\langle q, a_m, q' \rangle \in \Phi$.

Ясно, что тогда существует $q \in T$, что $(q, a_m) \mid\Rightarrow_M (q', \lambda)$, то есть $(q, w) \mid\Rightarrow_M (q', \lambda)$, и утверждение доказано.

Пусть $k > m$. Так как

$$(T, w) \mid\Rightarrow^+_{M'} (T', \lambda),$$

то

$$(T, w) \mid\Rightarrow^+_{M'} (T'', a_k),$$

$$(T'', a_k) \mid\Rightarrow_{M'} (T', \lambda)$$

для некоторого $T'' \subseteq Q$. Но тогда

$$(T, a_m \dots a_{k-1}) \mid\Rightarrow^+_{M'} (T'', \lambda).$$

По индукции, существует $q'' \in T''$ такое, что

$$(q'', a_k) \mid\Rightarrow_M (q', \lambda).$$

По индукции также существует такое $q \in T$, что

$$(q, a_m \dots a_{k-1}) \mid\Rightarrow_M^+ (q'', \lambda).$$

Но это значит, что

$$(q, a_m \dots a_k) \mid\Rightarrow_M^+ (q'', a_k).$$

Сопоставив, получим, что

$$(q, a_m \dots a_k) \mid\Rightarrow_M^+ (q', \lambda),$$

чем доказано утверждение.

В частности, пусть $w \in L(M')$. Тогда существует такое $T \in Q'_{\text{закл}}$, что

$$(\{q_0\}, w) \mid\Rightarrow_{M'}^+ (T, \lambda).$$

Так как $T \in Q'_{\text{закл}}$, то по определению найдется такое $q' \in Q_{\text{закл}}$, что $q' \in T$. Но тогда, по доказанному, существует $q \in \{q_0\}$ такое, что $(q, w) \mid\Rightarrow_M^+ (q', \lambda)$, то есть $(q_0, w) \mid\Rightarrow_M^+ (q', \lambda)$, и $q' \in Q_{\text{закл}}$. Значит, $w \in L(M)$.

Итак, $L(M') \subseteq L(M)$. ∎

Введенная ранее операция конкатенации на словах дополнительно индуцирует операции и на языках.

Под конкатенацией языков L и K, обозначаемой LK, мы понимаем множество слов

$$\{vw \mid \ v \in L \ \text{ и } \ w \in K\}.$$

Под i-ой степенью языка L, обозначаемой L^i, мы понимаем следующее определенное по индукции множество:

$$L^0 = \{\lambda\}, \text{ где } \lambda \text{ — пустое слово;} \quad L^{i+1} = L^i L.$$

Наконец, *итерацией* языка L, обозначаемой L^*, мы будем называть бесконечное объединение натуральных степеней языка L:

$$L^* = \{w \mid \quad \text{существует такое } i \geqslant 0, \text{ что } w \in L^i\}.$$

Например, если $L = \{00,\ 11\}$ и $K = \{101,\ 010\}$, то

$$LK = \{00101,\ 00010,\ 11101,\ 11010\}, \quad L^3 = \{000000,\ 000011,\\ 001100,\ 001111,\ 110000,\ 110011,\ 111100,\ 111111\},$$

а L^* состоит из пустого слова и слов чётной длины, которые начинаются на 00 или на 11 и в которых после этого начала следует слово снова из L^*. Следует заметить, что итерация пустого языка или языка, содержащего только пустое слово, представляет собой язык, содержащий только пустое слово. В остальных случаях итерация любого языка представляет собой бесконечный язык.

До сих пор мы не накладывали на языки никаких ограничений, считая их произвольными множествами слов. Для изучения языков, однако, важно по крайней мере уметь как-то задавать эти языки. Если язык конечен, то его можно задать путем перечисления всех его элементов. Однако этот способ не годится ни для какого бесконечного языка.

Способ, который мы сейчас собираемся рассмотреть, — задание языков при помощи операций над конечными языками. Выше мы видели, что операция итерации языка позволяет получить из конечного языка бесконечный.

Определение 3.0.4 (регулярности языка). *Язык называется регулярным, если его можно получить из конечного числа конечных языков применением конечного числа операций объединения, конкатенации и итерации языков. Регулярные языки часто называют регулярными множествами.*

Рассмотрим теперь понятие регулярного выражения над алфавитом Σ.

Определение 3.0.5 (регулярного выражения). *Пусть Σ — алфавит. Регулярное выражение над алфавитом Σ определяется по индукции. Одновременно по индукции мы будем каждому регулярному выражению сопоставлять регулярный язык в рассматриваемом алфавите.*

1. *$\emptyset$ — регулярное выражение; ему соответствует пустой регулярный язык.*

2. *λ — регулярное выражение; ему соответствует одноэлементный регулярный язык, состоящий из одного пустого слова.*

3. *Если $a \in \Sigma$, то a — регулярное выражение; ему соответствует регулярный язык $\{a\}$.*

4. *Если $r,\ r'$ — регулярные выражения, которым соответствуют регулярные языки $R,\ R'$, то $(rr'),\ (r+r'),\ r^*$ — регулярные выражения; им соответствуют регулярные языки $RR',\ R \cup R',\ R^*$.*

По определению, всякому регулярному выражению r соответствует некоторый регулярный язык R. Мы будем говорить, что регулярное выражение r задает этот регулярный язык R и писать $L(r) = R$. Имеет место очевидная

Теорема 3.3. *Всякий регулярный язык в алфавите Σ задается некоторым регулярным выражением в алфавите Σ.*

Доказательство. Следует проводить по индукции. Очевидно, что всякое множество слов, состоящее из одного слова, может быть задано регулярным выражением с использованием только пунктов 1, 2, 3 определения 3.0.5 и операции конкатенации. Поэтому всякое конечное множество слов может быть задано регулярным выражением с использованием только пунктов 1, 2, 3 определения 3.0.5 и операций конкатенации и + (объединения). Этим доказан базис индукции. Индукционный шаг также очевиден. ∎

Легко проверить, что регулярная операция + коммутативна и ассоциативна, а операция конкатенации ассоциативна и дистрибутивна по отношению к операции +.

Приведем примеры.

Пример 3.0.2.

Рассмотрим алфавит $\{0, 1\}$. Уже были приведены примеры простых регулярных языков над этим алфавитом. Рассмотрим язык L в этом алфавите, состоящий из всех слов ровно с двумя буквами 0. Он регулярен. Чтобы показать это, запишем задающее его регулярное выражение: 1*01*01*.

Пример 3.0.3.

В том же алфавите множество слов четной длины задается следующим регулярным выражением: $(01 + 10 + 11 + 00)^*$.

Теорема 3.4. *Пусть r — регулярное выражение в алфавите Σ, $R = L(r)$. Существует такой НКА*

$$M = \langle \Sigma, Q, q_0, Q_{\textit{закл}}, \Phi \rangle,$$

что $R = L(M)$.

Доказательство. Мы явно построим M индукцией по сложности построения r.

Базис.

1. Если r есть регулярное выражение $\emptyset$, положим $Q = \{q_0\}$, $Q_{\text{закл}} = \Phi = \emptyset$.

Так как заключительных состояний нет, никакое слово этим автоматом не воспринимается.

2. Если r есть регулярное выражение λ, положим

$$Q = \{q_0, q_1\},\ Q_{\text{закл}} = \{q_0\},\ \Phi = \{\langle q_0, a, q_1 \rangle \mid a \in \Sigma\}.$$

Этот автомат воспринимает только пустое слово, поскольку, считав любую букву, автомат переходит в незаключительное состояние, в котором и остаётся.

3. Если r есть регулярное выражение $a \in \Sigma$, положим

$$Q = \{q_0, q_1, q_2\},\ Q_{\text{закл}} = \{q_1\},$$
$$\Phi = \{\langle q_0, a, q_1\rangle\} \cup \{\langle q_0, b, q_2\rangle \mid\ b \in \Sigma,\ b \neq a\}$$
$$\cup \{\langle q_1, b, q_2\rangle \mid\ b \in \Sigma\}.$$

Этот автомат воспринимает только однобуквенное слово, состоящее из заданной буквы. Считав следующую букву, он переходит в незаключительное состояние, в котором и остаётся. Если же автомат сразу считывает другую букву, он переходит в это же незаключительное состояние.

Шаг индукции. Пусть r', r'' — регулярные выражения и пусть по индукции им уже сопоставлены такие недетерминированные конечные автоматы

$$M' = \langle \Sigma, Q', q_0', Q'_{\text{закл}}, \Phi'\rangle,\ M'' = \langle \Sigma, Q'', q_0'', Q''_{\text{закл}}, \Phi''\rangle,$$

что $L(M') = L(r')$ и $L(M'') = L(r'')$.

Считаем, что $Q' \cap Q'' = \emptyset$.

Если это не так, можно переобозначить состояния второго автомата. Более подробно, в качестве множества состояний нового второго автомата выбирем такое множество, элементы которого не являются состояниями первого автомата и для которого существует такое отображение τ множества состояний старого второго автомата на это множество, что каждому старому состоянию соответствует одно и только одно новое, а каждое новое состояние соответствует какому-то старому. Начальным состоянием объявим образ старого начального состояния, заключительными объявим все образы заключительных состояний и только эти образы. Каждое правило $< q, a, q' >$ старого второго автомата заменим на правило $< \tau(q), a, \tau(q') >$. Других правил новый второй автомат не содержит. Новый второй автомат обзначим через $\tau(M'')$. Ясно, что

$$(q, aw) \models^1_{M''} (q', w)$$

тогда и только тогда, когда

$$(\tau(q), aw) \mid\Rightarrow^1_{\tau(M'')} (\tau(q'), w).$$

Поэтому новый второй автомат задаёт тот же язык, который задаётся старым вторым автоматом.

Без потери общности можно считать, что в M' для всякого состояния $q' \in Q'$ и всякого $a \in \Sigma$ существует $q \in Q'$, что $\langle q', a, q\rangle \in \Phi'$. Того же будем требовать и от автомата M''. Грубо говоря, во всех пустых клетках таблицы, задающей правила автомата, можно написать некоторое новое состояние, а в клетках для этого нового состояния написать его же. Теперь пустых клеток не окажется, но новый автомат будет задавать тот же язык.

Без потери общности можно также считать, что первый автомат M' не имеет правил вида $< q, a, q'_0 >$. Если бы это было не так, мы могли бы добавить новое состояние s к состояниям первого автомата, объявить это новое состояние начальным и для каждого правила $< q'_0, a, q >$ первого автомата добавить правило $< s, a, q >$. Ясно, что преобразованный таким образом первый автомат тоже задаёт язык $L(r')$.

1. Пусть r есть регулярное выражение $(r'r'')$. Положим

$$M = \langle \Sigma, Q, q'_0, Q_{\text{закл}}, \Phi\rangle,$$

где $Q = Q' \cup Q''$, $Q_{\text{закл}} = Q''_{\text{закл}}$,

$$\Phi = \Phi' \cup \Phi'' \cup \{\langle q, a, q''_0\rangle \mid\ q \in Q',\ a \in \Sigma,\ q' \in Q'_{\text{закл}},\ \langle q, a, q'\rangle \in \Phi'\}.$$

Если $q''_0 \in Q''_{\text{закл}}$, добавим еще в $Q_{\text{закл}}$ все состояния из $Q'_{\text{закл}}$, то есть в этом случае $Q_{\text{закл}} = Q'_{\text{закл}} \cup Q''_{\text{закл}}$.

Если же $q'_0 \in Q'_{\text{закл}}$, добавим еще в Φ правила $< q'_0, a, q >$ во всех случаях, когда $< q''_0, a, q >$ лежит в Φ''.

Другими словами, автомат M представляет собой объединение автоматов M' и M'' и работает так. Сначала он работает как автомат M', с тем добавлением, что вместо перехода в любое из заключительных состояний автомата M', может перейти в начальное состояние автомата M''. Перейдя в это состояние, он далее работает как автомат M''. Если начальное состояние первого автомата является заключительным, результирующий автомат может сразу работать как второй автомат, а если начальное состояние второго автомата является заключительным, результирующий автомат воспринимает все слова, воспринимаемые первым автоматом.

Пустое слово воспринимается новым автоматом тогда и только тогда, когда оно воспринимается как первым, так и вторым автоматом.

Если второй язык содержит пустое слово, конкатенация заданных регулярных языков содержит первый язык. Но в этом случае, построенный автомат воспринимает все слова из первого языка.

Если первый язык содержит пустое слово, конкатенация заданных регулярных языков содержит второй язык. Но в этом случае, построенный автомат воспринимает все слова из второго языка.

Пусть теперь некоторое непустое слово воспринимается построенным автоматом. Если в допустимой последовательности состояний построенного автомата для этого слова не встречаются состояния второго автомата, то последнее состояние является заключительным состоянием первого автомата и слово воспринимается первым автоматом, а второй язык содержит пустое слово. Первое состояние этой допустимой последовательности, которое является состоянием второго автомата, может быть только начальным состоянием второго автомата и может быть написано только вместо заключительного состояния первого автомата. Поэтому рассматриваемое слово является конкатенацией слова из первого языка со словом из второго языка.

Обратно, рассмотрим конкатенацию непустого слова из первого языка с непустым словом из второго языка. Построим допустимую последовательность состояний построенного автомата M для этой конкатенации двух слов. Сначала берём допустимую последовательность состояний автомата M' для первого из этих слов, заканчивающуюся заключительным состоянием автомата M', в которой последнее состояние заменяем на начальное состояние второго автомата. Затем берем заканчивающуюся заключительным состоянием автомата M'' допустимую последовательность состояний автомата M'' для второго слова рассматриваемой конкатенации, выбросив из неё самое первое состояние. Получаем, что рассматриваемая конкатенация воспринимается построенным автоматом M.

2. Пусть r есть регулярное множество $(r' + r'')$. Положим

$$M = \langle \Sigma, Q, q_0, Q_{\text{закл}}, \Phi \rangle,$$

где

$$Q = Q' \cup Q'' \cup \{q_0\},$$
$$q_0 \notin (Q' \cup Q''),$$
$$Q_{\text{закл}} = Q'_{\text{закл}} \cup Q''_{\text{закл}},$$
$$\Phi = \Phi' \cup \Phi'' \cup \{\langle q_0, a, q\rangle \mid \ a \in \Sigma, (\langle q_0', a, q\rangle \in \Phi' \text{ или } \langle q_0'', a, q\rangle \in \Phi'')\}.$$

Если $q_0' \in Q'_{\text{закл}}$ или $q_0'' \in Q''_{\text{закл}}$, то дополнительно включим q_0 в $Q_{\text{закл}}$ (для учета пустого слова).

Другими словами, из начального состояния q_0 автомата M разрешены в точности все переходы, разрешенные из начального состояния q_0' автомата M' и из начального состояния q_0'' автомата M''.

3. Пусть r есть регулярное множество $(r')^*$. Положим

$$M = \langle \Sigma, Q, q_0, Q_{\text{закл}}, \Phi \rangle,$$

где

$$Q = Q' \cup \{q_0\},$$
$$q_0 \notin Q',$$
$$Q_{\text{закл}} = Q'_{\text{закл}} \cup \{q_0\},$$
$$\Phi = \Phi' \cup \{\langle q, a, q_0\rangle \mid \ q_1 \in Q'_{\text{закл}}, \ a \in \Sigma, \ \langle q, a, q_1\rangle \in \Phi'\} \cup$$
$$\{\langle q_0, a, q\rangle \mid \ a \in \Sigma, \ q \in Q', \ \langle q_0', a, q\rangle \in \Phi'\}.$$

Построение здесь аналогично случаю конкатенации, только произведенному с самим собой, что обеспечивает "зацикливание". Новое состояние q_0 введено для обеспечения распознавания пустого слова.

Индукцией по i доказываем, что любое слово из $(L(r'))^i$ воспринимается автоматом M. Слово из $(L(r'))^{i+1}$ является конкатенацией слова из $(L(r'))^i$ и слова из $L(r')$. Возьмём допустимую последовательность состояний автомата M для первого слова, в которой последнее состояние заменено на q_0, и дополним её допустимой для второго слова последовательностью состояний автомата M', выбросив из неё самое первое состояние.

Наоборот, пусть непустое слово воспринимается автоматом M. Индукцией по длине этого слова докажем, что это слово принадлежит $(L(r'))^*$. В допустимой последовательности состояний автомата M для этого слова найдём первое вхождение q_0 после начального вхождения q_0. Если такового нет, слово входит в $L(r')$. Иначе слово является конкатенацией слова из $L(r')$ и слова, воспринимаемого автоматом M.

Теперь очевидно, что для так построенного автомата M будет иметь место равенство $L(M) = L(r)$. ∎

Итак, мы показали, что всякое регулярное множество может быть распознано конечным автоматом.

Оказывается, что и наоборот, всякое множество, которое может быть распознано конечным автоматом, является регулярным. Однако доказательство этой теоремы мы не приводим.

Следствие 3.5. *Каждый регулярный язык может быть задан некоторым детерминированным конечным автоматом.*