§1 Основные понятия теории автоматов

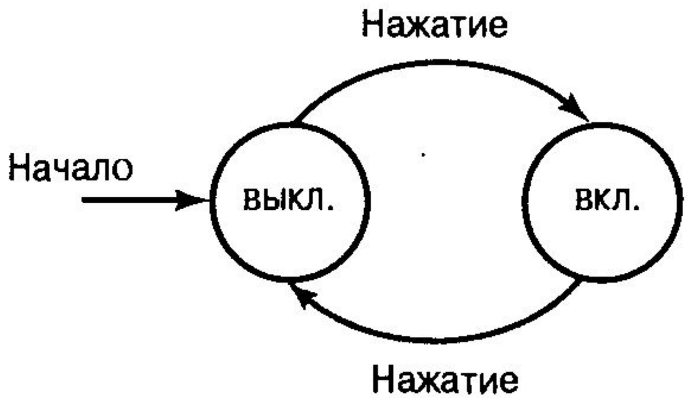
Теория автоматов занимается изучением абстрактных вычислительных устройств или «машин».

Конечные автоматы являются моделью для многих компонентов аппаратного и программного обеспечения.

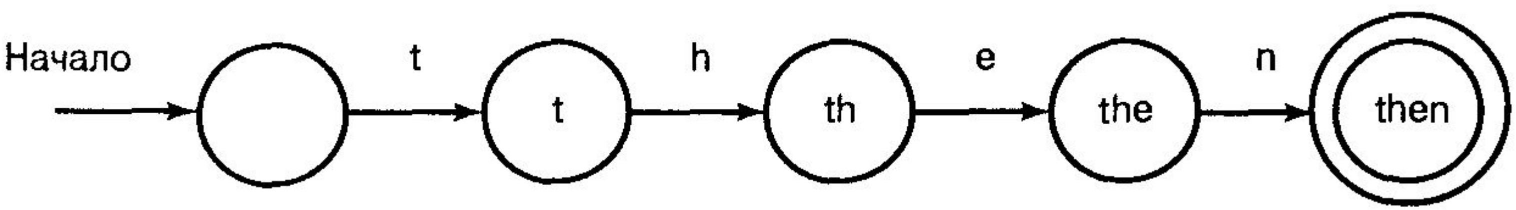
Примеры использования КА:

- Программное обеспечение, используемое для разработки и проверки цифровых схем.
- «Лексический анализатор» стандартного компилятора, т.е. тот компонент компилятора, который отвечает за разбивку исходного текста на такие логические единицы, как идентификаторы, ключевые слова и знаки пунктуации.
- Программное обеспечение для сканирования больших текстовых массивов с целью поиска заданных последовательностей символов.
- Программное обеспечение для проверки различного рода систем (протоколы связи и т.д.), которые могут находиться в конечном числе различных состояний.

ПРИМЕР 1: Конечный автомат, моделирующий переключатель



ПРИМЕР 2: Конечный автомат, который может служить частью лексического анализатора.



Алфавитом называется конечное непустое множество символов.

Σ={0,1} – бинарный алфавит;

Цепочка в алфавите Σ – это конечная последовательность символов из Σ.

01101 – цепочка в бинарном алфавите;

Пустая цепочка ε - это цепочка, не содержащая символов. Можно рассматривать, как цепочку в любом алфавите.

Длиной цепочки *x* называется число различных вхождений символов в x.

Если $\Sigma = \{0, 1\}$, то $\Sigma^1 = \{0, 1\}$, $\Sigma^2 = \{00, 01, 10, 11\}$

Степень алфавита – определим Σ^k как множество всех цепочек длины k, состоящих из символов алфавита Σ.

$\Sigma^0=\{\varepsilon\}$ независимо от алфавита;

Множество всех цепочек над алфавитом Σ принято обозначать Σ^*. Так, например, $\{0,1\}^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, \ldots\}$. По-другому это множество можно записать в виде

$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots$$

Иногда нам будет необходимо исключать из множества цепочек пустую цепочку. Множество всех непустых цепочек в алфавите Σ обозначают через Σ^+. Таким образом, имеют место следующие равенства:

- $\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \ldots$
- $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$.

<u>Конкатенацией</u> цепочек x и y называется цепочка *xy*.
Если цепочка имеет вид *xyz*, то *x*-префикс, *y*-подцепочка, *z*-суффикс.

Пусть I – алфавит. Регулярные выражения над I и языки, представляемые ими, рекурсивно определяются следующим образом:

1) ∅ является регулярным выражением и представляет пустое множество.
2) ε является регулярным выражением и представляет множество {ε}.
3) Для каждого a∈I символ a является регулярным выражением и представляет множество {a}.
4) Если p и q – регулярные выражения, представляющие множества P и Q соответственно, то (p+q), (pq) и (p*) являются регулярными выражениями и представляют множества , и P* соответственно.

P Q P Q

ПРИМЕР 1:

- 01 – регулярное выражение, представляющее множество {01}.
- (0+1)* представляет {0,1}*.
- 1(0+1)*1+1 представляет множество всех цепочек, начинающихся и заканчивающихся символом 1.

Язык называется *регулярным*, если его можно представить регулярным выражением.

Два регулярных выражения α и β называют *эквивалентными* (и пишут α=β), если они представляют одно и то же множество.

Например, (0+1)*=(0*1*)*.

Языком над алфавитом Σ называют произвольное множество цепочек в Σ.

Пусть L_1 и L_2 – два языка. Язык L_1L_2, называемый *конкатенацией* L_1 и L_2, есть множество

$$\{xy \mid x \in L_1 \ \text{ и } \ y \in L_2\}$$

Пусть L – язык. Тогда положим $L^0 = \{\varepsilon\}$ и $L^i = L \cdot L^{i-1}$ при $i \geq 1$. ***Замыканием Клини*** (*итерацией*) L* языка L называют язык

$$L^* = \bigcup_{i=0}^{\infty} L^i$$

а *позитивной итерацией* L^+ - язык

$$L^+ = \bigcup_{i=1}^{\infty} L^i$$

ПРИМЕРЫ ЯЗЫКОВ:

1. Язык, состоящий из всех цепочек, в которых n единиц следуют за n нулями для некоторого $n \geq 0$: $\{\varepsilon, 01, 0011, 000111, \ldots\}$.
2. Множество цепочек, состоящих из 0 и 1 и содержащих поровну тех и других: $\{\varepsilon, 01, 10, 0011, 1001, \ldots\}$.
3. Множество двоичных записей простых чисел: $\{10, 11, 101, 111, 1011, \ldots\}$.
4. Σ^* — язык для любого алфавита Σ.
5. $\varnothing$ — пустой язык в любом алфавите.
6. $\{\varepsilon\}$ — язык, содержащий одну лишь пустую цепочку. Он также является языком в любом алфавите. Заметим, что $\varnothing \neq \{\varepsilon\}$; первый не содержит вообще никаких цепочек, а второй состоит из одной цепочки.

Единственное существенное ограничение для множеств, которые могут быть языками, состоит в том, что все алфавиты конечны. Таким образом, хотя языки и могут содержать бесконечное число цепочек, но эти цепочки должны быть составлены из символов некоторого фиксированного конечного алфавита.

Язык представим регулярным выражением тогда и только тогда, когда он допускается некоторым конечным автоматом.

Недетерминированным конечным автоматом (НКА) называется пятёрка (S,I,δ,s_0,F), где:

- S – конечное *множество состояний* устройства управления;
- I – *алфавит* входных символов;
- δ - функция переходов, отображающая $S \times (I \cup \{\varepsilon\})$ множество подмножеств множества S;
- $s_0 \in S$ - начальное состояние устройства управления;
- $F \subseteq S$ - множество заключительных (допускающих состояний).

Мгновенным описанием (МО) НКА М называется пара (s,w), где $s \in S$ – состояние блока управления и $w \in I^*$ - неиспользованная часть входной цепочки (т.е. символ, обозреваемый входной головкой, и все символы справа от него).

Начальным МО автомата М называется МО, у которого первой компонентой служит начальное состояние, а второй – распознаваемая цепочка, т.е. (s_0,w) для некоторой цепочки $w \in I^*$.

Допускающие МО – это МО вида (s,ε), где $s \in F$.

Представим шаги НКА бинарным отношением $\vdash$ на множестве мгновенных описаний. Если $\delta(s,a)$ содержит s', то пишут $(s,aw) \vdash (s',w)$ для всех $w \in I^*$(где a – это или ε, или символ из I).

Если $a=\varepsilon$, то состояние изменяется независимо от обозреваемого входного символа. Если $a \neq \varepsilon$, то символ a должен стоять в очередной клетке входной ленты, а входная головка сдвигается на одну клетку вправо.

Рефлексивное и транзитивное замыкание отношения $\vdash$ обозначается через $\vdash^*$.

Говорят, что цепочка w *допускается автоматом* M, если $(s_0,w) \vdash^* (s,\varepsilon)$ для некоторого $s \in F$. Иными словами, входная цепочка w допускается автоматом M, если найдётся последовательность из нуля или более шагов, переводящая M из начального МО (s_0,w) в допускающее МО (s,ε).

Множество цепочек L(M), допускаемых автоматом M, называют *языком, допускаемым автоматом* M.

Пример 2:

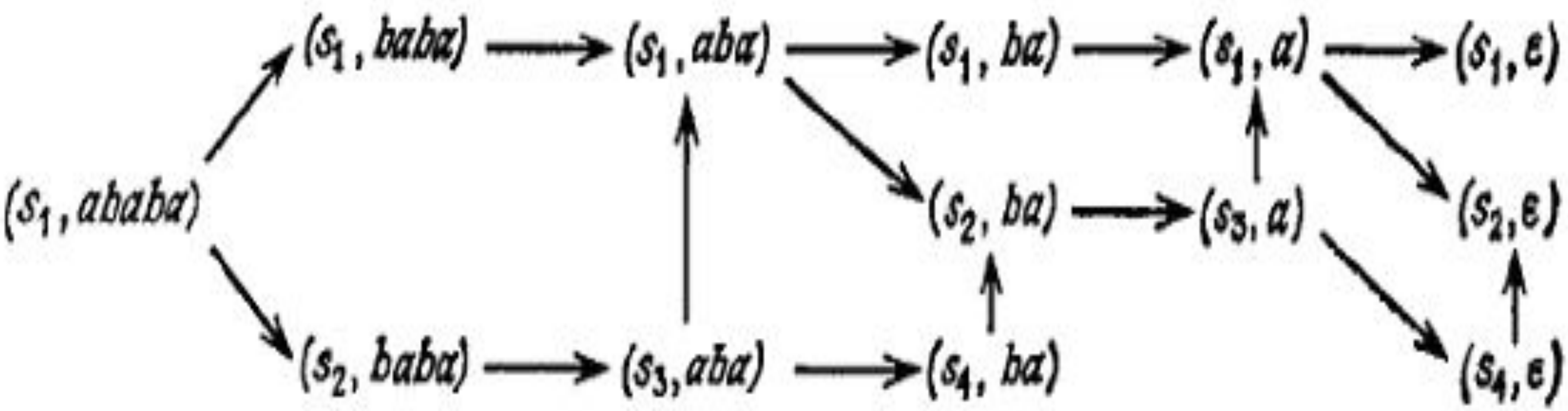
Рассмотрим НКА M, допускающий все те цепочки из символов a и b, которые оканчиваются цепочкой «aba», т.е. $L(M)=(a+b)^*aba$.

Пусть $M=(\{s_1,s_2,s_3,s_4\},\{a,b\},\delta,s_1,\{s_4\})$, где функция δ определена на рис. (без ε-переходов можно обойтись).

Вход / Состояние	**a**	**b**	**ε**
$\mathbf{s_1}$	$\{s_1,s_2\}$	$\{s_1\}$	$\varnothing$
$\mathbf{s_2}$	$\varnothing$	$\{s_3\}$	$\varnothing$
$\mathbf{s_3}$	{s4}	$\varnothing$	$\{s_1\}$
$\mathbf{s_4}$	$\varnothing$	$\varnothing$	$\{s_2\}$

Функция переходов δ

Пусть на вход автомата M подаётся цепочка «ababa». Тогда M может сработать в соответствии с последовательностями МО:



Последовательность МО для входа *ababa*.

Так как $(s_1, ababa) \vdash^* (s_4, \varepsilon)$ и s_4 – заключительное состояние, то цепочка ababa допускается автоматом M.

§2 Детерминированный конечный автомат

Детерминированный конечный автомат (ДКА) состоит из следующих элементов:

1. Конечное множество *состояний*, обозначаемое обычно как Q.
2. Конечное множество *входных символов*, обозначаемое обычно как Σ.
3. *Функция переходов*, аргументами которой являются текущее состояние и входной символ, а значением — новое состояние. Функция переходов обычно обозначается как δ. Представляя нестрого автомат в виде графа, мы изображали δ отмеченными дугами, соединяющими состояния. Если q — состояние и a — входной символ, то $\delta(q, a)$ — это состояние p, для которого существует дуга, отмеченная символом a и ведущая из q в p.[2]
4. *Начальное состояние*, одно из состояний в Q.
5. Множество *заключительных*, или *допускающих*, состояний F. Множество F является подмножеством Q.

Наиболее компактное представление ДКА — это список пяти вышеуказанных его компонентов. В доказательствах ДКА часто трактуется как пятерка

$$A = (Q, \Sigma, \delta, q_0, F),$$

где A — имя ДКА, Q — множество состояний, Σ — множество входных символов, δ — функция переходов, q_0 — начальное состояние и F — множество допускающих состояний.

Способы описания автоматов:

1. *Диаграмма переходов*, которая представляет собой граф.
2. *Таблица переходов*, дающая табличное представление функции δ. Из неё очевидны состояния и входной алфавит.

Диаграмма переходов для ДКА вида $A = (Q, \Sigma, \delta, q_0, F)$ есть граф, определяемый следующим образом:

а) всякому состоянию из Q соответствует некоторая вершина;

б) пусть $\delta(q, a) = p$ для некоторого состояния q из Q и входного символа a из Σ. Тогда диаграмма переходов должна содержать дугу из вершины q в вершину p, отмеченную a. Если существует несколько входных символов, переводящих автомат из состояния q в состояние p, то диаграмма переходов может содержать одну дугу, отмеченную списком этих символов;

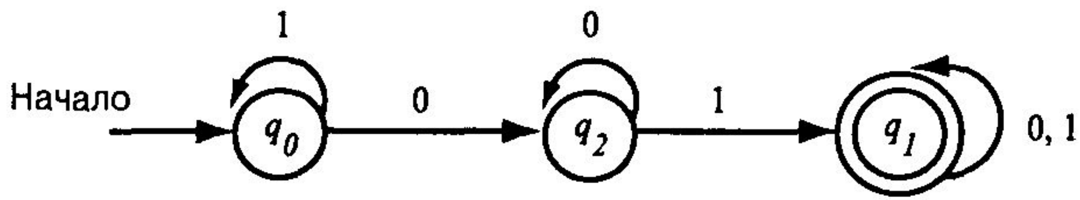
в) диаграмма содержит стрелку в начальное состояние, отмеченную как *Начало*. Эта стрелка не выходит ни из какого состояния;

г) вершины, соответствуюшие допускающим состояниям (состояниям из F), отмечаются двойным кружком. Состояния, не принадлежащие F, изображаются простым (одинарным) кружком.

Таблица переходов представляет собой обычное табличное представление функции, подобной δ, которая двум аргументам ставит в соответствие одно значение. Строки таблицы соответствуют состояниям, а столбцы — входным символам. На пересечении строки, соответствующей состоянию q, и столбца, соответствующего входному символу a, находится состояние $\delta(q, a)$.

ПРИМЕР:

ДКА, допускающий цепочки из 0 и 1, содержащие подцепочку 01.



	0	1
$\rightarrow q_0$	q_2	q_0
$*q_1$	q_1	q_1
q_2	q_2	q_1