ДМ РТ5
Лекция 13
Регулярные языки и конечные автоматы

Теория формальных языков

Три аспекта изучения языков:

1. **синтаксис** (правила составления слов и предложений, изучаются классической теорией формальных языков),
2. **семантика** (смысл текста),
3. **прагматика** (метасмысл – смысл, который хотел донести автор, цель создания текста).

Теория формальных языков

Три аспекта изучения языков:

1. **синтаксис** (правила составления слов и предложений, изучаются классической теорией формальных языков),
2. **семантика** (смысл текста),
3. **прагматика** (метасмысл – смысл, который хотел донести автор, цель создания текста).

Алфавит $V = \{a_1, \dots, a_n\}$ – непустое конечное (часто упорядоченное) множество элементов, называемых **буквами**.

Слово (**цепочка**) – конечный упорядоченный набор букв.

Примеры: $a_1,\ a_1a_1,\ a_1a_2 \neq a_2a_1,\ a_2a_2a_1$. λ – пустое слово.

V^* – множество всех слов, V^+ – множество всех непустых слов.

Теория формальных языков

Три аспекта изучения языков:

1. **синтаксис** (правила составления слов и предложений, изучаются классической теорией формальных языков),
2. **семантика** (смысл текста),
3. **прагматика** (метасмысл – смысл, который хотел донести автор, цель создания текста).

Алфавит $V = \{a_1, \dots, a_n\}$ – непустое конечное (часто упорядоченное) множество элементов, называемых **буквами**.

Слово (**цепочка**) – конечный упорядоченный набор букв.

Примеры: $a_1,\ a_1a_1,\ a_1a_2 \neq a_2a_1,\ a_2a_2a_1$. λ – пустое слово.

V^* – множество всех слов, V^+ – множество всех непустых слов.

Язык в алфавите V – подмножество в V^*.

V^* – универсальный язык.

Операции над языками: объединение, пересечение, разность, дополнение, соединение (конкатенация) и итерация.

Соединение (**конкатенация**) слов x и y – приписывание y справа к x. Обозначается: $x \cdot y = xy$

$(V^*, \ \cdot, \lambda)$ – свободный моноид, порождённый V.

Слово x называется **подсловом** (вхождением) в y, если ∃ такие слова u и v, что $y = uxv$, где u – **левое крыло**, v – **правое крыло**.

Первым (главным) вхождением называется вхождение с наименьшей длиной левого крыла.

Соединение (**конкатенация**) языков $L_1 L_2 = \{xy : x \in L_1 \text{ и } y \in L_2\}$.

Операции над языками: объединение, пересечение, разность, дополнение, соединение (конкатенация) и итерация.

Соединение (**конкатенация**) слов x и y – приписывание y справа к x. Обозначается: $x \cdot y = xy$

$(V^*, \ \cdot, \lambda)$ – свободный моноид, порождённый V.

Слово x называется **подсловом** (вхождением) в y, если ∃ такие слова u и v, что $y = uxv$, где u – **левое крыло**, v – **правое крыло**.

Первым (главным) вхождением называется вхождение с наименьшей длиной левого крыла.

Соединение (**конкатенация**) языков $L_1 L_2 = \{xy : x \in L_1 \text{ и } y \in L_2\}$.

Пример: над $V = \{a, b, c\}$ языки $L_1 = \{ab, bcc, cab\}$ и $L_2 = \{ca, bcc\}$,

$L_1 \cup L_2 = \{ab, bcc, cab, ca\}$, $L_1 \cap L_2 = \{bcc\}$, $L_1 \setminus L_2 = \{ab, cab\}$, $\bar{L}_1 = V^* \setminus L_1$,

$L_1 L_2 = \{abca, abbcc, bccca, bccbcc, cabca, cabbcc\}$,

$L_2 L_1 = \{caab, cabcc, cacab, bccab, bccbcc, bcccab\} \neq L_1 L_2$.

Теорема. $\mathcal{L}(V) = \left(2^{V^*},\ \cup,\ \cdot, \emptyset, \{\lambda\}\right)$ является замкнутым полукольцом.

Доказательство. Аксиомы идемпотентного полукольца выполняются очевидно,

естественный порядок является включением, $\sup\limits_{n\in\mathbb{N}} L_n = \bigcup_{n\in\mathbb{N}} L_n$,

непрерывность умножения $L(\bigcup_{i\in I} P_i) = \bigcup_{i\in I}(LP_i)$ и $(\bigcup_{i\in I} P_i)L = \bigcup_{i\in I}(P_iL)$ доказывается методом двух включений □

Теорема. $\mathcal{L}(V) = \left(2^{V^*},\ \cup,\ \cdot, \emptyset, \{\lambda\}\right)$ является замкнутым полукольцом.

Доказательство. Аксиомы идемпотентного полукольца выполняются очевидно,

естественный порядок является включением, $\sup\limits_{n\in\mathbb{N}} L_n = \bigcup_{n\in\mathbb{N}} L_n$,

непрерывность умножения $L(\bigcup_{i\in I} P_i) = \bigcup_{i\in I}(LP_i)$ и $(\bigcup_{i\in I} P_i)L = \bigcup_{i\in I}(P_iL)$ доказывается методом двух включений □

Возведение языка в натуральную степень $L^0 = \{\lambda\},\ L^n = L^{n-1}L$.

Итерация языка $L^* = \bigcup_{n=0}^{\infty} L^n$, **положительная итерация** $L^+ = \bigcup_{n=1}^{\infty} L^n$.

Если $\lambda \in L$, то $L^+ = L^*$; если $\lambda \notin L$, то $L^+ = L^* \setminus \{\lambda\}$.

Свойство: $L^+ = L^*L = LL^*$.

(Порождающей) грамматикой называется упорядоченная четвёрка $G = (V, N, S, P)$, где V **– терминальный алфавит**, N **– нетерминальный алфавит**, S **– аксиома** $(S \in N)$, P – конечное множество **правил вывода** $\alpha \to \beta$, где α и β – слова в объединённом алфавите $V \cup N$, причём α содержит хотя бы один нетерминальный символ.

Применение правила вывода $\alpha \to \beta$ к слову x означает замену подслова α в x на β.

Пример: правило $a_1 a_2 \to a_3$ применяем к слову $x = a_1 \underline{a_1 a_2} a_1 a_2 a_3$, получаем $a_1 a_3 a_1 a_2 a_3$.

Слово y называется **выводимым** из слова x в грамматике G, если y может быть получено из x путём последовательного применения конечное число раз правил вывода.

Обозначается: $x \vdash_G^* y$.

Языком $L(G)$, порождаемым грамматикой G, называется множество всех выводимых из аксиомы терминальных слов, т. е. слов в терминальном алфавите: $L(G) = \{x : S \vdash_G^* x, \ x \in V^*\}$.

Соглашение: набор правил $\alpha \to \beta_1, \ \alpha \to \beta_2, \ \dots, \ \alpha \to \beta_n$ будем писать $\alpha \to \beta_1 \mid \beta_2 \mid \dots \mid \beta_n$.

Пример: грамматика G_1, $V = \{$носорог, крокодил, львёнок, идёт, плывёт, лежит$\}$,

$N = \{$предложение, подлежащее, сказуемое$\}$,

$$S = \text{предложение},\ P_1 = \left\{ \begin{array}{c} \text{предложение} \rightarrow \text{подлежащее сказуемое}, \\ \text{подлежащее} \rightarrow \text{носорог} \mid \text{крокодил} \mid \text{львёнок} \\ \text{сказуемое} \rightarrow \text{идёт} \mid \text{плывёт} \mid \text{лежит} \end{array} \right\}.$$

Пример: грамматика G_1, $V = \{$носорог, крокодил, львёнок, идёт, плывёт, лежит$\}$,

$N = \{$предложение, подлежащее, сказуемое$\}$,

$$S = \text{предложение},\ P_1 = \left\{ \begin{array}{c} \text{предложение} \to \text{подлежащее сказуемое}, \\ \text{подлежащее} \to \text{носорог} \mid \text{крокодил} \mid \text{львёнок} \\ \text{сказуемое} \to \text{идёт} \mid \text{плывёт} \mid \text{лежит} \end{array} \right\}.$$

$L(G_1) = \{$ носорог идёт, носорог плывёт, носорог лежит, крокодил идёт, крокодил плывёт, крокодил лежит, львёнок идёт, львёнок плывёт, львёнок лежит $\}$.

Пример: грамматика G_1, $V = \{$носорог, крокодил, львёнок, идёт, плывёт, лежит$\}$,

$N = \{$предложение, подлежащее, сказуемое$\}$,

$$S = \text{предложение},\ P_1 = \left\{ \begin{array}{c} \text{предложение} \to \text{подлежащее сказуемое}, \\ \text{подлежащее} \to \text{носорог} \mid \text{крокодил} \mid \text{львёнок} \\ \text{сказуемое} \to \text{идёт} \mid \text{плывёт} \mid \text{лежит} \end{array} \right\}.$$

$L(G_1) = \{$ носорог идёт, носорог плывёт, носорог лежит, крокодил идёт, крокодил плывёт, крокодил лежит, львёнок идёт, львёнок плывёт, львёнок лежит $\}$.

Пример: грамматика $G_2 = (\{a, b, 0, 1\},\ \{I, D\},\ I,\ P_2),\ P_2 = \left\{ \begin{array}{l} I \to aD \mid bD \mid a \mid b\ , \\ D \to aD \mid bD \mid 0D \mid 1D \mid a \mid b \mid 0 \mid 1 \end{array} \right\}$.

Пример: грамматика G_1, $V = \{$носорог, крокодил, львёнок, идёт, плывёт, лежит$\}$,

$N = \{$предложение, подлежащее, сказуемое$\}$,

$$S = \text{предложение},\ P_1 = \left\{ \begin{array}{c} \text{предложение} \to \text{подлежащее сказуемое}, \\ \text{подлежащее} \to \text{носорог} \mid \text{крокодил} \mid \text{львёнок} \\ \text{сказуемое} \to \text{идёт} \mid \text{плывёт} \mid \text{лежит} \end{array} \right\}.$$

$L(G_1) = \{$ носорог идёт, носорог плывёт, носорог лежит, крокодил идёт, крокодил плывёт, крокодил лежит, львёнок идёт, львёнок плывёт, львёнок лежит $\}$.

Пример: грамматика $G_2 = (\{a, b, 0, 1\},\ \{I, D\},\ I,\ P_2),\ P_2 = \left\{ \begin{array}{l} I \to aD \mid bD \mid a \mid b\ , \\ D \to aD \mid bD \mid 0D \mid 1D \mid a \mid b \mid 0 \mid 1 \end{array} \right\}$.

G_2 порождает язык из всевозможных слов, начинающихся с буквы a или b.

Пример: грамматика G_1, $V = \{$носорог, крокодил, львёнок, идёт, плывёт, лежит$\}$,

$N = \{$предложение, подлежащее, сказуемое$\}$,

$$S = \text{предложение},\ P_1 = \left\{ \begin{array}{c} \text{предложение} \to \text{подлежащее сказуемое}, \\ \text{подлежащее} \to \text{носорог} \mid \text{крокодил} \mid \text{львёнок} \\ \text{сказуемое} \to \text{идёт} \mid \text{плывёт} \mid \text{лежит} \end{array} \right\}.$$

$L(G_1) = \{$ носорог идёт, носорог плывёт, носорог лежит, крокодил идёт, крокодил плывёт, крокодил лежит, львёнок идёт, львёнок плывёт, львёнок лежит $\}$.

Пример: грамматика $G_2 = (\{a, b, 0, 1\},\ \{I, D\},\ I,\ P_2),\ P_2 = \left\{ \begin{array}{l} I \to aD \mid bD \mid a \mid b\ , \\ D \to aD \mid bD \mid 0D \mid 1D \mid a \mid b \mid 0 \mid 1 \end{array} \right\}$.

G_2 порождает язык из всевозможных слов, начинающихся с буквы a или b.

Пример: грамматика $G_3 = (\{\ (\ ,\ )\ \},\ \{S\},\ S,\ \{S \to () \mid (S) \mid SS\})$ порождает язык из правильных скобочных структур.

Грамматика называется **регулярной**, если каждое её правило вывода имеет вид $A \to aB$ или $A \to a$, где $A, B \in N$ и $a \in V \cup \{\lambda\}$.

Пример: G_2 – регулярная, G_1 и G_3 – нет.

Язык называется **регулярным**, если ∃ порождающая его регулярная грамматика.

Обозначим через $\mathcal{R}(V)$ подполукольцо в $\mathcal{L}(V)$, порождённое множеством из $\emptyset$, $\{\lambda\}$ и всех языков $\{a\}$, где $a \in V$, и замкнутое относительно итерации.

Теорема: язык L регулярный $\Leftrightarrow$ $L \in \mathcal{R}(V)$.

Множество регулярных языков $\mathcal{R}(V)$ может быть описано индуктивно:

- языки $\emptyset$, $\{\lambda\}$ и $\{a\}$, где $a \in V$, считаются регулярными,
- для любых регулярных языков P и Q их объединение $P \cup Q$, соединение PQ и итерация P^* также считаются регулярными,
- никаких других регулярных языков нет.

Регулярные языки удобно записывать в виде **регулярных выражений**:

- языки $\emptyset, \{\lambda\}$ и $\{a\}$ записываются символами $\emptyset, \lambda$ и a,
- объединение, соединение и итерация записываются как $p + q$, pq и p^*.

Регулярные выражения называются **эквивалентными**, если они обозначают один и тот же регулярный язык.

Пример: $a^* + (bc)^* = a^+ + (bc)^*$,

$a^* = \{\lambda, a, aa, aaa, aaaa, \dots\}, \qquad a^+ = \{a, aa, aaa, aaaa, \dots\},$

$(bc)^* = \{\lambda, bc, bcbc, bcbcbc, \dots\},$

$a^* + (bc)^* = \{\lambda, a, aa, aaa, \dots, bc, bcbc, bcbcbc, \dots\} = a^+ + (bc)^*.$

Конечным автоматом называется орграф, размеченный над полукольцом $\mathcal{R}(V)$ регулярных языков в алфавите V, причём в орграфе выделены начальная вершина q_0 и множество заключительных вершин F, а меткой каждой дуги является либо язык $\{\lambda\}$, либо непустое подмножество в V.

Множество вершин $Q = \{q_0, \dots, q_n\}$. Метки записываются как регулярные выражения.

Метка пути в размеченном орграфе – соединение меток дуг, составляющих путь – регулярный язык.

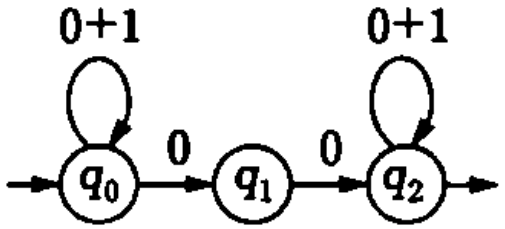
Про слово x из метки пути γ говорят, что x **читается** на пути γ, а путь γ **несёт** слово x.

Языком $L(M)$ конечного автомата M называется множество всех слов в алфавите V, читаемых на пути из начальной вершины q_0 в одну из заключительных вершин, т. е. сумма стоимостей прохождения из начальной вершины в заключительные.

Про язык $L(M)$ и каждое его слово x говорят, что они **допускаются** (**воспринимаются**) конечным автоматом M.

Пример: для $V = \{0, 1\}$

меткой пути $q_0 q_0 q_1 q_2$ будет $(0 + 1)00 = 000 + 100$,

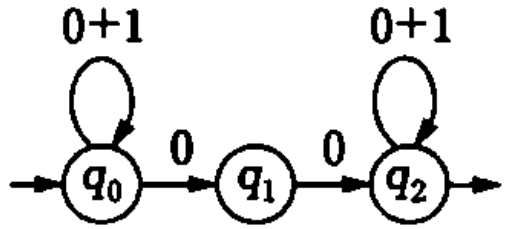


Пример: для $V = \{0, 1\}$

меткой пути $q_0 q_0 q_1 q_2$ будет $(0 + 1)00 = 000 + 100$,

$L(M) = (0 + 1)^* 00 (0 + 1)^*$ – множество всех слов,

содержащих два нуля подряд.

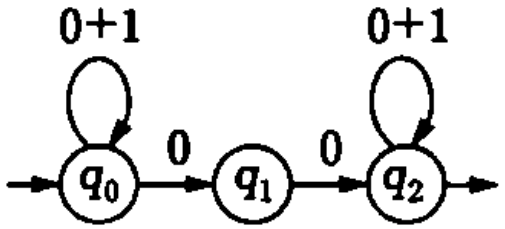


Пример: для $V = \{0, 1\}$

меткой пути $q_0 q_0 q_1 q_2$ будет $(0 + 1)00 = 000 + 100$,

$L(M) = (0 + 1)^* 00 (0 + 1)^*$ – множество всех слов,

содержащих два нуля подряд.



Механическая интерпретация конечного автомата:

- слово в алфавите V записывается на бесконечную вправо ленту,
- Q – множество всех внутренних состояний блока управления, который обозревает текущую ячейку ленты и за каждый такт сдвигается на одну ячейку вправо, при этом может меняться внутреннее состояние,
- слово будет допустимым, если автомат, начиная с первой буквы слова и начального состояния q_0, через конечное число тактов переходит в одно из заключительных состояний.

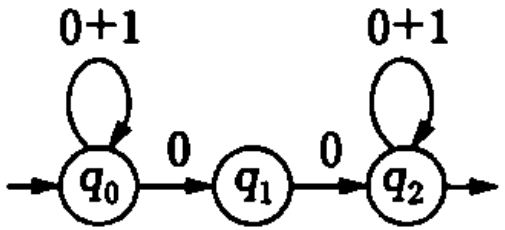
Эквивалентно конечный автомат – упорядоченная пятёрка $M = (V, Q, q_0, F, \delta)$, где V – входной алфавит, Q – множество состояний, $q_0 \in Q$ – начальное состояние, $F \subseteq Q$ – множество заключительных состояний, $\delta : Q \times (V \cup \{\lambda\}) \to 2^Q$, $\delta(q, a) = \{r : q \to_a r\}$ – функция переходов, такая что $\forall q \in Q,\ \forall a \in V,\ \delta(q, a) \cap \delta(q, \lambda) = \emptyset$.

Если $\delta(q, a) = \{r_1, \dots, r_n\}$, то из q идут дуги в $r_1, \dots, r_n$ с весом a.

Эквивалентно конечный автомат – упорядоченная пятёрка $M = (V, Q, q_0, F, \delta)$, где V – входной алфавит, Q – множество состояний, $q_0 \in Q$ – начальное состояние, $F \subseteq Q$ – множество заключительных состояний, $\delta : Q \times (V \cup \{\lambda\}) \to 2^Q$, $\delta(q, a) = \{r : q \to_a r\}$ – функция переходов, такая что $\forall q \in Q,\ \forall a \in V,\ \delta(q, a) \cap \delta(q, \lambda) = \emptyset$.

Если $\delta(q, a) = \{r_1, \dots, r_n\}$, то из q идут дуги в $r_1, \dots, r_n$ с весом a.

Пример:



$\delta(q_0, 0) = \{q_0, q_1\}$, $\delta(q_0, 1) = \{q_0\}$,

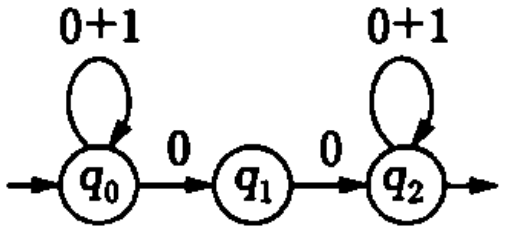
$\delta(q_1, 0) = \{q_2\}$, $\delta(q_1, 1) = \emptyset$,

$\delta(q_2, 0) = \{q_2\}$, $\delta(q_2, 1) = \{q_2\}$.

Эквивалентно конечный автомат – упорядоченная пятёрка $M = (V, Q, q_0, F, \delta)$, где V – входной алфавит, Q – множество состояний, $q_0 \in Q$ – начальное состояние, $F \subseteq Q$ – множество заключительных состояний, $\delta : Q \times (V \cup \{\lambda\}) \to 2^Q$, $\delta(q, a) = \{r : q \to_a r\}$ – функция переходов, такая что $\forall q \in Q,\ \forall a \in V,\ \delta(q, a) \cap \delta(q, \lambda) = \emptyset$.

Если $\delta(q, a) = \{r_1, \dots, r_n\}$, то из q идут дуги в $r_1, \dots, r_n$ с весом a.

Пример:



$\delta(q_0, 0) = \{q_0, q_1\}$, $\delta(q_0, 1) = \{q_0\}$,

$\delta(q_1, 0) = \{q_2\}$, $\delta(q_1, 1) = \emptyset$,

$\delta(q_2, 0) = \{q_2\}$, $\delta(q_2, 1) = \{q_2\}$.

Конечный автомат называется полностью **определённым**, если $\forall q \in Q,\ \forall a \in V,\ \delta(q, a) \neq \emptyset$.

Конечный автомат называется **детерминированным**, если $\forall q \in Q,\ \delta(q, \lambda) = \emptyset$ и $\forall q \in Q,$ $\forall a \in V,\ |\delta(q, a)| = 1$; и **квазидетерминированным**, если в последнем условии $|\delta(q, a)| \leq 1$.

Для конечного автомата:

- задача **анализа –** вычисление языка, допускаемого заданным конечным автоматом;
- задача **синтеза –** построение конечного автомата по регулярному выражению.

Решение задачи анализа

Язык L конечного автомата есть сумма стоимостей прохождения из начальной вершины в заключительные, т. е. сумма элементов матрицы стоимостей, расположенных в строке начальной вершины и столбцах заключительных вершин.

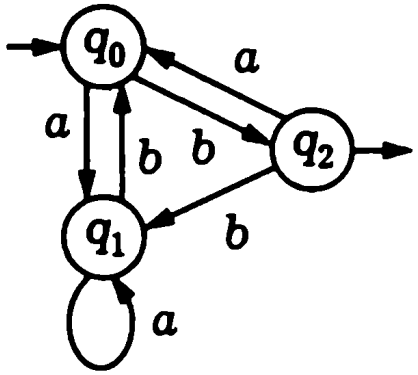
Для матрицы меток дуг A орграфа, задающего конечный автомат, составляем систему

$$X = AX + B,$$

где B – столбец, компоненты которого с номерами заключительных состояний равны λ (единице полукольца $\mathcal{R}(V)$), а остальные компоненты равны $\emptyset$ (нулю полукольца $\mathcal{R}(V)$).

Тогда $X = A^*B$, где A^* – матрица стоимостей, и $L = x_s$, где s – номер начального состояния.

Пример. Найдём язык конечного автомата.



Напишем матрицу меток дуг и составим систему:

$$A = \begin{pmatrix} \emptyset & a & b \\ b & a & \emptyset \\ a & b & \emptyset \end{pmatrix}, \qquad \begin{cases} x_0 = ax_1 + bx_2 \\ x_1 = bx_0 + ax_1 \\ x_2 = ax_0 + bx_1 + \lambda \end{cases}$$

Так как вершина q_0 начальная, то искомый язык $L = x_0$.

Из второго уравнения системы находим $x_1 = a^*bx_0$ и подставляем в другие уравнения:

$$\begin{cases} x_0 = aa^*bx_0 + bx_2 = a^*bx_0 + bx_2 \\ x_2 = ax_0 + ba^*bx_0 + \lambda = (a + ba^*b)x_0 + \lambda \end{cases}$$

Теперь выражение для x_2 подставляем в первое уравнение:

$$x_0 = a^*bx_0 + b(a + ba^*b)x_0 + b\lambda = (a^*b + ba + b^2a^*b)x_0 + b,$$

отсюда $L = x_0 = (a^*b + ba + b^2a^*b)^*b$.