

Теория и реализация языков программирования.

Задание 4: Замкнутость регулярных языков, теорема Майхилла-Нероуда и минимальные автоматы

Сергей Володин, 272 гр.

задано 2013.09.25

Упражнение 1

1. Докажем, что алгоритм конечен. Q можно разделить не больше, чем на $|Q|$ подмножеств, на каждом шаге происходит некоторое разделение.

Действительно, на каждом шаге и на каждом символе количество подмножеств не уменьшается, так как $Q_{k,l}$ различно при разных k (значит, элементы из разных «старых» подмножеств попадут в разные «новые» подмножества).

А если количество подмножеств не увеличилось после $|\Sigma|$ разбиений, алгоритм завершается (по построению).

2. Докажем, что все состояния из одного подмножества эквивалентны. Предположим противное. Тогда

$$\exists q_1, q_2 \in Q_i: q_1 \not\sim_L q_2 \Rightarrow \forall x_1, x_2: q_0 \xrightarrow{x_1} q_1, q_0 \xrightarrow{x_2} q_2 \hookrightarrow x_1 \not\sim_L x_2 \Rightarrow \exists w \in \Sigma^*: x_1 w \in L, x_2 w \notin L.$$

Фиксируем x_1, x_2, w . Тогда $\delta(q_1, w) \in F, \delta(q_2, w) \notin F$. Пусть $|w| = n$.

Если $|w| = n = 0$, то получаем, что q_1 — принимающее, а q_2 — нет. Это противоречие, так как $q_1, q_2 \in Q_i$, на первом шаге принимающие и не принимающие были разделены, и (как было доказано выше), состояния, лежащие в различных подмножествах в процессе выполнения алгоритма не могут оказаться в одном.

Пусть $|w| = n > 0$. $w = w_1 \dots w_n$. Тогда $(q_1, w) \vdash (q_1^1, w_2 \dots w_n) \vdash \dots \vdash (q_1^n, \varepsilon), q_1^n \in F$. Аналогично $(q_2, w) \vdash (q_2^1, w_2 \dots w_n) \vdash \dots \vdash (q_2^n, \varepsilon), q_2^n \notin F$. Поскольку $q_1, q_2 \in Q_i$, $\delta(q_1, w_1)$ и $\delta(q_2, w_1) \in Q_j$ по условию окончания алгоритма. Значит, q_1^1 и q_2^1 лежат в одном подмножестве. Повторяя рассуждение, получаем, что q_1^n и q_2^n лежат в одном подмножестве, что невозможно (доказано выше), так как $q_1^n \in F, q_2^n \notin F$.

- 2.1. Получаем, что были склеены только эквивалентные состояния. Значит, язык, распознаваемый автоматом, не изменился.

3. Докажем, что если некоторые два состояния q_1, q_2 исходного автомата были эквивалентны, они будут в одном подмножестве Q_i . Пусть иначе: они были разделены на некотором шаге.

Это не мог быть второй шаг, так как принимающее и не принимающее состояние не эквивалентны. Докажем это: пусть $F \ni q_1 \sim_L q_2 \notin F \Rightarrow \exists x_1 \sim_L x_2: \delta(q_0, x_1) = q_1 \in F, \delta(q_0, x_2) = q_2 \notin F \Rightarrow x_1 \in L, x_2 \notin L$. $x_1 \sim_L x_2 \Rightarrow \forall w \in \Sigma^* \hookrightarrow x_1 w \in L \Leftrightarrow x_2 w \in L$. Выберем $w = \varepsilon$. Тогда $x_1 \in L \Leftrightarrow x_2 \in L$ — противоречие.

Значит, они были разделены на некотором последующем шаге. Найдем первый такой шаг, на котором некоторые эквивалентные состояния q_1, q_2 были разделены. Пусть при этом рассматривался символ $\sigma: q_1 \xrightarrow{\sigma} q_a \in Q_a, q_2 \xrightarrow{\sigma} q_b \in Q_b, Q_a \neq Q_b$. Поскольку до этого эквивалентные состояния оставались в одном подмножестве, получаем, что q_a и q_b не эквивалентны (если это не так, то этот шаг не первый из таких, на котором эквивалентные состояния были разделены — противоречие). Значит (доказано ранее), $\exists w: \delta(q_a, w) \in F, \delta(q_b, w) \notin F$. Тогда $\delta(q_1, \sigma w) \in F, \delta(q_2, \sigma w) \notin F \Rightarrow$ (доказано ранее) состояния q_1, q_2 не эквивалентны — противоречие.

- 3.1. Получаем, что эквивалентные состояния, и только они, будут склеены. Также количество состояний ДКА не может быть меньше, чем количество классов эквивалентности по \sim_L (доказано в условии). Больше оно тоже быть не может, так как тогда бы в автомате были два эквивалентных состояния, что невозможно (они все были склеены). Значит, количество состояний построенного ДКА будет равно количеству классов эквивалентности по \sim_L .

4. (Далее считаем Q_i за состояния). Установим биекцию между классами эквивалентности и состояниями минимального ДКА, которая сохраняет функцию переходов, т.е. построим изоморфизм $\varphi: \{Q_i\} \leftrightarrow \{C_i\}$. На классах эквивалентности функцию переходов определим так: $x_i \in C_i \Rightarrow \delta(C_i, \sigma) = C(x_i \sigma)$ (эта же функция является функцией переходов ДКА из доказательства теоремы 1 третьего задания). Выполним обход графа минимального ДКА и найдем слова x_i , по которым можно попасть в $Q_i: \delta(Q_0, x_i) = Q_i$. Определим $\varphi(Q_i) = C(x_i)$. Поскольку состояния Q_i попарно неэквивалентны (иначе бы они были склеены), слова x_i попарно не эквивалентны. Значит, $C(x_i)$ попарно различны, и φ инъективно. Но поскольку $|\{Q_i\}| = |\{C_i\}|$, оно биективно. Обозначим $C_i = C(x_i) = \varphi(Q_i)$. Докажем сохранение функции переходов:

Пусть $\delta(Q_i, \sigma) = Q_j$. Тогда $\delta(Q_0, x_j) = \delta(Q_0, x_i \sigma) = Q_j$. Поэтому $\forall w \in \Sigma^* \hookrightarrow L \ni x_j w \Leftrightarrow \delta(Q_0, x_j w) \equiv \delta(Q_j, w) \equiv \delta(Q_i, \sigma w) \equiv \delta(Q_0, x_i \sigma w) \in F \Leftrightarrow x_i \sigma w \in L$. Значит, $x_j \sim_L x_i \sigma \Rightarrow C_j = C(x_j) = C(x_i \sigma) = \delta(C_i, \sigma)$ ■.

Обратно: $\delta(C_i, \sigma) = C_j \Rightarrow x_i \sigma \sim_L x_j \Rightarrow$ состояния $\delta(Q_0, x_i \sigma)$ и $\delta(Q_0, x_j)$ эквивалентны, а значит, что они совпадают (доказано ранее). Но $Q_j = \delta(Q_0, x_j) = \delta(Q_0, x_i \sigma) = \delta(\delta(Q_0, x_i), \sigma) = \delta(Q_i, \sigma)$ ■.

- 4.1. Таким образом доказано, что любой минимальный ДКА изоморфен в смысле сохранения функции переходов классам эквивалентности. Значит, любые два минимальных ДКА \mathcal{A}, \mathcal{B} для данного языка изоморфны между собой (можно построить изоморфизм $\varphi_{\mathcal{A}, \mathcal{B}}: Q^{\mathcal{A}} \leftrightarrow Q^{\mathcal{B}}$ как композицию изоморфизмов $Q^{\mathcal{A}} \leftrightarrow \{C_i\}, \{C_i\} \leftrightarrow Q^{\mathcal{B}}$).

Задача 1

$L \subset \Sigma^* \in \text{REG}$, $\Sigma^* / \sim_L = \{C_i\} \equiv \{C_0, \dots, C_n\}$ (n неизвестно, C_i попарно различны).

$f: \Sigma^* \times \Sigma^* \longrightarrow \{0, 1\}$ — задана, $f(x, y) = 1 \Leftrightarrow x \sim_L y$. $g: \Sigma^* \longrightarrow \{0, 1\}$ — задана, $g(w) = 1 \Leftrightarrow w \in L$.

Построим ДКА $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$: $L(\mathcal{A}) = L$, изоморфный минимальному ДКА (определение изоморфизма в упражнении). $Q = \{q_i\}$ — множество состояний.

$X: Q \longrightarrow \Sigma^*$ — представители классов, соответствующих состояниям. Для краткости будем писать $x_i \equiv X(q_i)$.

В алгоритме для состояний из Q всегда известен представитель соответствующего класса (по построению, добавляются состояния только с известными представителями).

1. $\Sigma^* \ni \varepsilon$ принадлежит какому-то классу. Без ограничения общности $\varepsilon \in C_0$. Добавим состояние q_0 , соответствующее $C(\varepsilon)$. Определим его как начальное.
Рассмотрим все $\sigma_k \in \Sigma$.
 1. Если класс $C(\varepsilon\sigma_k)$ еще не встречался (не соответствует ни одному состоянию), то есть, $\forall q_i \in Q \hookrightarrow f(x_i, \varepsilon\sigma_k) = 0$, то добавим в Q новое состояние q , которое будет соответствовать классу $C(\varepsilon\sigma_k)$. Для него известен представитель соответствующего класса — σ_k . Поскольку $\delta(C_0, \sigma_k) = C(\varepsilon\sigma_k)$ (см. упражнение), для установления изоморфизма необходимо направить переход из q_0 в q : $\delta(q_0, \sigma_k) = q$.
 2. Иначе $\exists q_i \in Q$: $f(x_i, \varepsilon\sigma_k) = 1 \Rightarrow x_i \sim_L \varepsilon\sigma_k$, то есть, $\delta(C_0, \sigma_k) = C(x_i)$. Поэтому необходимо направить переход из q_0 в q_i : $\delta(q_0, \sigma_k) = q_i$.

Заметим, что определены все переходы из q_0 , и, возможно, добавлены новые состояния. Повторим алгоритм для них:

2. (цикл) Если имеется состояние q_i , которому соответствует класс эквивалентности C_i с найденным представителем x_i , и для q_i не определен переход по σ_k , то рассмотрим два варианта. Иначе выход.
 1. Класс $C(x_i\sigma_k)$ не встречался $\Leftrightarrow \forall q_j \in Q \hookrightarrow f(x_j, x_i\sigma_k) = 0$. Тогда добавим новое состояние q (представитель $x_i\sigma_k$ известен) и определим переход из q_i по σ_k : $\delta(q_i, \sigma_k) = q$.
 2. Иначе $\exists q_j \in Q$: $f(x_j, x_i\sigma) = 1$. Определим переход: $\delta(q_i, \sigma_k) = q_j$.

Переходов в автомате не больше, чем $|Q||\Sigma|$. Поскольку строится изоморфизм между Q и $\{C\}$, то $|Q| \leq |\{C\}| \Rightarrow$ количество переходов конечно \Rightarrow цикл (2) будет конечным \Rightarrow алгоритм завершится. По построению $\forall q \in Q \hookrightarrow \delta(q_i, \sigma) = q_j$, где q_j соответствует $C(x_i\sigma)$. Также автомат полный. Из этих двух свойств заключаем («все переходы есть, и они такие, какими должны быть»), что множества состояний Q и минимального ДКА изоморфны.

Осталось определить принимающие состояния:

1. Выполним обход графа автомата, найдем кратчайшие пути y_i (слова) до каждого состояния q_i .
2. Для каждого состояния выясним, должно быть оно принимающим или нет: $g(y_i) \stackrel{?}{=} 1 \Leftrightarrow y_i \in L \Leftrightarrow \delta(q_0, y_i) = q_i \in F$.

Задача 2

Задача 3

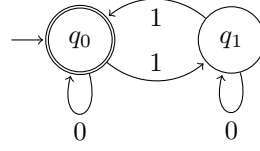
Пусть x, y — РВ. Ответим на вопрос $L(x) \stackrel{?}{=} L(y)$.

1. Построим по x, y НКА A, B .
2. Построим по A, B ДКА A', B'
3. Построим по A', B' минимальные ДКА $\mathcal{A}'', \mathcal{B}''$.
- 4.1 В случае, если $L(\mathcal{A}'') = L(\mathcal{B}'')$, они будут изоморфны (в смысле сохранения функции перехода, доказано в упражнении), что можно проверить одновременным обходом их графов.
- 4.2 Иначе тот же обход графов покажет, что автоматы различны.

Данный алгоритм не является эффективным, так как количество состояний построенного в (2) ДКА может экспоненциально зависеть от количества состояний НКА, и каждое состояние нужно как минимум создать за $O(1)$, а количество состояний НКА не меньше, чем длина РВ. То есть, $T = \Omega(2^{|Q^A|} + 2^{|Q^B|}) = \Omega(2^{|x|} + 2^{|y|})$ — не полином.

Задача 4

1. $\Sigma = \{0, 1\}$. Докажем, что $L(\mathcal{A}) = L$, $L_1 \equiv L = \{w \mid |w|_1 = 2t, t \in \mathbb{Z}\}$, ДКА \mathcal{A} :



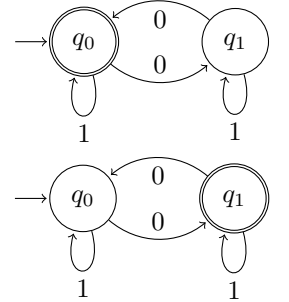
Докажем утверждение $P(n) = [\forall w \in \Sigma^*: |w| = n \hookrightarrow (q_0 \xrightarrow{w} q_i \Rightarrow i = |w|_1 \bmod 2)]$.

- (a) Докажем $P(0)$. Поскольку $|w| = 0 \Rightarrow w = \varepsilon$, $P(0) = [q_0 \xrightarrow{\varepsilon} q_i \Rightarrow i = |\varepsilon|_1 \bmod 2]$. Поскольку $\delta(q_0, \varepsilon) = q_0$, и $0 = |\varepsilon|_1$, получаем $P(0)$ ■
- (b) Пусть доказано $P(n)$, докажем $P(n+1)$. $P(n) = [\forall w \in \Sigma^*: |w| = n \hookrightarrow (q_0 \xrightarrow{w} q_i \Rightarrow i = |w|_1 \bmod 2)]$. Фиксируем $w \in \Sigma^*, |w| = n+1, w = w_0\sigma, |w_0| = n, |\sigma| = 1$. \mathcal{A} — полный $\Rightarrow (q_0, w) \equiv (q_0, w_0\sigma) \vdash^* (q_i, \sigma) \vdash (q_j, \varepsilon)$. $|w_0| = n \xrightarrow{P(n)} i = |w_0|_1 \bmod 2$. $i \in \{0, 1\}, \sigma \in \{0, 1\} \Rightarrow$ рассмотрим четыре случая:
- $(i = 0, \sigma = 0)$. $(q_0, w_0) \vdash^* (q_0, 0) \vdash (q_0, \varepsilon) \Rightarrow q_0 \xrightarrow{w} q_0 \Rightarrow j = 0$. $|w|_1 \bmod 2 = |w_0|_1 \bmod 2 + |0|_1 \bmod 2 = 0 + 0 = 0 \Rightarrow 0 = j = |w|_1 \bmod 2 = 0$.
 - $(i = 0, \sigma = 1)$. $(q_0, w_0) \vdash^* (q_0, 1) \vdash (q_1, \varepsilon) \Rightarrow q_0 \xrightarrow{w} q_1 \Rightarrow j = 1$. $|w|_1 \bmod 2 = |w_0|_1 \bmod 2 + |1|_1 \bmod 2 = 0 + 1 = 1 \Rightarrow 1 = j = |w|_1 \bmod 2 = 1$.
 - $(i = 1, \sigma = 0)$. $(q_0, w_0) \vdash^* (q_1, 0) \vdash (q_1, \varepsilon) \Rightarrow q_0 \xrightarrow{w} q_1 \Rightarrow j = 1$. $|w|_1 \bmod 2 = |w_0|_1 \bmod 2 + |0|_1 \bmod 2 = 1 + 0 = 1 \Rightarrow 1 = j = |w|_1 \bmod 2 = 1$.
 - $(i = 1, \sigma = 1)$. $(q_0, w_0) \vdash^* (q_1, 1) \vdash (q_0, \varepsilon) \Rightarrow q_0 \xrightarrow{w} q_0 \Rightarrow j = 0$. $|w|_1 \bmod 2 = |w_0|_1 \bmod 2 + |1|_1 \bmod 2 = (1 + 1) \bmod 2 = 0 \Rightarrow 0 = j = |w|_1 \bmod 2 = 0$.

Таким образом, $\forall n \in \mathbb{N} \cup \{0\} \hookrightarrow P(n) \Rightarrow \forall n \in \mathbb{N} \cup \{0\} \hookrightarrow [\forall w \in \Sigma^*: |w| = n \hookrightarrow (q_0 \xrightarrow{w} q_i \Rightarrow i = |w|_1 \bmod 2)] \Rightarrow \forall w \in \Sigma^* \hookrightarrow q_0 \xrightarrow{w} q_{|w|_1 \bmod 2}$. Пусть $w \in L \Leftrightarrow |w|_1 \bmod 2 = 0 \Leftrightarrow q_0 \xrightarrow{w} q_0 \Leftrightarrow w \in L(\mathcal{A})$ ■

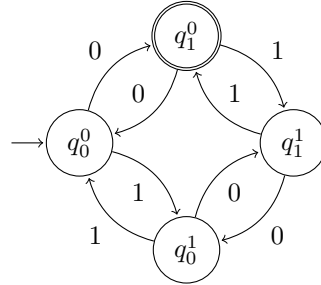
2. $\Sigma = \{0, 1\}$. $L_2 = \{w \mid |w|_0 = 2t + 1, t \in \mathbb{Z}\}$. Воспользуемся результатом (4.1) и построим ДКА \mathcal{B} :

Поменяем в автомате из (4.1) нули и единицы местами. Получим \mathcal{A}' . Очевидно, \mathcal{A}' будет распознавать все слова, в которых четное количество нулей. \mathcal{A}' — полный, и все состояния достижимы из q_0 .



Поэтому, переопределив $F'' = Q'' \setminus F$, получим $\mathcal{A}'' \equiv \mathcal{B}$, который распознает все слова, в которых нечетное количество нулей.

3. Поскольку $L_3 = \{\text{слова из 0 и 1, в которых четное число единиц и нечетное число нулей}\} = \{\text{слова из 0 и 1, в которых четное число единиц}\} \cap \{\text{слова из 0 и 1, в которых нечетное число нулей}\} \equiv L_1 \cap L_2$, построим \mathcal{C} : $L(\mathcal{C}) = L_3$ по алгоритму, который докажем далее, в (4.4):



4. Дано: Σ — алфавит, $\mathcal{A} = (Q^{\mathcal{A}}, \Sigma, q_0^{\mathcal{A}}, \delta^{\mathcal{A}}, F^{\mathcal{A}})$, $\mathcal{B} = (Q^{\mathcal{B}}, \Sigma, q_0^{\mathcal{B}}, \delta^{\mathcal{B}}, F^{\mathcal{B}})$ — полные ДКА, в которых все состояния достижимы из начальных. $\Sigma^* \supset L^{\mathcal{A}} = L(\mathcal{A})$, $\Sigma^* \supset L^{\mathcal{B}} = L(\mathcal{B})$. Задача: построить ДКА $\mathcal{C} = (Q^{\mathcal{C}}, \Sigma, q_0^{\mathcal{C}}, \delta^{\mathcal{C}}, F^{\mathcal{C}})$: $L(\mathcal{C}) = L^{\mathcal{A}} \cap L^{\mathcal{B}}$.

Определим $Q^{\mathcal{C}} = Q^{\mathcal{A}} \times Q^{\mathcal{B}}$ — множество всех пар состояний исходных автоматов.

Для краткости будем обозначать $Q^{\mathcal{C}} \ni (q_i^{\mathcal{A}}, q_j^{\mathcal{B}}) \stackrel{\text{def}}{=} q_j^i$.

Определим $q_0^{\mathcal{C}} \stackrel{\text{def}}{=} q_0^0$, $F^{\mathcal{C}} = \{q_j^i \mid q_i^{\mathcal{A}} \in F^{\mathcal{A}} \wedge q_j^{\mathcal{B}} \in F^{\mathcal{B}}\}$

Определим $\delta^{\mathcal{C}}(q_j^i, \sigma) = (\delta^{\mathcal{A}}(q_i^{\mathcal{A}}, \sigma), \delta^{\mathcal{B}}(q_j^{\mathcal{B}}, \sigma))$

Докажем утверждение

$P(n) = [\forall w \in \Sigma^*: |w| = n \hookrightarrow q_0^0 \xrightarrow{w} (\delta^{\mathcal{A}}(q_0^{\mathcal{A}}, w), \delta^{\mathcal{B}}(q_0^{\mathcal{B}}, w))]$

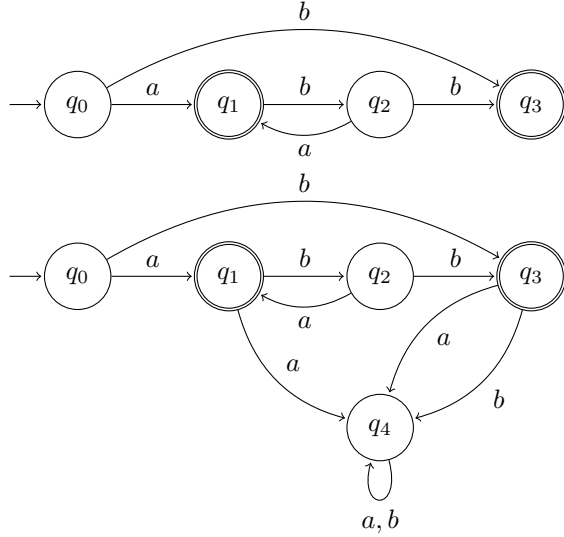
- $(n = 0)$ $\Sigma^* \ni w, |w| = 0 \Rightarrow w = \varepsilon$. Тогда $\delta^{\mathcal{C}}(q_0^0, \varepsilon) \stackrel{\text{но}}{=}^{\text{онп.}} (\delta^{\mathcal{A}}(q_0^{\mathcal{A}}, \varepsilon), \delta^{\mathcal{B}}(q_0^{\mathcal{B}}, \varepsilon))$, как и требовалось.
- $(n = 1)$ $\Sigma^* \ni w, |w| = 1 \Rightarrow w = \sigma \in \Sigma$. Тогда $\delta^{\mathcal{C}}(q_0^0, w) = \delta^{\mathcal{C}}(q_0^0, \sigma) \stackrel{\text{но}}{=}^{\text{онп.}} (\delta^{\mathcal{A}}(q_0^{\mathcal{A}}, \sigma), \delta^{\mathcal{B}}(q_0^{\mathcal{B}}, \sigma))$, как и требовалось.

с. $(n+1)$. Пусть $P(n)$. Докажем $P(n+1)$. Фиксируем $\Sigma^* \ni w: |w| = n+1$. Тогда $w \equiv w_0\sigma$, $|w_0| = n$, $\sigma \in \Sigma$. $\delta^C(q_0^0, w) = \delta^C(q_0^0, w_0\sigma) \equiv \delta^C(\delta^C(q_0^0, w_0), \sigma) \stackrel{P(n)}{=} \delta^C((\delta^A(q_0^A, w_0), \delta^B(q_0^B, w_0)), \sigma) \stackrel{\text{по опр.}}{=} (\delta^A(\delta^A(q_0^A, w_0), \sigma), \delta^B(\delta^B(q_0^B, w_0), \sigma)) \stackrel{\text{сб-во}}{=} (\delta^A(q_0^A, w), \delta^B(q_0^B, w)) \Rightarrow P(n+1)$.

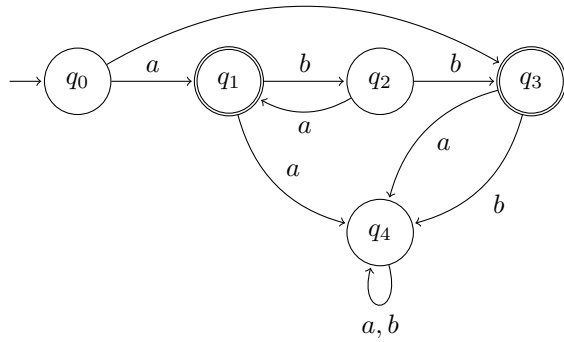
Получаем $w \in L^A \cap L^B \Leftrightarrow w \in L(\mathcal{A}) \cap L(\mathcal{B}) \Leftrightarrow \begin{cases} w \in L(\mathcal{A}) \\ w \in L(\mathcal{B}) \end{cases} \Leftrightarrow \begin{cases} \delta^A(q_0^A, w) \in F^A \\ \delta^B(q_0^B, w) \in F^B \end{cases} \Leftrightarrow (\delta^A(q_0^A, w), \delta^B(q_0^B, w)) \in F^C \stackrel{P(|w|)}{\Leftrightarrow} \delta^C(q_0^0, w) \in F^C \Leftrightarrow w \in L(\mathcal{C}) \blacksquare$

Задача 5

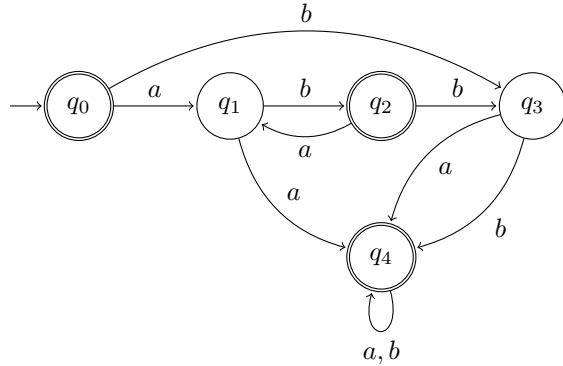
Исходный автомат \mathcal{A} :



Пополним автомат \mathcal{A} до \mathcal{A}' и удалим недостижимые из q_0 состояния: добавим $q_4 \in Q'$, $q_4 \notin F'$, в него направим недостающие переходы:



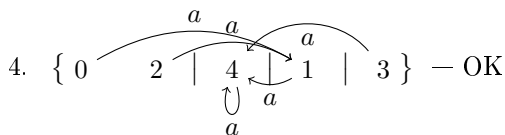
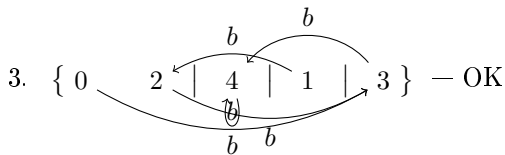
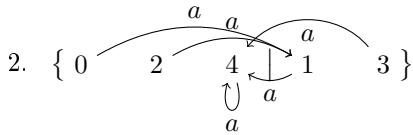
$L(\mathcal{A}') = L(\mathcal{A})$, так как $x \in L(\mathcal{A}) \Rightarrow x \in L(\mathcal{A}')$, потому что $Q \subset Q'$, $F = F'$, $\delta \subset \delta'$. $x \notin L(\mathcal{A}) \Rightarrow$ либо $q_0 \xrightarrow{x} q \notin F$, но тогда $q_0 \xrightarrow{x} q \notin F' \Rightarrow x \notin L(\mathcal{A}')$, либо $\delta(q_0, x) = \emptyset$, тогда $\delta'(q_0, x) = q_4$, потому что был выполнен переход в q_4 , которого не было в \mathcal{A} (по построению, добавлены переходы только в q_4), и при обработке последующих символов \mathcal{A}' остается в q_4 . То есть, в этом случае также $x \notin L(\mathcal{A}')$.



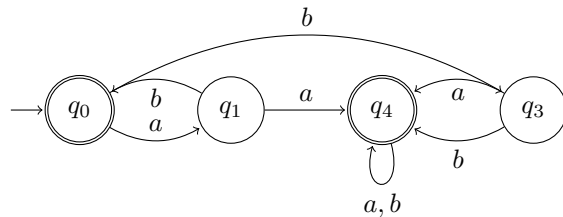
Построим \mathcal{A}'' : $L(\mathcal{A}'') = \overline{L(\mathcal{A}')} \equiv \overline{L(\mathcal{A})}$ по полному автомату \mathcal{A}' , определив $F'' \stackrel{\text{def}}{=} Q' \setminus F'$:

Далее построим по \mathcal{A}'' минимальный \mathcal{A}''' по алгоритму:

1. $\{q_0, q_2, q_4\} \in F''$, $\{q_1, q_3\} \notin F'' \Rightarrow$ они должны быть в разных подмножествах:



5. Склеим q_0 и q_2 .



Задача 6