

Теория и реализация языков программирования.

Задание 5: Регулярные грамматики

Сергей Володин, 272 гр.

задано 2013.10.02

Упражнение 1

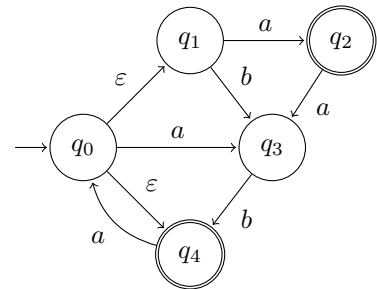
1. Докажем, что $\forall G_1 \text{ — ПРГ } \exists G_2 \text{ — ПГ: } L(G_1) = L(G_2)$: каждая ПРГ является ПГ, так как $T \subset T^*$, $\varepsilon \in T^*$. Поэтому можно взять $G_2 \stackrel{\text{def}}{=} G_1$.
2. Докажем, что $\forall G_2 \text{ — ПГ } \exists G_1 \text{ — ПРГ: } L(G_1) = L(G_2)$:
 1. Построим по G_2 НКА \mathcal{A} : $L(\mathcal{A}) = L(G_2)$ (алгоритм и доказательство в задаче 2).
 2. По НКА \mathcal{A} построим ПРГ G_1 : $L(G_1) = L(\mathcal{A})$ (алгоритм и доказательство в задаче 1).

Это доказывает, что ПРГ и ПГ порождают одно и то же множество языков.

Упражнение 2

Задача 1

Нет, предложенный алгоритм может построить грамматику, которая не будет праволинейной регулярной. Например, для автомата \mathcal{A}_0 из условия переход $q_0 \xrightarrow{\varepsilon} q_1$ по алгоритму должен соответствовать правилу $q_0 \rightarrow \varepsilon q_1$, но это правило не имеет вид $A \rightarrow xB$ ($\varepsilon = x \notin \Sigma$) или $A \rightarrow x$ или $A \rightarrow \varepsilon$.



Далее \mathcal{A} — абстрактный входной автомат

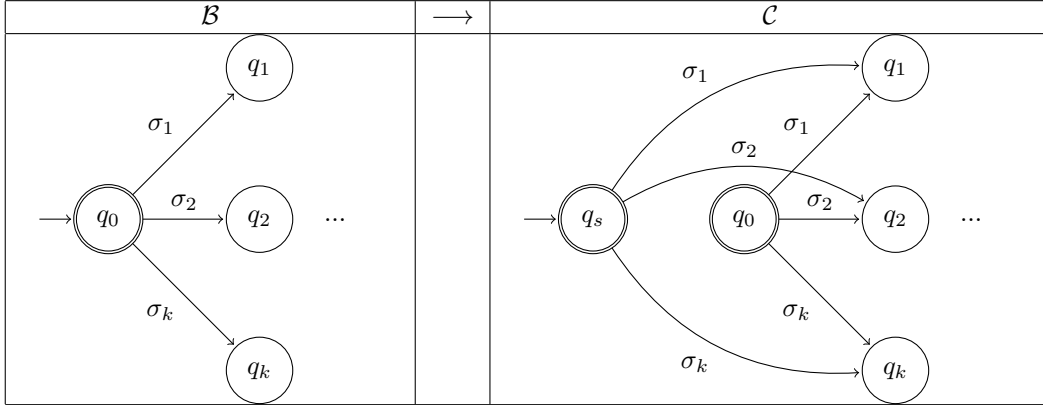
Заметим, что проблему можно решить, преобразовав НКА \mathcal{A} в ДКА \mathcal{B} . Тогда ε -переходов не будет. Остается один случай, в котором $q_0 \in F$, и в q_0 есть переходы из других состояний: $\exists q_1: \delta(q_1, \sigma) = q_0$. Соответствующими правилами были бы $q_0 \rightarrow \varepsilon$, $q_1 \rightarrow \sigma q_0$, которые не подходят для праволинейной регулярной грамматики (аксиома q_0 встречается в правой части при том, что есть переход $q_0 \rightarrow \varepsilon$)

Алгоритм:

- Преобразуем данный НКА \mathcal{A} в ДКА \mathcal{B} . $L(\mathcal{A}) = L(\mathcal{B})$.
- По ДКА $\mathcal{B} = (Q, \Sigma, \delta, q_0, F)$ построим другой ДКА $\mathcal{C} = (Q', \Sigma, \delta', q_s, F')$:

- $Q' \stackrel{\text{def}}{=} Q \cup \{q_s\}$ (добавим состояние q_s)
- $\delta'(q, \sigma) \stackrel{\text{def}}{=} \begin{cases} \delta(q_0, \sigma), & q = q_s \\ \delta(q, \sigma), & \text{иначе} \end{cases}$ (добавим такие же переходы из q_s , как из q_0)
- $F' \stackrel{\text{def}}{=} F \cup (q_0 \in F ? \{q_s\} : \emptyset)$ (если $q_0 \in F$, то сделаем q_s принимающим)

Пример построения:



Докажем, что $L(\mathcal{B}) = L(\mathcal{C})$:

- $L(\mathcal{B}) \subset L(\mathcal{C})$. Пусть $w \in L(\mathcal{B})$.
 - $|w| = 0 \Rightarrow w = \varepsilon \Rightarrow q_0 \in F \Rightarrow q_s \in F' \Rightarrow w \in L(\mathcal{C})$ ■
 - $|w| > 0 \Rightarrow w = \sigma x, \sigma \in \Sigma, x \in \Sigma^*$. Тогда $\delta'(q_s, w) \equiv \delta'(q_s, \sigma x) \equiv \delta'(\delta'(q_s, \sigma), x)$. Обозначим $q_i \stackrel{\text{def}}{=} \delta'(q_s, \sigma) \stackrel{\text{def}}{=} \delta(q_0, \sigma)$. Очевидно, что $q_i \neq q_s$ (иначе получим переход $q_0 \xrightarrow{\sigma} q_s$ в \mathcal{B} , но $q_s \notin F$). Значит, $\delta'(q_i, x) = \delta(q_i, x) \Rightarrow \delta'(q_s, w) \equiv \delta'(q_i, x) \equiv \delta(q_i, x) \equiv \delta(q_0, \sigma x) \equiv \delta(q_0, w) \in F$, т.к. $w \in L(\mathcal{B}) \Rightarrow w \in L(\mathcal{C})$ ■
- $L(\mathcal{C}) \subset L(\mathcal{B})$. Пусть $w \in L(\mathcal{C})$.
 - $|w| = 0 \Rightarrow w = \varepsilon \Rightarrow q_s \in F' \Rightarrow q_0 \in F \Rightarrow \delta(q_0, w) \equiv \delta(q_0, \varepsilon) = q_0 \in F \Rightarrow w \in L(\mathcal{B})$ ■
 - $|w| > 0 \Rightarrow w = \sigma x, \sigma \in \Sigma, x \in \Sigma^*$. $F' \ni \delta'(q_s, w) \equiv \delta'(q_s, \sigma x) \equiv \delta'(\delta'(q_s, \sigma), x)$. Аналогично $q_i \stackrel{\text{def}}{=} \delta'(q_s, \sigma) \equiv \delta(q_0, \sigma) \in Q \Rightarrow \delta'(q_i, x) = \delta(q_i, x)$. Получаем $F' \ni \delta(q_s, w) = \delta(q_i, x) = \delta(q_0, w) \Rightarrow \delta(q_0, w) \in F \Rightarrow w \in L(\mathcal{B})$ ■

- По \mathcal{C} построим ПРГ $G = (Q', \Sigma, P, q_s)$:

$$1. P = \underbrace{\{q_a \longrightarrow \sigma q_b \mid \delta'(q_a, \sigma) = q_b\}}_{(1)} \cup \underbrace{\{q_a \longrightarrow \sigma \mid \delta'(q_a, \sigma) = q_b \in F'\}}_{(2)} \cup \underbrace{\{q_s \in F ? \{q_s \longrightarrow \varepsilon\} : \emptyset\}}_{(3)}.$$

То есть, каждому переходу $q_a \xrightarrow{\sigma} q_b$ в \mathcal{C} соответствует правило $q_a \longrightarrow \sigma q_b \mid \sigma$, причем вторая часть имеется тогда и только тогда, когда $q_b \in F'$. Если q_s — принимающее, то добавляется правило $q_s \longrightarrow \varepsilon$.

- G — ПРГ. Действительно, правила имеют один из видов $(\sigma \in \Sigma) A \longrightarrow \sigma B, A \longrightarrow \sigma$. Поскольку переходов в q_s в \mathcal{C} нет, то аксиома q_s не будет встречаться в правых частях.

Также отметим некоторые свойства правил:

- По построению $\forall P \ni p \equiv (\alpha, \beta) \hookrightarrow |\beta|_{\Sigma} - |\alpha|_{\Sigma} \in \overline{0, 1}$, то есть, количество нетерминальных символов справа либо такое же, как слева, либо на 1 больше. Действительно, для групп правил (1) и (2) эта разница равна 1, а для (3) (если там есть правила) разница равна 0.
- Также по построению \mathcal{C} для всех правил справа нет q_s , так как в противном случае в \mathcal{C} были бы переходы в q_s , что невозможно.
- В правилах слева всегда один нетерминал.
- Если $w_1 \dots w_n \equiv w \in L(G), n > 0, \forall i \in \overline{1, n} \hookrightarrow w_i \in \Sigma$, то есть, $q_s \Longrightarrow^* w_1 \dots w_n$, то оно было получено применением $n - 1$ правил из (1), а затем одного правила из (2), то есть, $q_s \Longrightarrow w_1 q_1 \Longrightarrow w_1 w_2 q_2 \Longrightarrow \dots \Longrightarrow w_1 w_2 \dots w_{n-1} q_{n-1} \Longrightarrow w_1 \dots w_n$.

Действительно,

- Количество нетерминальных символов в конце равно n , значит, было применено n правил из (1) и (2)
- Если первым было применено правило из (3), то количество нетерминалов стало равно 0, и далее ни одно правило не могло быть применено (см. 3(2)c). При этом количество нетерминалов осталось бы равно $0 \neq n$ — противоречие.
- Правило из (3) не могло быть применено и далее, так как тогда бы получили q_s в правой части некоторого (предыдущего по выводу) правила.
- Из 3(2)db и 3(2)dc получаем, что применялись только правила из (1) и (2), а из 3(2)da — что всего таких применений было n .

е. Применение правила из (2) ранее, чем на последнем шаге привело бы к тому, что количество нетерминальных символов стало бы равно 0, после чего (см. 3(2)с) правила бы применяться не могли. Однако, количество нетерминальных символов было бы меньше n — противоречие.

3. Докажем, что $L(G) = L(C)$, т.е. $\forall w \in \Sigma^* \hookrightarrow w \in L(G) \Leftrightarrow w \in L(C)$:

а. $|w| = 0 \Rightarrow w = \varepsilon$

а. $w \in L(C) \Rightarrow q_s \in F \Rightarrow$ правило $q_s \rightarrow \varepsilon \in P$. Значит, $\varepsilon \equiv w \in L(G)$ ■

б. $w \in L(G) \Rightarrow$ правило $q_s \rightarrow \varepsilon \in P$. Тогда $q_s \in F' \Rightarrow w \in L(C)$ ■

б. $n \stackrel{\text{def}}{=} |w| > 0 \Rightarrow w = w_1 \dots w_n, \forall i \in \overline{1, n} \hookrightarrow w_i \in \Sigma$.

а. $w \in L(C) \Rightarrow (q_s, w) \equiv (q_s, w_1 \dots w_n) \vdash (q_1, w_2 \dots w_n) \vdash (q_2, w_3 \dots w_n) \vdash \dots \vdash (q_{n-1}, w_n) \vdash (q_n, \varepsilon), q_n \in F'$. Тогда, по построению G имеем $P \ni q_s \rightarrow w_1 q_1, q_1 \rightarrow w_2 q_2, \dots, q_{n-1} \rightarrow w_n$. Значит, $q_s \rightarrow w_1 q_1 \rightarrow w_1 w_2 q_2 \rightarrow \dots \rightarrow w_1 w_2 \dots w_{n-1} q_{n-1} \rightarrow w_1 \dots w_n \in \Sigma^* \Rightarrow w \in L(G)$ ■

б. $w \in L(G) \stackrel{3(2)d}{\Rightarrow} q_s \rightarrow w_1 q_1 \rightarrow w_1 w_2 q_2 \rightarrow \dots \rightarrow w_1 w_2 \dots w_{n-1} q_{n-1} \rightarrow w_1 \dots w_n$, и были применены правила $q_s \Rightarrow w_1 q_1, \dots, q_{n-1} \Rightarrow w_n$ (также см. 3(2)d).

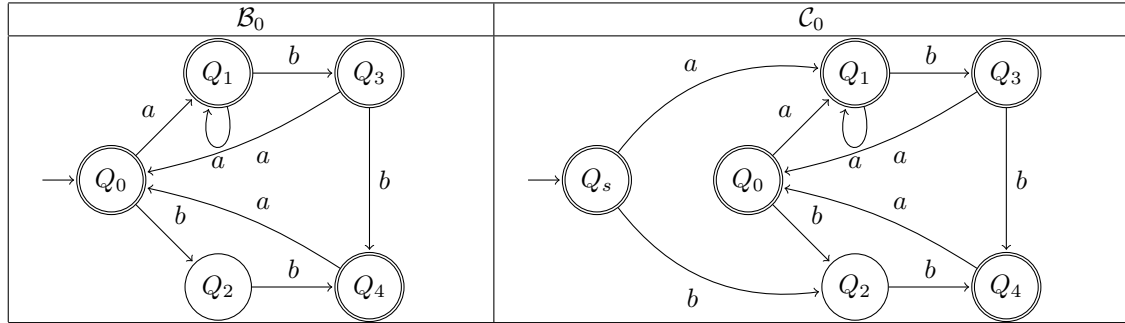
Отсюда $\delta'(q_s, w_1) = q_1, \delta'(q_1, w_2) = q_2, \dots, \delta'(q_{n-1}, w_n) \in F \Rightarrow \delta'(q_s, w) \in F \Rightarrow w \in L(C)$ ■

Применим описанный алгоритм для автомата \mathcal{A}_0 из условия:

1. Построим по \mathcal{A}_0 ДКА \mathcal{B}_0 :



2. Построим \mathcal{C}_0 по \mathcal{B}_0 , как описано в алгоритме:



3. Определим грамматику $G = (\Sigma, Q, P, Q_s)$. Выпишем правила $p \in P$:

Q	ε	a	b
Q_s	$Q_s \rightarrow \varepsilon$	$Q_s \rightarrow aQ_1 a$	$Q_s \rightarrow bQ_2$
Q_0		$Q_0 \rightarrow aQ_1 a$	$Q_0 \rightarrow bQ_2$
Q_1		$Q_1 \rightarrow aQ_1 a$	$Q_1 \rightarrow bQ_3 b$
Q_2			$Q_2 \rightarrow bQ_4 b$
Q_3		$Q_3 \rightarrow aQ_0 a$	$Q_3 \rightarrow bQ_4 b$
Q_4		$Q_4 \rightarrow aQ_0 a$	

Задача 2

1. Пусть дана ПГ $G = (N, T, P, S)$. Построим по ней НКА $\mathcal{A} = (Q, T, \delta, q_0, F): L(\mathcal{A}) = L(G)$.

1. Определим

- $Q = N \cup \{q_f\}$
- $q_0 \stackrel{\text{def}}{=} S$
- $F \stackrel{\text{def}}{=} \{q_f\}$
- $\delta \stackrel{\text{def}}{=} \emptyset$

2. Рассмотрим переходы $P \ni p_i = A \rightarrow xB$ (правила типа 1). Для них добавим

- Состояния $q_1^i, \dots, q_{|x|-1}^i$, если $|x| > 1$.
- Переходы:
 - $A \xrightarrow{\varepsilon} B$, если $x = \varepsilon$.
 - $A \xrightarrow{x_1} B$, если $|x| = 1, x_1 \equiv x$.
 - $A \xrightarrow{x_1} q_1^i \xrightarrow{x_2} q_2^i \rightarrow \dots \rightarrow q_{|x|-1}^i \xrightarrow{x_n} B$, если $|x| > 1, x = x_1 \dots x_n$.

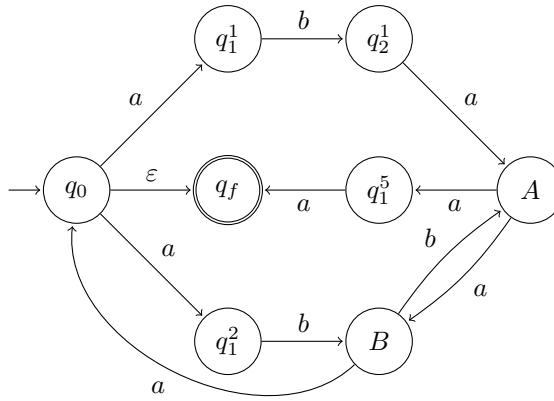
3. Рассмотрим переходы $P \ni p_i = A \rightarrow x$ (правила типа 2). Для них добавим

- Состояния $q_1^i, \dots, q_{|x|-1}^i$, если $|x| > 1$.
- Переходы:
 - $A \xrightarrow{\varepsilon} q_f$, если $x = \varepsilon$.
 - $A \xrightarrow{x_1} q_f$, если $|x| = 1, x_1 \equiv x$.
 - $A \xrightarrow{x_1} q_1^i \xrightarrow{x_2} q_2^i \rightarrow \dots \rightarrow q_{|x|-1}^i \xrightarrow{x_n} q_f$, если $|x| > 1, x = x_1 \dots x_n$.

1.5. Докажем, что $L(G) = L(\mathcal{A})$

- Пусть $w \in L(G), n = |w| > 0, w \equiv w_1 \dots w_n$. Правила типа 1 не изменяют количество нетерминальных символов, а правила типа 2 — уменьшают на 1. Аксиома — один нетерминал. В левой части каждого правила один нетерминал. Поэтому вывод слова w имеет следующий вид: $m - 1$ применений правил типа 1, затем применение правила типа 2 — т.к. после правила (2) нельзя применить никакое правило. Также каждое правило не уменьшает количество нетерминальных символов. То есть, $S = q_0 \rightarrow w_1 w_2 \dots w_{i_1} q_1 \rightarrow w_1 w_2 \dots w_{i_1} \dots w_{i_2} q_2 \rightarrow \dots \rightarrow w_1 \dots w_{i_{m-1}} q_{m-1} \rightarrow w_1 \dots w_n$. По построению имеем $(q_0, w) \vdash^* (q_1, w_{i_1+1} \dots w_n) \vdash^* (q_2, w_{i_2+1} \dots w_n) \vdash^* (q_{m-1}, w_{i_{m-1}+1} \dots w_n) \vdash (q_f, \varepsilon), q_f \in F \Rightarrow w \in L(\mathcal{A})$.
- Пусть $w \in L(G), |w| = 0 \Rightarrow w = \varepsilon$. Тогда во всех m примененных правилах $x_i = \varepsilon$, а последнее — типа 2: $q_0 \rightarrow q_{i_1} \rightarrow \dots \rightarrow q_{i_{m-1}} \varepsilon$. По построению, $\delta(q_0, \varepsilon) = q_{i_1}, \dots, \delta(q_{i_{m-1}}, \varepsilon) = q_f$. Получаем $w \in L(\mathcal{A})$.
- Пусть $w \in L(\mathcal{A})$. Тогда $F \ni \delta(q_0, w) = q_f$ (т.к. $F = \{q_f\}$). Рассмотрим цепочку конфигураций $(q_0, w) \vdash^* (q_f, \varepsilon)$. Выпишем оттуда все состояния $A_i: A_i \in N$. Тогда переходы $A_i \xrightarrow{x_i} A_{i+1}$ между ними были добавлены на шагах 1(2)b, а последний переход $A_m \xrightarrow{x_m} q_f$ — на шаге 1(3)b. Поэтому в грамматике есть правила $S \Rightarrow x_1 A_1, \dots, A_m \Rightarrow x_m$. Отсюда $w \in L(G)$.

2. Построим НКА \mathcal{A} по грамматике $G: S \rightarrow abaA|abB|\varepsilon, A \rightarrow aB|aa, B \rightarrow bA|aS$



- Запишем систему уравнений:
$$\begin{cases} (1) & S = abaA + abB + \varepsilon \\ (2) & A = aB + aa \\ (3) & B = bA + aS \end{cases}.$$
Подставим (3) в (2), получим $A = abA + aaS + aa$. Отсюда

$A = (ab)^*(aaS + aa)$. Подставим в (3): $B = b(ab)^*(aaS + aa) + aS$. Подставим A, B в (1):

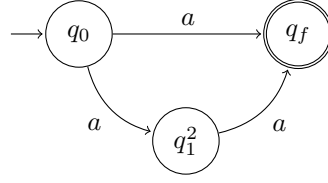
$$\begin{aligned} S &= aba(ab)^*(aaS + aa) + ab(b(ab)^*(aaS + aa) + aS) + \varepsilon \equiv aba(ab)^*aaS + aba(ab)^*aa + abb(ab)^*aaS + abb(ab)^*aa + abaS + \varepsilon \equiv \\ &\equiv (aba(ab)^*aa + abb(ab)^*aa + aba)S + aba(ab)^*aa + abb(ab)^*aa + \varepsilon \Rightarrow \end{aligned}$$

$$S = (aba(ab)^*aa + abb(ab)^*aa + aba)^*(aba(ab)^*aa + abb(ab)^*aa + \varepsilon).$$

- Нет: $S \rightarrow abaA \rightarrow abaaa, S \rightarrow abaA \rightarrow abaaB \rightarrow abaaaS \rightarrow abaaa$.

Задача 3

Для приведенного мной алгоритма это неверно. Возьмем грамматику $G: P = \{S \rightarrow a, S \rightarrow aa\}$. Она будет праволинейной (все правила имеют вид $S \rightarrow x, x \in \Sigma^*$) и однозначной: она порождает язык $\{a, aa\}$, каждое слово может быть получено единственным способом. Но алгоритм построит автомат \mathcal{A} , который не будет детерминированным, так как $\delta(q_0, a) = \{q_f, q_1^2\}$.



Задача 4

Рассмотрим грамматику $G: P = \{S \rightarrow 0S1, S \rightarrow \varepsilon\}$. $L \stackrel{\text{def}}{=} \{0^n 1^n \mid n \geq 0\}$. Докажем, что $w \in L(G) \Leftrightarrow w \in L$.

1. $w \in L \Rightarrow w = 0^n 1^n$

а. $n = 0 \Rightarrow w = \varepsilon$. Но $P \ni S \rightarrow \varepsilon \Rightarrow w \equiv \varepsilon \in L(G)$.

б. $n > 0$. Применим первое правило n раз, после чего применим второе: $S \rightarrow 0S1 \rightarrow 00S11 \rightarrow \dots \rightarrow \underbrace{0\dots 0}_n S \underbrace{1\dots 1}_n \rightarrow 0^n 1^n \Rightarrow w \in L(G)$

2. $w \in L(G)$. Первое правило сохраняет количество нетерминалов, второе уменьшает на 1. Поэтому в выводе сначала n применений первого правила, а затем одно применение второго. Количество терминалов не уменьшается. Вывод имеет вид $S \rightarrow 0S1 \rightarrow \dots \rightarrow \underbrace{0\dots 0}_n S \underbrace{1\dots 1}_n \rightarrow 0^n 1^n \in L$

G — линейная, так как в правых частях правил не более одного нетерминала. Но $L \equiv L(G) \notin \text{REG}$ — было доказано на семинаре. Поэтому получаем, что утверждение $\forall G \text{ — линейная} \Leftrightarrow L(G) \in \text{REG}$ — неверно.

Задача 5

Пусть $\Sigma = \{\sigma_1, \dots, \sigma_n\} \supseteq L, L_{\sigma_1}, \dots, L_{\sigma_n} \in \text{REG}$. Докажем, что подстановка $L_{\sigma_1}, \dots, L_{\sigma_n}$ в L $L' = \{L_{w_1} \dots L_{w_k} \mid w_1 \dots w_k \in L\} \in \text{REG}$ индукцией по $R_3(L)$ — количеству применений третьего пункта определения при получении языка L (см. решение предыдущего задания)

1. $R_3(L) = 0 \Rightarrow$

(а) $L = \emptyset$. Тогда $L' = \emptyset$, так как ни одно слово w не в L . Получаем $L' \in \text{REG}$ ■

(б) $L = \{\varepsilon\}$. Тогда $L' = \varepsilon$ (конкатенация из 0 строк). Получаем $L' \in \text{REG}$ ■

(с) $L = \{\sigma\}$. Тогда $L' = L_\sigma \in \text{REG}$ ■

2. Пусть $\forall L: R_3(L) \leq n \Leftrightarrow L' \in \text{REG}$. Докажем для $n + 1$. Варианты:

(а) $L = XY, X, Y \in \text{REG}$. Тогда подстановка в X $L'_X \in \text{REG}$ по предположению индукции. Аналогично для L'_Y . Но $L' = L'_X L'_Y: L' = \{L_{w_1} \dots L_{w_k} \mid w_1 \dots w_k \in XY\} = \{L_{w_1} \dots L_{w_m} \dots L_{w_k} \mid w_1 \dots w_m \in X, w_{m+1} \dots w_k \in Y\} = \{L_{w_1} \dots L_{w_m} \mid w_1 \dots w_m \in X\} \cdot \{L_{w_{m+1}} \dots L_{w_k} \mid w_{m+1} \dots w_k \in Y\} = L'_X L'_Y$, поэтому $L' \in \text{REG}$.

(б) $L = X|Y, X, Y \in \text{REG}$. Аналогично $L'_X, L'_Y \in \text{REG}$. Но $L' = \{L_{w_1} \dots L_{w_k} \mid w_1 \dots w_k \in X|Y\} = \{L_{w_1} \dots L_{w_k} \mid w_1 \dots w_k \in X\} \cup \{L_{w_1} \dots L_{w_k} \mid w_1 \dots w_k \in Y\} = L'_X L'_Y \in \text{REG}$ ■

(с) $L = X^*, X \in \text{REG}$. По предположению, $L'_X \in \text{REG}$. Но $L' = \{L_{w_1} \dots L_{w_k} \mid w_1 \dots w_k \in X^*\} = \{(L_{w_1} \dots L_{w_k})^* \mid w_1 \dots w_k \in X\} = L'^*_X \in \text{REG}$ ■

Задача 6

Задача 7