

## Упражнение 1

1. В алфавите  $\Sigma = \{a, b, c\}$  постройте грамматику для языка  $L = \{\omega \in \Sigma^* \mid \omega \text{ содержит подстроку } aa\}$

$$\begin{aligned} S &\rightarrow cS \mid bS \mid aaS_1 \mid aS \\ S_1 &\rightarrow aS_1 \mid bS_1 \mid cS_1 \mid \lambda \end{aligned}$$

2. В алфавите  $\Sigma = \{a, b, c\}$  постройте грамматику для языка  $L = \{\omega \in \Sigma^* \mid \omega \text{ не полиндром}\}$

$$\begin{aligned} S &\rightarrow aS_1 \mid bS_2 \mid cS_3 \\ S_1 &\rightarrow S_4b \mid S_4c \mid Sa \\ S_2 &\rightarrow S_4a \mid S_4c \mid Sb \\ S_3 &\rightarrow S_4a \mid S_4b \mid Sc \\ S_4 &\rightarrow aS_1 \mid bS_2 \mid cS_3 \mid \lambda \end{aligned}$$

3. В алфавите  $\Sigma = \{\emptyset, \mathbb{N}, '\{', '\}', ',', \cup\}$  постройте грамматику для языка  $L = \{\omega \in \Sigma^* \mid \omega \text{ — синтаксически корректная строка обозначающая множество}\}$

$$\begin{aligned} S &\rightarrow \emptyset \mid \mathbb{N} \mid \{S_1\} \mid \cup \\ S_1 &\rightarrow S, S_2 \mid S \mid \lambda \\ S_2 &\rightarrow S, S_2 \mid S \end{aligned}$$

## Упражнение 2

В алфавите  $\Sigma = \{1, +, =\}$  Рассмотрим язык  $A = \{1^m + 1^n = 1^{n+m} \mid m, n \in \mathbb{N}\}$

1. Докажите что язык  $A$  регулярный (построением) или не регулярный (через лемму о накачке)

Фиксируем  $\forall n = N$  возьмём слово  $\alpha = 1^N + 1^N = 1^{2N}$ ,  $\alpha \in A$ ,  $|\omega| = 4N+2 > N$ , так как в лемме  $|xy| \leq N$ , как бы мы не взяли  $xy$ ,  $y = 1^i \implies xy^kz$  выходит за пределы языка при  $\forall k \neq 1$ , так как тогда в выражении перестаёт выполняться тождество  $\implies$  язык  $A$  не регулярный

2. Постройте КС-грамматику для языка  $A$ , показывающую, что  $A$  — контекстно-свободный

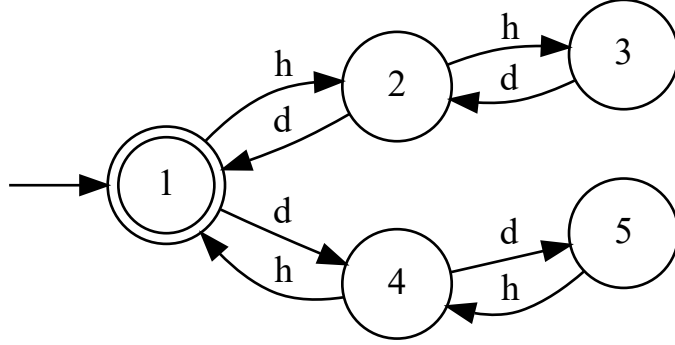
$$\begin{aligned} S &\rightarrow 1S1 \mid +S_1 \\ S_1 &\rightarrow 1S_11 \mid = \end{aligned}$$

## Упражнение 3

1. Вы пошли гулять с собакой, ваша собака на поводке длины 2. Это значит что она не может отойти от вас более чем на 2 шага. Пусть  $\Sigma = \{h, d\}$ , где  $h$  — ваше перемещение на один шаг вперёд, а  $d$  — шаг собаки. Прогулка может быть завершена, если собака и человек оказались в одной точке.

Пусть  $D_1 = \{\omega \in \Sigma^* \mid \omega \text{ описывает последовательность ваших шагов и шагов вашей собаки на прогулке с поводком}\}$

- (а) Докажите, что язык  $D_1$  регулярный (построением) или не регулярный (через лемму о накачке)



- (b) Постройте КС-грамматику для языка  $D_1$ , показывающую, что  $D_1$  — контекстно-свободный

$$\begin{aligned}
 S &\rightarrow hS_1 \mid dS_2 \mid \lambda \\
 S_1 &\rightarrow hdS_1 \mid dS \\
 S_2 &\rightarrow dhS_2 \mid hS
 \end{aligned}$$

2. Допустим теперь, что вы также пошли на прогулку с собакой. но не взяли с собой поводок. Это значит, что вы можете отдалиться от собаки на любое расстояние.

Пусть  $D_2 = \{\omega \in \Sigma^* \mid \omega \text{ описывает последовательность ваших шагов и шагов вашей собаки на прогулке без поводка}\}$

- (a) Докажите, что язык  $D_2$  регулярный(построением) или не регулярный (через лемму о накачке)

Фиксируем  $\forall n = N$  возьмём слово  $\alpha = h^N d^N$ ,  $\alpha \in D_2$ ,  $|\omega| = 2N > N$ , так как в лемме  $|xy| \leq N$ , как бы мы не взяли  $xy$ ,  $y = h^i \implies xy^k z$  выходит за пределы языка при  $\forall k \neq 1$ , так как тогда в конце прогулки вы и собака окажитесь в разных местах  $\implies$  язык  $D_2$  не регулярный

- (b) Постройте КС-грамматику для языка  $D_2$ , показывающую, что  $D_2$  — контекстно-свободный

$$S \rightarrow dShS \mid hSdS \mid \lambda$$

## Упражнение 4

Пусть  $Perm(\omega)$  — это множество всех пермутаций строки  $\omega$ , то есть, множество всех уникальных строк, состоящих из тех же букв и в том же количестве, что и в  $\omega$ . Если  $L$  — регулярный язык, то  $Perm(L)$  — это объединение  $Perm(\omega)$  для всех  $\omega$  в  $L$ . Если  $L$  регулярный, то  $Perm(L)$  иногда тоже регулярный, иногда контекстно-свободный, но не регулярный, а иногда даже не контекстно-свободный. Рассмотрите следующие регулярные выражения  $R$  и установите, является ли  $Perm(R)$  регулярным, контекстно-свободным или ни тем и ни другим:

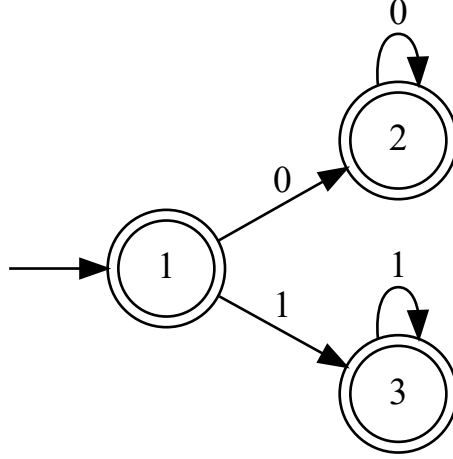
### 1. $(01)^*$

- (a) Фиксируем  $\forall n = N$  возьмём слово  $\alpha = 0^N 1^N$ ,  $\alpha \in Perm(R)$ ,  $|\omega| = 2N > N$ , так как в лемме  $|xy| \leq N$ , как бы мы не взяли  $xy$ ,  $y = 0^i \implies xy^k z$  выходит за пределы языка при  $\forall k \neq 1$ , так как тогда количество нулей  $\neq$  количество единиц  $\implies \alpha$  уже не является перестановкой символов в слове состоящем из пар нулей и единиц  $\implies$  язык  $Perm(R)$  не регулярный

(b)  $S \rightarrow 0S1S \mid 1S0S \mid \lambda$

### 2. $0^* + 1^*$

- (a) Построим автомат



- (b)  $S \rightarrow 0S_1 \mid 1S_2 \mid \lambda$   
 $S_1 \rightarrow 0S_1 \mid \lambda$   
 $S_2 \rightarrow 1S_2 \mid \lambda$

### 3. $(012)^*$

- (a) Фиксируем  $\forall n = N$  возьмём слово  $\alpha = 0^N 1^N 2^N$ ,  $\alpha \in Perm(R)$ ,  $|\omega| = 3N > N$ , так как в лемме  $|xy| \leq N$ , как бы мы не взяли  $xy$ ,  $y = 0^i \implies xy^kz$  выходит за пределы языка при  $\forall k \neq 1$ , так как тогда количество нулей  $\neq$  количество единиц и двоек  $\implies \alpha$  уже не является перестановкой символов в слове состоящем из троек нулей, единиц и двоек  $\implies$  язык  $Perm(R)$  не регулярный
- (b) Фиксируем  $\forall p = N$  возьмём слово  $\alpha = 0^N 1^N 2^N$ ,  $\alpha \in Perm(R)$ ,  $|\omega| = 3N > N$ , так как в лемме  $|v\omega x| \leq N$ , как бы мы не взяли  $v\omega x$ , оно никогда не сможет включать в себя и единицы и двойки и нули, а значит если начать что то наращивать в этом слове, тех цифр что в нем не присутствуют станет меньше чем других  $\implies uv^\omega x^k y$  выходит за пределы языка

при  $\forall k \neq 1, \implies$  язык  $Perm(R)$  не контекстно-свободный

## Упражнение 5

1. Пусть грамматика  $G$  — праволинейная. Опишите алгоритм построения НКА  $N$ , такого что  $(N) = (G)$ . Коротко докажите (от противного), что ваш алгоритм может получить только слова из языка грамматики. Проиллюстрируйте алгоритм на грамматике:

$$\begin{aligned} A &\rightarrow aB \mid bC \\ B &\rightarrow aB \mid \lambda \\ C &\rightarrow aD \mid A \mid bC \\ B &\rightarrow aD \mid bD \mid \lambda \end{aligned}$$

Имеем  $G = \langle V, \Sigma, R, S \rangle$ , где

$V$  — множество нетерминальных символов

$\Sigma$  — множество терминальных символов

$R$  — правила вида  $\begin{cases} 1. A \rightarrow \lambda \\ 2. A \rightarrow B \\ 3. A \rightarrow aB \end{cases}$

$S$  — стартовый нетерминал

Тогда НКА  $N = \langle M, Q, y, T, \sigma \rangle$  будет таким:

Мн-во терминальных символов:  $M = \Sigma$

Мн-во вершин:  $Q = V$

Стартовая вершина:  $y = s$

Мн-во конечных вершин:  $T = \{t : t \in V, \exists \omega \in R : t \rightarrow \lambda \in \Sigma\}$

$\omega\}$

Мн-во переходов  $\sigma$  формируем по принципу (относительно номеров из  $R$ ):

$$\begin{cases} 1. \text{ничего} \\ 2. \lambda - \text{переход из } A \text{ в } B \\ 3. \text{переход по } a \text{ из } A \text{ в } B \end{cases}$$

Докажем  $(N) = (G)$

Пусть  $x$  не выводится из  $G$ , но выводится из  $N$

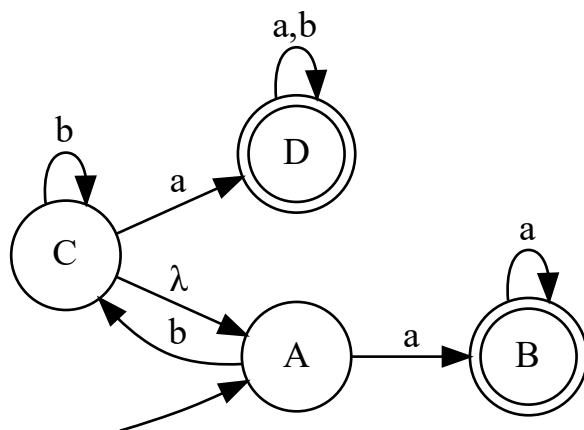
Пусть  $x = a_1 a_2 \dots a_n$  ( $a_i \in \Sigma, \forall i$ )

При разборе слова автоматом, для каждой буквы  $a_i$ , мы должны пройти по  $\lambda$ -переходам из текущей вершины  $S$  в вершину  $S_1$  из которой есть переход по букве  $a_i$ , когда буквы закончатся мы аналогично должны дойти до какой то конечной вершины по  $\lambda$ -переходам. Так как в  $N$  каждой вершине мы можем поставить в соответствие правило из  $G$  (т.к.  $Q = V$ ) а  $\lambda$ -переходы между вершинами  $A$  и  $B$  мы ставили тогда когда в  $R$  было правило  $A \rightarrow B$  то выходит везде где по  $\lambda$ -переходам мы добираемся из вершины  $S$  в вершину  $S_1$ , в  $G$   $S \Rightarrow^* S_1$ , далее так как в  $N$  мы ставили переход из  $A$  в  $B$  по  $a$  тогда когда в  $R$  было правило  $A \rightarrow aB$ , значит если в автомате есть переход из  $S_1$  в  $S_2$  по  $a_i$ , то в  $G$   $S_1 \Rightarrow^* a_i S_2$ , и наконец так как мы в  $N$  обозначали вершину  $A$  конечной если в  $R$  есть правило  $A \rightarrow \lambda$ , то если в автомате мы смогли добраться по  $\lambda$ -переходам из  $S_m$  в вершину  $S_k$  которая является конечной, значит в  $G$   $S_m \Rightarrow^* S_k$ ,  $S_k \Rightarrow^* \lambda \implies S_m \Rightarrow^* \lambda \implies G \Rightarrow^* x$  — противоречие.

Пусть теперь  $x$  выводится из  $G$ , но не выводится из  $N$

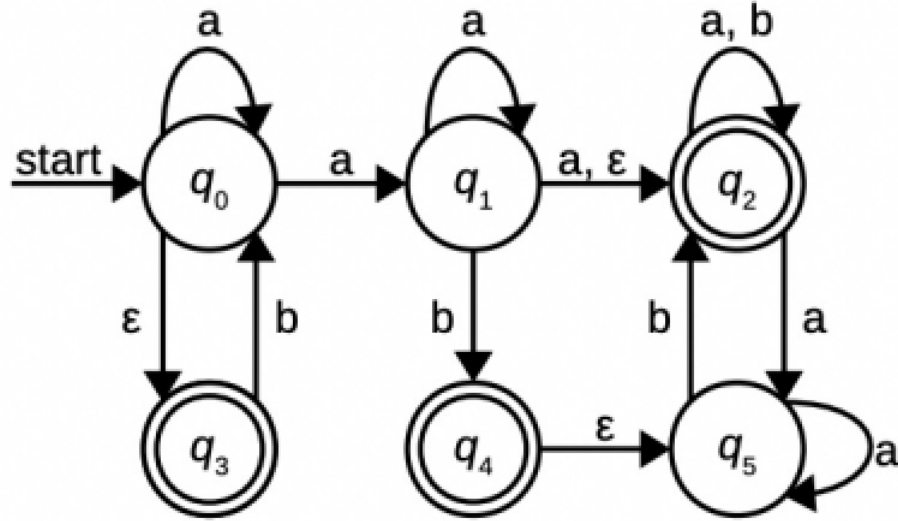
Пусть  $x = a_1 a_2 \dots a_n$  ( $a_i \in \Sigma, \forall i$ )

Доказывать будем аналогично. Чтобы  $y \Rightarrow^* a_1 S_1$  нужно пройти по переходам вида  $A \rightarrow B$  из  $y$  в  $S$ : есть переход  $S \rightarrow a_1 S_1$ , в  $N$  можно сделать тоже-самое перейдя по  $\lambda$ -переходам из  $y$  в  $S$ , а далее из  $S$  в  $S_1$  по  $a_1$ , из за принципа построения  $N$ . Аналогично происходит до конца слова, когда закончатся буквы, допустим на  $S_n$ , далее  $S_n \Rightarrow^* \lambda$ , по тем же соображениям мы и в  $N$  можем сделать так-же и прийти из  $S_n$  в конечную вершину, выходит что если слово выводится из  $G$  оно выводится и из  $N$ .



2. Пусть  $N$  — НКА. Опишите алгоритм построения КС-грамматики  $G$ , такой что  $(G) = (N)$ . Коротко докажите (от противного), что ваш алгоритм может получить только слова из языка НКА. Проиллюстрируйте алгоритм на автомате:





Имеем  $N = \langle M, Q, y, T, \sigma \rangle$ , где  
 $M$  – множество терминальных символов  
 $Q$  – множество вершин  
 $y$  – стартовая вершина  
 $T$  – множество конечных вершин  
 $\sigma$  – множество переходов

Тогда грамматика  $G = \langle V, \Sigma, R, S \rangle$  будет такой:

Мн-во нетерминальных символов:  $V = Q$

Мн-во терминальных символов:  $\Sigma = M$

Стартовый нетерминал:  $S = y$

Мн-во правил вывода  $R$  сформируем

по принципу: 
$$\begin{cases} A \rightarrow B \text{ (есть } \lambda \text{ – переход из } A \text{ в } B) \\ A \rightarrow aB \text{ (есть переход по } a \text{ из } A \text{ в } B) \\ A \rightarrow \lambda \text{ (} A \text{ – терминальная вершина)} \end{cases}$$

Докажем  $(N) = (G)$

Пусть  $x$  не выводится из  $G$ , но выводится из  $N$

Пусть  $x = a_1 a_2 \dots a_n$  ( $a_i \in \Sigma, \forall i$ )

При разборе слова автоматом, для каждой буквы  $a_i$ , мы должны пройти по  $\lambda$ -переходам из текущей вершины  $S$  в

вершину  $S_1$  из которой есть переход по букве  $a_i$ , когда буквы закончатся мы аналогично должны дойти до какой то конечной вершины по  $\lambda$ -переходам. Так как в  $N$  каждой вершине мы можем поставить в соответствие правило из  $G$  (т.к.  $V = M$ ) и так как мы создавали правила вида  $A \rightarrow B$  когда есть  $\lambda$ -переход из  $A$  в  $B$ ,  $S \Rightarrow^* S_1$ , так как мы создавали правила вида  $A \rightarrow aB$  когда из вершины  $A$  есть переход в  $B$  по  $a$ , то  $S_1 \Rightarrow^* a_i S_2$ , аналогично со всеми переходами, теперь если мы можем из вершины  $S_n$  перейти в конечную вершину значит  $S_n \Rightarrow^* \lambda$ , значит  $x$  выводится из  $G$  – противоречие.

Пусть теперь  $x$  выводится из  $G$ , но не выводится из  $N$   
Пусть  $x = a_1 a_2 \dots a_n$  ( $a_i \in \Sigma, \forall i$ )

Доказывать будем аналогично. Чтобы  $y \Rightarrow^* a_1 S_1$  нужно пройти по переходам вида  $A \rightarrow B$  из  $y$  в  $S$ : есть переход  $S \rightarrow a_1 S_1$ , в  $N$  можно сделать тоже-самое, так как для каждого подобного правила есть  $\lambda$ -переход, перейдя по  $\lambda$ -переходам из  $y$  в  $S$ , а далее из  $S$  в  $S_1$  по  $a_1$ , из за принципа построения  $N$ . Аналогично происходит до конца слова, когда закончатся буквы, допустим на  $S_n$ , далее  $S_n \Rightarrow^* \lambda$ , по тем же соображениям мы и в  $N$  можем сделать так-же и прийти из  $S_n$  в конечную вершину, выходит что если слово выводится из  $G$  оно выводится и из  $N$ .

$$\begin{aligned}
q_0 &\rightarrow aq_0 \mid aq_1 \mid q_3 \\
q_1 &\rightarrow aq_1 \mid bq_4 \mid aq_2 \mid q_2 \\
q_2 &\rightarrow aq_2 \mid bq_2 \mid aq_5 \\
q_3 &\rightarrow bq_0 \mid \lambda \\
q_4 &\rightarrow q_5 \mid \lambda \\
q_5 &\rightarrow aq_5 \mid bq_2
\end{aligned}$$