ТРЯП. Домашнее задание № 8

Шарапов Денис, Б05-005

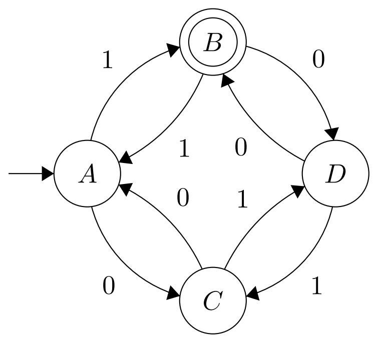
Задача 1

L — язык, состоящий из всех слов в алфавите $\{0, 1\}$, которые содержат чётное число нулей и нечётное число единиц. Выполнить следующие задания:

1. Построить эквивалентную праволинейную грамматику. Будет ли она однозначной?
2. Построить регулярное выражение для языка L^R.

Решение.

1. Построим автомат, распознающий язык всех слов, которые содержат чётное число нулей. Далее построим автомат, распознающий язык всех слов, которые содержат нечётное число единиц. После чего пересечём эти автоматы и получим следующий автомат (задача 5 из домашнего задания № 2):



Теперь воспользуемся алгоритмом перевода НКА → ПГ. Пусть G — искомая праволинейная грамматика. Обозначим $G = (N, T, P, S)$, где:

- $N = \{A, B, C, D\}$ — множество нетерминалов;
- $S = A$ — начальный символ грамматики;
- $T = \{0, 1\}$ — алфавит;
- P — множество правил вывода.

Алгоритм:

(a) $(A \to \sigma B) \in P$, если $\delta(A, \sigma) = \{B\},\ \sigma \in T$;

(b) $(A \to \sigma) \in P$, если $\delta(A, \sigma) = \{B\},\ \sigma \in T$ и $B \in F$ — принимающее;

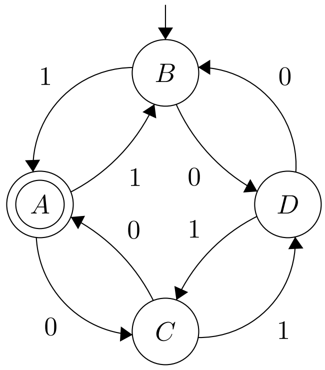
(c) $(S \to \varepsilon) \in P$, если $q_0 \in F$.

Тогда грамматика задаётся правилами:

$$A \to 0C \mid 1,$$
$$B \to 0D \mid 1A,$$
$$C \to 0A \mid 1D,$$
$$D \to 0 \mid 1C.$$

Алгоритм построения грамматики строит G буквально по переходам в автомате. Заметим, что автомат — детерминированный конечный, т. е. в нём каждый переход определён однозначно (с помощью перехода можно получить конкретное слово и только его). Поэтому данная грамматика является однозначной.

2. Построим РВ для языка L^R. Для этого воспользуемся алгоритмом перевода автомата, распознающего язык L, в автомат, распознающий язык L^R:



Составляем систему:

$$\begin{cases} R_A = 0R_C + 1R_B + \varepsilon, \\ R_B = 0R_D + 1R_A, \\ R_C = 0R_A + 1R_D, \\ R_D = 0R_B + 1R_C. \end{cases} \quad \begin{cases} R_A = 0(0R_A + 1R_D) + 1(0R_D + 1R_A) + \varepsilon, \\ R_B = 0R_D + 1R_A, \\ R_C = 0R_A + 1R_D, \\ R_D = 0R_B + 1R_C. \end{cases}$$

$$\begin{cases} R_A = (00+11)^*((01+10)R_D + \varepsilon), \\ R_B = 0R_D + 1R_A, \\ R_C = 0R_A + 1R_D, \\ R_D = 0R_B + 1R_C. \end{cases} \quad \begin{cases} R_A = (00+11)^*((01+10)R_D + \varepsilon), \\ R_B = 0R_D + 1R_A, \\ R_C = 0R_A + 1R_D, \\ R_D = 0(0R_D + 1R_A) + 1(0R_A + 1R_D). \end{cases}$$

$$\begin{cases} R_A = (00+11)^*((01+10)R_D + \varepsilon), \\ R_B = 0R_D + 1R_A, \\ R_C = 0R_A + 1R_D, \\ R_D = (00+11+(01+10)(00+11)^*(01+10))^*(01(00+11)^* + 10(00+11)^*). \end{cases}$$

Откуда и получаем искомое РВ. □

Задача 2

Покажите индукцией по длине слова, что КС-грамматика с правилами

$$S \to SS \mid aSb \mid bSa \mid \varepsilon$$

порождает язык всех слов с равным числом символов a и b.

Решение.

Пусть L — язык всех слов с равным числов символов a и b, $L(G)$ — язык, порождаемый грамматикой G из условия.

Утверждение. $L(G) \subseteq L$.

Доказательство. Выберем произвольное слово $w \in L(G)$. Докажем по индукции числа n букв в слове w.

База индукции:

1) $n = 0 : w = \varepsilon \in L$, выведенное из правила $S \to \varepsilon$

2) $n = 2$: $w = ab \in L$ или $w = ba \in L$, выведенные цепочками $S \to aSb \to ab$ или $S \to bSa \to ba$ соответственно.

Далее будем использовать натуральную константу $k \geq 2$.

Случай 1. Индукция для правила вывода $S \to aSb$.

Предположение индукции $n = 2k$:

Пусть существует цепочка, выводящая слово $w \in L$ длиной $|w| = n$ применением правила $S \to \varepsilon$.

Переход индукции $n = 2k + 2$:

Применим k раз правило вывода $S \to aSb$. По предположению индукции найдётся цепочка (просто предъявим её) такая, что из неё можно вывести слово длиной $n = 2k$:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow \ldots \Rightarrow a^k S b^k.$$

Применяем выбранное правило ещё раз:

$$S \Rightarrow^* a^{k+1} S b^{k+1} \Rightarrow a^{k+1} b^{k+1} \in L.$$

Случай 2. Индукция для правила вывода $S \to bSa$ — аналогично.

Случай 3. Индукция для комбинации правил вывода $S \to aSb$ и $S \to bSa$.

Предположение индукции $n = 2k$:

Пусть существует цепочка, выводящая слово $w \in L$ длиной $|w| = 2k$ применением правила $S \to \varepsilon$.

Переход индукции $n = 2k + 2$:

Воспользуемся уже доказанными утверждениями (случаи 1 и 2). Пусть без ограничения общности правила были применены в таком порядке (для другого порядка рассуждения аналогичны): сначала $m_1 \geq 1$ раз применили правило $S \to aSb$, затем $m_2 \geq 1$ раз применили правило $S \to bSa$, после чего $m_3 \geq 0$ раз применили правило $S \to aSb$ и т. д. по индукции числа чередований правил. Случай 1 утверждает, что после применения $m_1 \geq 1$ раз первого правила выводится слово $v \in L$. Далее второе правило $S \to bSa$ изменит количество символов a и b в слове v (текущее на данный момент) на одинаковое количество. Далее применяем случай 2 и по индукции числа чередований правил получим слово $w \in L$.

Случай 4. Индукция для правила вывода $S \to SS$.

Предположение индукции $n = 2^k$:

Пусть существует цепочка, выводящая слово $w \in L$ длиной $|w| = 2^k$ применением правила $S \to aSb \mid bSa \mid \varepsilon$.

Переход индукции $n = 2^{k+1}$:

Применим k раз правило вывода $S \to SS$:

$$S \Rightarrow SS \Rightarrow SSSS \Rightarrow^* S^{2^k}.$$

Делаем ещё один переход:

$$S \Rightarrow^* S^{2^{k+1}}.$$

По предложению индукции найдётся цепочка (предъявим её) выводящая слово $w \in L$ длиной $|w|_n$: используя доказанные случаи 1, 2, 3, 4, подставляем всевозможные комбинации правил 1 и 2 в вывод

$$S \Rightarrow^* S^{2^k}$$

и получаем слово длиной 2^k при последней замене $S \to \varepsilon$. Теперь представим, что последней замены не было. Применяем правило $S \to SS$, после чего применяем комбинации правил 1 и 2, затем применяем правило $S \to \varepsilon$ и получаем слово длиной $n = 2^{k+1}$, т. к. при замене каждого терминального символа количество букв в слово изменяется ровно на единицу («удаляется» S, подставляется ab или ba), а при последнем переходе индукции было использовано правило $S \to SS$, которое дало четное число терминалов S.

Случай 5. Произвольная комбинация правил $S \to SS \mid aSb \mid bSa \mid \varepsilon$. Используются доказанные случаи 1, 2, 3, 4. □

Утверждение. $L \subseteq L(G)$.

Доказательство. Требуется доказать утверждение

$$\forall w \in L \hookrightarrow w \in L(G).$$

Докажем по индукции числа n букв в слове в $w \in L$.

База индукции:

1) $n = 0 : w = \varepsilon \in L(G)$, выведенное из правила $S \to \varepsilon$

2) $n = 2$: $w = ab \in L(G)$ или $w = ba \in L(G)$, выведенные цепочками $S \to aSb \to ab$ или $S \to bSa \to ba$ соответственно.

Далее будем использовать натуральную константу $k \geq 2$.

Предположение индукции $n = 2k$: пусть верно утверждение

$$\forall w \in L : |w| = 2k \hookrightarrow w \in L(G).$$

Переход индукции $n = 2k + 2$:

Используем утверждение индукции. Если произвольное слово $w \in L$ длиной $|w| = 2k$ лежит в языке $L(G)$, то его можно вывести последовательным применением правил. Пусть в цепочке вывода было сделано $m - 1$ шагов так, что если сделать ещё один шаг $(m - 1) \to m$, то можно вывести слово w. Тогда на шаге m применим одно из правил вывода (кроме $S \to \varepsilon$). Т. к. мы рассматриваем произвольное слово $w \in L$, то будут учтены всевозможные комбинации подстановок правил (потому что в предположении индукции можно вывести произвольное слово из L длины $2k$). После применения одного из правил и дальшейшего применения правила $S \to \varepsilon$ получим слово длиной $n = 2k+2, 2k+4, 2k+6, 2k+8 \ldots$ в зависимости от предыстории (например, в выводе была длинная последовательность SS, расмотренная в прошлом утверждении). Используя доказанные случаи прошлого утверждения, получаем, что произвольное слово, выбранное в предположении индукции, не могло покинуть множество L после применения правил. Переход индукции доказан. □

Замечение:

Утверждение, данное в условии задачи, удобнее доказывать следующим образом: доказать сначала по индукции (того, что удобнее выбрать) включение $L \subseteq L(G)$, а затем, используя уже доказанное утверждение, по контрапозиции доказать обратное включение.

Задача 3

1. Показать, что язык палиндромов в произвольном алфавите ялвяется КС-языком.

2. Показать, что дополнение к языку палиндромов также является КС-языком.

Решение.

1. Покажем, что язык палиндромов в произвольном алфавите является КС-языком. Для этого достаточно привести грамматику $G = (N, T, P, S)$ в качестве примера. Пусть

   - $N = \{S\}$ — множество нетерминалов;
   - $T = \{a_i\}_{i=1}^{I}$ — конечный заданный алфавит;

- $P = \{S \to a_i,\ S \to a_iSa_i\ S \to \varepsilon\} \quad \forall i \in \overline{1, I}$.

Тогда G порождает язык PAL.

Следующие утверждения будем доказывать для $I = 2$ (для краткости). Тогда грамматика G принимает вид

- $N = \{S\}$;
- $T = \{a, b\}$;
- $P : S \to aSa \mid bSb \mid a \mid b \mid \varepsilon$.

Утверждение. $L(G) \subseteq \mathrm{PAL}$.

Доказательство. Докажем по индукции числа n букв в произвольно выбранном слове $w \in L(G)$.

База индукции:

1) $n = 0$: $w = \varepsilon \in \mathrm{PAL}$;
2) $n = 1$: $w = a \in \mathrm{PAL}$ или $w = b \in \mathrm{PAL}$;
3) $n = 2$: $w = aa \in \mathrm{PAL}$ или $w = bb \in \mathrm{PAL}$;

Далее используем натуральную константу $k > 2$.

Предположение индукции $n = k$:

Пусть верно утверждение

$$\forall w \in L(G) :\ |w| = k \hookrightarrow w \in \mathrm{PAL}.$$

Переход индукции $n = k + 1$:

Рассмотрим произвольное слово $w \in L(G)$ длиной $|w| = k$. Пусть без ограничения общности на предпоследнем шаге вывод слова w имеет вид

$$S \Rightarrow^* ava,$$

где v содержит в себе последовательность применения правил $S \to aSa$ или $S \to bSb$. Далее применяется одно из правил, не содержащих справа терминала и по предположению индукции получается слово $w \in \mathrm{PAL}$. Представим, что последнего перехода не было. Из рассматриваемого предпоследнего шага делаем переход по одному из правил, содержащих справа терминал. Затем делаем переход по правилу, не содержащему справа терминала (a или b), и в силу того, что по предположению индукции на предыдущем шаге был получен палиндром, переход по данному правилу выведет слово $u \in \mathrm{PAL}$.

Переход индукции $n = k + 2$:

Аналогичные рассуждения, но только последние два перехода имеют вид $S \to aSa \to aa$. □

Утверждение. $\mathrm{PAL} \subseteq L(G)$.

Доказательство. Приведём конструктивное доказательство. Построим грамматики G' и G'' по произвольным словам из языка PAL. Для это рассмотрим два случая.

Случай 1. Рассмотрим произвольное слово $w \in \mathrm{PAL}$ нечётной длины. Тогда будем действовать следующему алгоритму:

(a) Смотрим на первый и последний символы слова w (они совпадают) и добавляем их в определенный «класс», содержащий терминальный символ (например, слово $abbba$ добавляем в класс aSa, затем bbb в класс bSb и т. д.).

(b) Повторяем процедуру, пока не получим на конце единственный символ (например, $abbba$ — bbb — b).

Получили некоторый «разбор» слова w, который соответствует правилам

$$S \to aSa \mid bSb \mid a \mid b$$

грамматики G'.

Случай 2. Рассмотрим произвольное слово $w \in \mathrm{PAL}$ чётной длины. Строим аналогичный алгоритм, как в случае 1, но только в пункте (b) получим на конце пустое слово. Тогда правила грамматики G'' имеют вид

$$S \to aSa \mid bSb \mid \varepsilon.$$

«Объединяя» G' и G'', получим грамматику G. Утверждение доказано. □

Осталось заменить в рассуждениях символы a и b на символы a_i и получить утверждение для произвольного конечного алфавита.

2. Пусть R — дополнение языка палиндромов. Приведём КС-грамматику G для произвольного конечного алфавита T:

   - $N = \{S, A\}$;
   - $T = \{a_i\}_{i=2}^{I}$;
   - S — начальный терминал;
   - P :

$$S \to a_iSa_i \mid a_iBa_j \quad \forall i, j \in \overline{2, I} : \ i \neq j,$$
$$B \to a_iBa_j \mid a_i \mid \varepsilon \quad \forall i, j \in \overline{2, I} : \ i \neq j.$$

   Утверждение. $L(G) \subseteq R$.

   Доказательство. Рассмотрим вывод произвольного слова $w \in L(G)$. Слово w на концах имеет либо разные символы (правило $S \to a_iBa_j$), либо одинаковые (правило $S \to a_iSa_i$), но в выводе обязательно встретится последовательность символов, которая читается с конца и с начала по-разному, т. к. если слово w было выведено, то существует конечная последовательность применения правил вывода (цепочка), следовательно должны применяться правила $B \to a_iBa_j \mid a_i \mid \varepsilon$, которые и нарушают палиндромность. □

   Утверждение. $R \subseteq L(G)$.

   Доказательство. Рассмотрим произвольное слово $w \in R$. Докажем по индукции n длины вывода слова w.

   База индукции $n = 2$:

$$w = a_ia_ia_j \in L(G) \ \text{ или } \ a_ia_j \in L(G), \text{ где } i \neq j$$

   Предположение индукции $n = k$ ($k > 2$ — натуральное):

   Пусть существует вывод произвольного слова $v \in R$ из группы слов одинаковой длины за k шагов.

   Переход индукции $n = k + 1$:

   Докажем, что слово w, выведенное за $k + 1$ шагов, принадлежит языку R. Возможно два случая.

Случай 1. Слово w может иметь вид $a_i u a_i$, где $u \in R$. Тогда по предположению индукции существует вывод слова $v \in R$ за k шагов:

$$S \Rightarrow^k v,$$

следовательно, возьмём вывод v, сделаем ещё два шага и получим вывод слова w (без ограничения общности была выбрана именно такая последовательность):

$$S \Rightarrow^{k-1} xBy \Rightarrow xa_iBa_jy \Rightarrow xa_ia_jy = w \in R,$$

где x, y — некоторые слова, получившиеся при выводе, удовлетворяющие условиям случая 1.

Замечание. *При выводе было известно, что после $k-1$ шага при подстановки вместо нетерминала B любого символа (отличного от нетерминального) получалось слово $v \in R$. Поэтому на k шаге вместо нетерминала B было подставлено правило, содержащее терминал B, а дальше был подставлен произвольный символ (отличный от терминального).*

Случай 2. Слово w может иметь вид $a_i u a_j$, где u — слово, получившиеся при выводе. Аналогино случаю 1 возьмём вывод слова v из предложения индукции, затем сделаем ещё два шага и получим искомое слово $w \in R$.

Таким образом, утверждение доказано. □

Задача 4

Покажите, что дополнение языка $U = \{a^n b^n c^n \mid n = 0, 1, 2, \ldots\}$ является КС-языком.

Задача 6

Построить ДМП-автомат, распознающий язык D_2, порожденный грамматикой:

$$S \to SS \mid (S) \mid [S] \mid \varepsilon.$$

Решение.