Домашнее задание №9

Томинин Ярослав, 778 группа

15 ноября 2018 г.

1.

G_n**:**

$$
\begin{aligned}
S &\to A_1A_2...A_nA_n...A_1\\
A_i &\to aA_{i+1}\forall i \in [1, n-1]\\
A_n &\to ab
\end{aligned}
$$

Докажем, что эта грамматика порождает нужное слово и только его. Заметим, что мы можем убрать нетерминал используя второй тип правил, пока не добъемся A_n, а только потом уже убрать A_n. Именно поэтому терминал A_i превратится в $a^{n-i+1}b$.(Еще убедительней это можно доказать по индукции по i: для n-очевидно, для i мы можем свести его к i+1 только единственным правилом, поэтому в нем будет на 1 букву a больше, далее используем предположение индукции и получаем то же утверждение для нетерминала с меньшим индексом). Тогда мы можем получить желаемое слово и только его. Ясно, что длина описания cn, на первой строчке cn, на следующих n строчках по константе.

2.

Алгоритм: поставим в соответствие

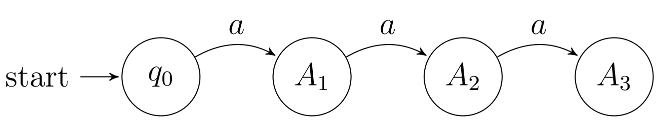
3.

a)

Построение SLG G_w**:**

$$
\begin{aligned}
S &\to A_1A_2A_3A_4\\
A_1 &\to a, u = a^7\\
A_2 &\to A_1a, u = a^5\\
A_3 &\to A_2a, u = a^2\\
A_4 &\to A_2, u = \epsilon
\end{aligned}
$$

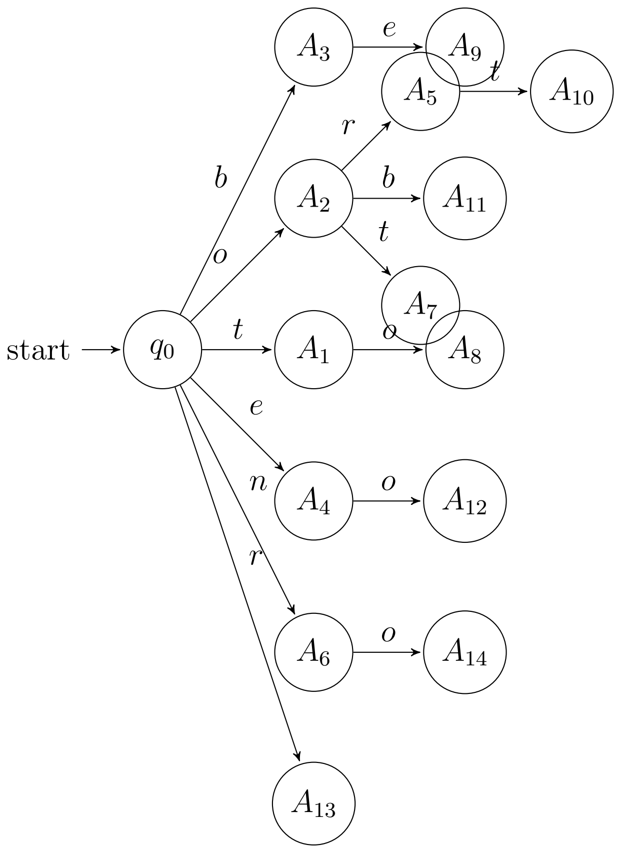
Построение LZW-автомата:



b)
Построение SLG G_w:

$$
\begin{aligned}
S &\to A_1A_2A_3A_4A_5A_6A_7A_8A_9A_{10}A_{11}A_{12}A_{13}A_{14}A_{15}\\
A_1 &\to t, u = obeornottobeortobeornot\\
A_2 &\to o, u = beornottobeortobeornot\\
A_3 &\to b, u = eornottobeortobeornot\\
A_4 &\to e, u = ornottobeortobeornot\\
A_5 &\to A_2r, u = nottobeortobeornot\\
A_6 &\to n, u = ottobeortobeornot\\
A_7 &\to A_2t, u = tobeortobeornot\\
A_8 &\to A_1o, u = beortobeornot\\
A_9 &\to A_3e, u = ortobeornot\\
A_{10} &\to A_5t, u = obeornot\\
A_{11} &\to A_2b, u = eornot\\
A_{12} &\to A_4o, u = rnot\\
A_{13} &\to r, u = not\\
A_{14} &\to A_6o, u = t\\
A_{15} &\to A_1, u = \epsilon
\end{aligned}
$$

Построение LZW-автомата:



4.

$$
\begin{aligned}
S &\to A_2A_1A_2, 3\\
A_1 &\to tobeor, 6\\
A_2 &\to A_1not, 4
\end{aligned}
$$

Всего 13 символов. Изначально было 38. Аналогично 1 заданию можно доказать что, из A_1 можно породить tobeor и только его(по построению), а из A_2 можно породить tobeornot и только его(следует из предыдущего утвержения и построения грамматики).

5.

1)

Пункт a покрывается пунктом b, поэтому я буду решать 2 пункта.

b)

Если решим систему, то мы как раз найдем минимальное по включению

решение
$X = ((110)^* + 111^*)^*$ в)
Общее решение:$X = (L')^*(\epsilon + L)$,где L произвольный язык, а L'-язык, содержащий $((110)^* + 111^*)$
Внимание, для слова существуют такие L,L', то оно является решением, я не утверждаю, что если существует L,L' для слова, которые не подходят под мои условия, то это слово не является решением(так как у него может быть подходящая пара L,L')
Докажем, что любое решение можно выразить так. От противного: допустим, что это не так, тогда найдется решение, в котором L' не содрежит $((110)^* + 111^*)$. Но после замыкания X (рекурсивного повторения операции вывода) X будет иметь вид $X = ((110)^* + 111^*)^*(L')(\epsilon + L)$, что противоречит предположению, так как теперьL' содержит $((110)^* + 111^*)^*$.
Докажем, что все, что все множнство содержит только решения, достаточно доказать, что $\alpha^i L'(\epsilon + L) = L'(\epsilon + L)$. Но, должно быть понятно, что 1 содержится во 2 в силу ограничения L'. А 2 содержится в 1 так как мы можем всегда из α выбрать ϵ.
2)
b)

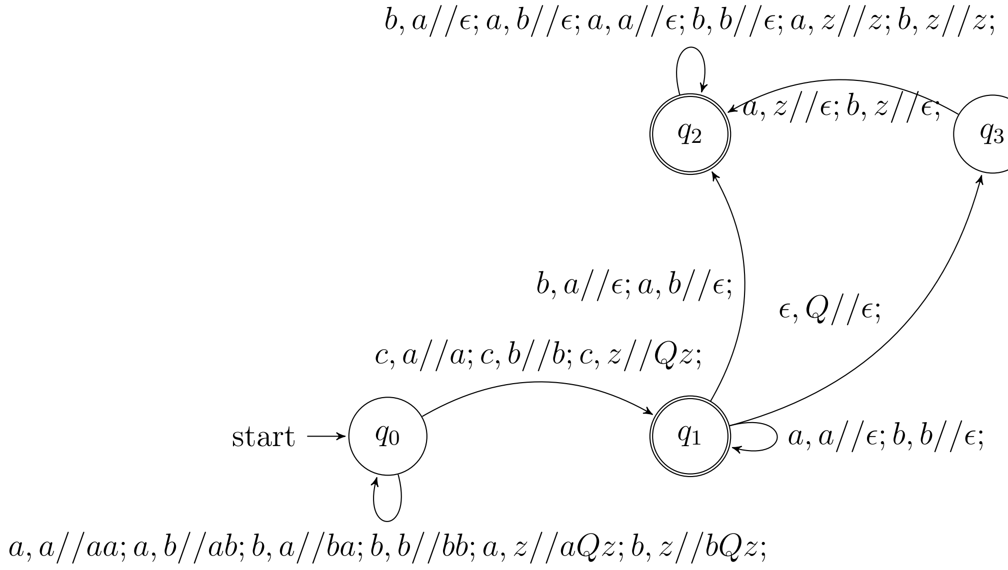
$$X = (00 + 01 + 10 + 11)^*(0 + 1 + \epsilon)$$

в)
Так как это весь алфавит, то это и есть общее решение.
3)
b)

$$\begin{aligned}
&Q_0 = 0^*(1Q_1 + \epsilon) \\
&Q_1 = 10^*1Q_1 + 10* + 0Q_2 \\
&Q_1 = (10^*1)^*0Q_2 \\
&Q_2 = (0(10^*1)^*0 + 1)Q_2 \\
&Q_2 = (0(10^*1)^*0 + 1)^* \\
&Q_1 = (10^*1)^*0(0(10^*1) * 0 + 1)^* \\
&Q_0 = 0^*(1(10^*1)^*0(0(10^*1)^*0 + 1)^* + \epsilon)
\end{aligned}$$

в)

6.



Утверждение:данный автомат удовлетворяет условию задачи.
Будем говорить, что слово - паллиндром, если оно имеет следующий вид:$w = xcx^R$ **План доказательства**:
1. После перехода в 1 состояние в нашем стеке будет лежать символы zQ и слово x(если смотреть снизу вверх)
2. Находясь в 1 состоянии нет нарушений "паллиндромности"слова
3.При переходе в 3 состояние прочитанное слово имеет вид $w = xcx^R$ **и если слово имеет вид** $w = xcx^R$**, то оно в 3 состоянии**
4. Находясь в состоянии 2, мы уверены, что слово не является паллиндромом и если есть нарушение паллиндромности, то слово находися в 2
Доказательство 1:При считывании букв из x мы будем добавлять в стек каждую букву, так как мы можем перейти в 1 состояние только по c, то на данный момент в стеке будут лежать буквы, соответствующие нашему утверждению(если слово x не пустое). Если же слово x пустое, то при переходе по c мы добавим в стек нужный символ и этот случай тоже будет подходять под наше утверждение.
Доказательство 2:Так как мы можем попасть в 1 состояние только из 0, то выполняется утверждение 1. Так как для того, чтобы остаться в этом состоянии, верхний символ стека должен соответствовать прочитанному нами символу, то это означает, что если мы еще не перешли из

этого состояния в никакое другое, то не было нарушения паллиндромности.
Доказательство 3: В 3 состояние мы можем попасть только из 1, исходя из 2 утверждения и того факта, что мы прочитали Q, мы можем утверждать, что не было нарушения паллиндромности и мы прочитали все слова, следовательно, выполняется утверждение 3.
Доказательство 4:Попасть во 2 состояние мы можем либо из первого: тогда у нас есть ошибка паллиндромности(так как, когда мы находились в 1 ее не было в силу 2 утверждения, а перейти во 2 мы могли лишь тогда, когда прочитанный символ не равен верхнему символу в стеке); или из 3 состояния, но тогда $|y| > |x|$(в силу 3 утверждения).
Так как, приходя во 2 состояние, мы будем переходить в него же до тех пор, пока слово не закончится(так как определены переходы в случаях, когда в стеке лежит a,b,z, а z всегда лежит в стеке). Поэтому, в силу того, что в 1 наше слово не содержит нарушений, но не является паллиндром и, что во второе состояние мы попадаем тогда и только тогда, когда у нас есть нарушение, этот автомат распознает заданный язык.
7.
8.
Сначала удалим бесплодные символы.

$$\begin{aligned} &V_0 = a, b \\ &V_1 = a, b, A, B, C \\ &V_2 = a, b, A, B, C, S, F, E \end{aligned}$$

Следовательно, G- бесплодный символ.
После его удаления, грамматика будет иметь вид:

$$\begin{aligned} &S \to A|B|C|E \\ &A \to C|aABC|\epsilon \\ &B \to bABa|\epsilon \\ &C \to BaAbC|\epsilon \\ &F \to aBaaCbA \\ &E \to A \end{aligned}$$

Удалим недостижимые.

$$V_0 = S$$
$$V_1 = S, A, B, C, E$$
$$V_2 = S, A, B, C, E$$

F- недостижимо. Удалим его и получим приведенную грамматику:

$$S \to A|B|C|E$$
$$A \to C|aABC|\epsilon$$
$$B \to bABa|\epsilon$$
$$C \to BaAbC|\epsilon$$
$$E \to A$$

9.

Построим КС-грамматику по алгоритму:
1)Добавим правила вывода:

$$S \to [qZ_0q]|[qZ_0p]$$

2)рассмотрим переход $(q, X) \in \delta(q, a, Z_0)$ и добавим соответствующие правила вывода

$$[qZ_0q] \to a[qXq]$$
$$[qZ_0p] \to a[qXp]$$

3)рассмотрим переход $(q, XX) \in \delta(q, a, X)$ и добавим соответствующие правила вывода

$$[qXq] \to a[qXq][qXq]$$
$$[qXq] \to a[qXp][pXq]$$
$$[qXp] \to a[qXq][qXp]$$
$$[qXp] \to a[qXp][pXp]$$

4)рассмотрим переход $(q, XX) \in \delta(q, b, X)$ и добавим соответствующие правила вывода

$$[qXq] \to b[qXq][qXq]$$
$$[qXq] \to b[qXp][pXq]$$
$$[qXp] \to b[qXq][qXp]$$
$$[qXp] \to b[qXp][pXp]$$

5)рассмотрим переход $(p, X) \in \delta(q, b, X)$ и добавим соответствующие правила вывода

$$[qXq] \to b[pXq]$$
$$[qXp] \to b[pXp]$$

6)рассмотрим переход $(p, \epsilon) \in \delta(p, a, X)$ и добавим соответствующие правила вывода

$$[pXp] \to a$$

7)рассмотрим переход $(p, \epsilon) \in \delta(p, b, X)$ и добавим соответствующие правила вывода

$$[pXp] \to b$$

В результате получим:

$$
\begin{aligned}
S &\to [qZ_0q]|[qZ_0p] \\
[qZ_0q] &\to a[qXq] \\
[qZ_0p] &\to a[qXp] \\
[qXq] &\to a[qXq][qXq] \\
[qXq] &\to a[qXp][pXq] \\
[qXp] &\to a[qXq][qXp] \\
[qXp] &\to a[qXp][pXp] \\
[qXq] &\to b[qXq][qXq] \\
[qXq] &\to b[qXp][pXq] \\
[qXp] &\to b[qXq][qXp] \\
[qXp] &\to b[qXp][pXp] \\
[qXq] &\to b[pXq] \\
[qXp] &\to b[pXp] \\
[pXp] &\to a \\
[pXp] &\to b
\end{aligned}
$$

Удалим бесплодные символы

$$
\begin{aligned}
&V_0 = a, b \\
&V_1 = a, b, [pXp] \\
&V_2 = a, b, [pXp], [qXp] \\
&V_3 = a, b, [pXp], [qXp], [qZ_0p] \\
&V_4 = a, b, [pXp], [qXp], [qZ_0p], S
\end{aligned}
$$

После удаления бесплодных

$$
\begin{aligned}
S &\to [qZ_0p] \\
[qZ_0p] &\to a[qXp] \\
[qXp] &\to b[pXp] \\
[qXp] &\to b[qXp][pXp] \\
[qXp] &\to a[qXp][pXp] \\
[pXp] &\to a \\
[pXp] &\to b
\end{aligned}
$$

Найдем недостижимые

$$
\begin{aligned}
&V_0 = S\\
&V_1 = S, [qZ_0p]\\
&V_2 = S, [qZ_0p], [qXp]\\
&V_3 = S, [qZ_0p], [qXp], [pXp]
\end{aligned}
$$

Все достижимы, поэтому получаем:

$$
\begin{aligned}
S &\to [qZ_0p]\\
[qZ_0p] &\to a[qXp]\\
[qXp] &\to b[pXp]\\
[qXp] &\to b[qXp][pXp]\\
[qXp] &\to a[qXp][pXp]\\
[pXp] &\to a\\
[pXp] &\to b
\end{aligned}
$$