Assignment 13, Authomata Theory

Oleg Sivokon

<2015-09-11 Fri>

Contents

1 Problems

1.1 Problem 1

Prove that

$$L = \{w \in \{a, b, c, d\}^* \mid w = dv, v \in \{a, b, c\}^*, \#_a(w) \cdot \#_c(w) < \#_b(w)\}$$

is not regular.

1.1.1 Answer 1

Assume, for contradiction that L is regular, then pumping lemma must hold. Let p be the pumping length of L, the $dabc^p b^p b \in L$ because $1 \cdot p = p$. Consider cases of selecting the middle substring y defined by pumping lemma:

1. If y contains d, then we cannot repeat it, since by definition d happens only once in the beginning of each word.
2. If y contains a we cannot repeat it, since in that case $p \cdot x > p$ whenever $x > 1$, which violates the definition of L.
3. If we repeat any string containing c, (but not containing b, which see in the next bullet point), we will violate the requirement that $\#_b(w) > \#_c(w) \cdot \#_a(w)$.
4. Since the prefix concatenated with y should not be longer than p, we cannot repeat any string containing b because the prefix of length p has no b in it.

Since these are our only options, it must be impossible to satisfy the requirements of pumping lemma, hence, by contradiction, the language L is not regular.

1.2 Problem 2

Prove that the language

$$L = \{w \in \{x, y\}^* \mid (w = x^k(yyyy)^{m!}, k, m \geq 2) \vee w = y^{2l}, l \geq 2\}$$

is not regular.

1.2.1 Answer 2

Suppose, for contradiction L is regular, then pumping lemma must be applicable to it. Consider the substrings of $y^{2(p+1)}$, where p is the pumping lenght. Consider the second condition of pumping lemma, i.e. that the prefix concatenated to the pumping substring should be no longer than the pumping lenght, thus $|vw| \leq p$, this leaves us with the prefix of at most y repeated p times, but if we repeat it, we will produce the string $y^{2(p+1)+p} = y^{2+3p}$. Provided $p > 2$, this string will be too short to satisfy the requirements of the lemma. Hence, the original conjecture is falsified, hence L is not regualr.

1.3 Problem 3

Let L be a regular language over alphabet Σ. Prove that the following languagage is also regular.

$$\begin{aligned}
Reversed(\mu_1\mu_2 \ldots \mu_n, n) =& \mu_1, \mu_2, \ldots, \mu_n \in \Sigma, \\
& \mu_1\mu_2 \ldots \mu_n \in L^R. \\
Interleaved(\mu_1\mu_2 \ldots \mu_n, n) =& \exists(\sigma_1, \sigma_2, \ldots, \sigma_n, \zeta_1, \zeta_2, \ldots, \zeta_n \in \Sigma) : \\
& (\mu_1\sigma_1\zeta_1\mu_2\sigma_2\zeta_2 \ldots \mu_n\sigma_n\zeta_n \in L) \\
\hat{L} = & \{\mu_1\mu_2 \ldots \mu_n \mid \\
& n \geq 0, \\
& Reversed(\mu_1\mu_2 \ldots \mu_n, n), \\
& Interleaved(\mu_1\mu_2 \ldots \mu_n, n)\}
\end{aligned}$$

And whenever $\epsilon \in L$, it is also the case that $\epsilon \in \hat{L}$.

1.3.1 Answer 3

First I note that L^R is also regular, this is so because there must be a DFA accepting L (by definition), we can transform this DFA in the following way:

1. Reverse all transitions.
2. Make the start state an accepting one.
3. Make all previously accepting states connect to a newly added start state by ϵ-transitions. Thus we obtain an NFA for the reversed language.

Thus, $Rvererersed(\dots)$ alone selects a regular language.

Next, I note that regular languages are closed under intersection. Thus proving that $Interleaved(\dots)$ predicate selects a regular language will prove that $\hat{L}$ is regular, but we are given by definition, that $Interleaved(\dots)$ selects the language L, or some regular subset of it, hence $\hat{L}$ must be regular. The subset is regular because it is essentially given to us by a regular expression $(\mu_n \sigma_n \zeta_n)^*$, where μ_n, σ_n and ζ_n are character classes of size at most n.

1.4 Problem 4

Let L be a regular language over Σ. Prove that the following language is also regular:

$$\frac{1}{3}L = \{w \in \Sigma^* \mid wxy \in L, |x| = |y| = |w|\}$$

1.4.1 Answer 4

The language $\frac{1}{3}L$ is regular because it is possible to construct a regular expression accepting it. Since we know that regular languages are closed under concatenation, we can devise a regular expression accepting the w part of the language L, and, similarly, for the xy part. The regular expression accepting the w part guarantees us that the language $\frac{1}{3}L$ is regular.

1.5 Problem 5

1. Write regular expression accepting the language $0^*01^*/0^+$.
2. Prove that if L is regular then $\overset{\leftrightarrow}{L} = \{xy \in \Sigma^* \mid yx \in L\}$.
3. What is wrong with $\overset{\leftrightarrow}{L} = (\Sigma^* \setminus L).(L/\Sigma^*)$ if it was offered as a solution for the previous question?

1.5.1 Answer 5

$0^*01^*/0^+ = 0^*01^+$. The rationale for this answer is that the string in this language cannot end in 0, but the original regex would not accept strings 0^k where $k < 2$, thus we have the result, where at least one zero must be followed by at least one one.

1.5.2 Answer 6

The proof is immediate from concatenation closure properties: Languages of $x's$ and $y's$ must be regular, because L is regular. Hence their concatenation xy is regular too.

1.5.3 Answer 7

Assuming backwards slash means left quotient rather than complement, then the general idea for the proof seems to be a workable one, except that one shouldn't use the same character L to denote languages made of $x's$ and $y's$. To fix this, we could do the following:

$$
\begin{aligned}
L_y &= L/\{x\} \\
L_x &= L \setminus \{y\} \\
\overset{\leftrightarrow}{L} &= (L \setminus L_y).(L/L_x)
\end{aligned}
$$