Albert-Ludwigs-Universität, Inst. für Informatik
Prof. Dr. Fabian Kuhn
Philipp Bamberger, Yannic Maus

Theoretical Computer Science - Bridging Course
Summer Term 2018
Exercise Sheet 3

for getting feedback submit (electronically) before the start of the tutorial on 12th of November 2018.

Exercise 1: Regular Expressions *(2+6 Points)*

(a) Which of the following words are in the language described by the regular expression $r = a(ab)^*a$.

(a) abababa

(b) aaba

(c) aabbaa

(d) aba

(e) aabababa

(b) Give a regular expression for each of the following languages (you do not need to prove the correctness of your answers).

(a) $L_1 = \{ab, ba, bb, aa, abab\}$.

(b) $L_2 = \{w_1 w_2 w_3 \in \{a, b, c\}^* \mid w_1 \text{ contains an even number of a's}, w_2 \text{ contains no } b, w_3 \in L_1\}$

(c) L_3 is the language over alphabet $\{a, b\}$ that consists of all words that do not contain the substring aa.

Sample Solution

1. b) and e) are contained in the language. All other ones are not contained.
2. We use the + symbol where the lecture uses the $\cup$ symbol. Note that both notations are standard.

(a) $r_1 = ab + ba + bb + aa + abab$

(b) Let

- $A_1 = \{w \mid w \text{ contains an even number of a's}\}$
- $A_2 = \{w \mid w \text{ contains no } b\}$

$r_{A_1} = (b + c)^* + ((b + c)^* a(b + c)^* a(b + c)^*)^*$ is a regular expression for A_1.
$r_{A_2} = (a + c)^*$ is a regular expression for A_2.
Then $r_{A_1} r_{A_2} r_1$ is a regular expression for L (with r_1 as in (a)).

(c) $r_3 = (\epsilon + a)(b + ba)^*$.

Exercise 2: The Pumping Lemma: Sufficiency or Necessity? *(4 Points)*

Consider the language $L = \{c^m a^n b^n \mid m, n \geq 0\} \cup \{a, b\}^*$ over the alphabet $\Sigma = \{a, b, c\}$.

(a) Describe in words (not using the pumping lemma), why L cannot be a regular language.

(b) Show that the property described in the Pumping Lemma is a necessary condition for regularity but not sufficient for regularity.

Hint: Use L as counter example, i.e., show that it can be 'pumped' (in the sense of the pumping lemma), but is still not regular.

Sample Solution

(a) For recognizing that a word has the same number of a's and b's, a DFA would have to count the number of appearances of these characters, requiring at least one state for each appearance. But as the number of appearances can be arbitrary large, the automaton would need an infinite number of states.

(b) We show that L has the properties described in the Pumping Lemma. Then we showed that for a language, having these properties do not imply regularity.

As the pumping length we choose an arbitrary $p \geq 1$. Let x be some word of length at least p. We must show that there is a composition $x = uvw$ having the three properties from the lemma:

(1) $|v| \geq 1$

(2) $|uv| \leq p$

(3) for all $i = 0, 1, 2, \ldots$ it holds: $uv^i w \in L$

This is clear if $x \in \{a, b\}^*$. So assume $x = c^m a^n b^n$ with $m \geq 1$. We can choose $u = \epsilon$, $v = c$ and $w = c^{m-1} a^n b^n$ as a composition of x having properties 1, 2, 3.

Exercise 3: *(6 Points)*

Let $\Sigma = \{0, 1\}$, prove the following:

(a) (3 points) The language $A = \{0^k w 0^k | k \geq 1 \text{ and } w \in \Sigma^*\}$ is regular.

(b) (3 points) The language $B = \{0^k 1 w 0^k | k \geq 1 \text{ and } w \in \Sigma^*\}$ is not regular.

Sample Solution

(a) The language A contains any string which starts and ends with 0. This because the substring w in the middle could be an arbitrary string, e.g., $0^k w 0^k = 0w'0 \in A$, since $w' = 0^{k-1} w 0^{k-1} \in \Sigma^*$. Hence, we can easily get an automaton which accepts the language containing strings which start and end with 0.

(b) Intuitively, an automaton needs to count the number of 0s before 1, so that it can verify the given string contains at least equal number of 0s on the other end. However, we know that the finite automata can not count or memorize a large number, if k is very large. Therefore, there would always exists a string (possibly very large) in B which can not be accepted by a finite automaton. Thus B can not be a regular language.

Formally, one can show a contradiction directly from the Pumping Lemma. Consider a string $s = 0^p 1 0^p$ (assuming $p > 0$ be the pumping length) that is in B. Let $s = xyz$ be a decomposition of the word that satisfies the requirements of the pumping lemma. As $|xy| \leq p$ we have that $x = 0^{p-q-r}, y = 0^q, z = 0^r 1 0^p$, for some $q \geq 1$ and $r \geq 0$. Then all the conditions of the Pumping Lemma are satisfied. Therefore, if B is regular, then xy^2z must be in B (by Pumping Lemma). However, $xy^2z = 0^{p+q} 1 0^p$ does not belongs to B, which contradicts that B is regular.

Exercise 4: Context Free Grammar *(9 Points)*

Give context-free grammars that generate the following languages. The alphabet set is $\Sigma = \{0, 1\}$.

(a) (2 points) $\{w \mid w \text{ starts and ends with the same symbol}\}$

(b) (3 points) $\{w \mid \text{the length of } w \text{ is odd}\}$

(c) (4 points) The complement of the language $\{0^n 1^n \mid n \geq 0\}$

Sample Solution

(a) $S \to 0A0 \mid 1A1 \mid \epsilon$, $A \to 0A \mid 1A \mid \epsilon$

(b) $S \to 0S0 \mid 0S1 \mid 1S0 \mid 1S1 \mid A$, $A \to 0 \mid 1$

(c) $S \to 1A \mid A0 \mid 0S1$, $A \to 0A \mid 1A \mid \epsilon$. Notice, for a string s to be in the language, it must either: (1) s starts with 1; or (2) s ends with 0; or (3) s starts with 0 and ends with 1 but the intermediate part is in the language.