

Syntax and Semantics

Exercise Session 5

Before presenting the solutions of the exercises, we recall the statement of the pumping lemma and we describe how it can be used to prove that a language is not regular.

Pumping Lemma. *If $L \subseteq \Sigma^*$ is a regular language, then there exists a number $p \in \mathbb{N}$, called pumping length for L , such that for any word $w \in L$ with $|w| \geq p$, there exist $x, y, z \in \Sigma^*$ such that $w = xyz$ and the following hold*

- (i) $xy^iz \in L$ for any $i \in \mathbb{N}$.
- (ii) $|y| > 0$
- (iii) $|xy| \leq p$

How to use the Pumping Lemma: The pumping lemma is often used to prove that a language L is not regular. This is done by contradiction (i.e., showing that the assumption that L is regular leads to a contradiction) following the steps listed below:

1. Assume that L is regular and that $p \in \mathbb{N}$ is a pumping length for L ;
2. Choose a word $w \in L$ s.t. $|w| \geq p$ (notice that w depends on p);
3. Show that there is no way to choose $x, y, z \in \Sigma^*$ and satisfy $w = xyz$, (i), (ii) and (iii) at the same time.

In the last step it is often the case that one shows that in order to fulfill $w = xyz$, (ii) and (iii), the substrings x and y get a shape that allows one to find an index $j \in \mathbb{N}$ such that $xy^jz \notin L$, making (i) impossible to hold.

Remark. We recall that regular languages are closed under complement, intersection, union, concatenation, and Kleene star. Sometimes it can be helpful to exploit such closure properties in order to either make easier the application of the pumping lemma or reduce the problem to a well known one.

Take for instance the language $L \subseteq \{a, b, c\}^*$ defined as

$$L = \left\{ w \in \{a, b, c\}^* \left| \begin{array}{l} \text{if } w \text{ has an odd number of } a\text{'s} \\ \text{then it has as many } c\text{'s as } b\text{'s} \end{array} \right. \right\}$$

We proceed by contradiction. Assume L to be regular. Since regular languages are closed by intersection we have that $L' = L \cap \mathcal{L}(c^*ab^*)$ is regular. Notice that

$$L' = \{c^nab^n \mid n \geq 0\}.$$

Following the steps listed above one can easily prove that L' is not regular¹. This leads to a contradiction thus L cannot be regular \square

Exercise 1.

Prove that if a regular language $L \subseteq \Sigma^*$ has a pumping length p , then any $q \geq p$ is also a pumping length for L .

Exercise 2.

Prove that the following languages are not regular

$$\begin{aligned} L_1 &= \{a^ib^jc^k \mid i, j, k \geq 0 \text{ and if } i = 1 \text{ then } j = k\} \\ L_2 &= \{0^n1^m0^n \mid m, n \geq 0\} \\ L_3 &= \{0^m1^n \mid m \neq n\}. \end{aligned}$$

Exercise 3.

Consider the alphabet Σ and let $L \subseteq \Sigma^*$. We say that two strings $x, y \in \Sigma^*$ are indistinguishable by L , and we write $x \equiv_L y$, if for any string $z \in \Sigma^*$, $xz \in L$ if and only if $yz \in L$.

Prove that \equiv_L is an equivalence relation.

¹Left as exercise.

Exercise 4.

Construct an NFA that recognizes the language $(01 \cup 001 \cup 010)^*$. Convert this NFA to an equivalent DFA.

Exercise 5.

Consider the language over the alphabet $\Sigma = \{a, b\}$.

$$L = \left\{ w \left| \begin{array}{l} w \text{ has an even number of } a\text{'s and an odd number of } b\text{'s,} \\ \text{and } w \text{ does not contain the substring } ab \end{array} \right. \right\}.$$

Prove that L is regular. Find a regular expression that generates L .