Free University of Bozen-Bolzano – Faculty of Computer Science
Bachelor in Computer Science and Engineering
Formal Languages and Compilers– A.Y. 2017/18 Mid-Term Exam – 21/11/2017
Prof. Alessandro Artale – *Time: 120 minutes*

`This is a closed book exam: the only resources allowed are blank paper, pens, and your head. Explain your reasoning. Write clearly, in the sense of logic, language and legibility. The clarity of your explanations affects your grade.`

Problem 1 [9 points] Decide which of the following statements is TRUE and which is FALSE. You must give a **clear explanation** of your answer to receive full credit.

(a) Let L_1, L_2 be any two regular languages over the same alphabet Σ, then the language $L = \{w \in \Sigma^* \mid w \in L_1 \text{ or } w \notin L_2\}$ is regular.

(b) If $L = L_1 \cap L_2$ is regular and L_2 is regular, then L_1 must be regular.

(c) The language $L = \{a^n b^n \mid n \geq 1\}$ is regular (use the Pumping Lemma).

Problem 2 [9 points]

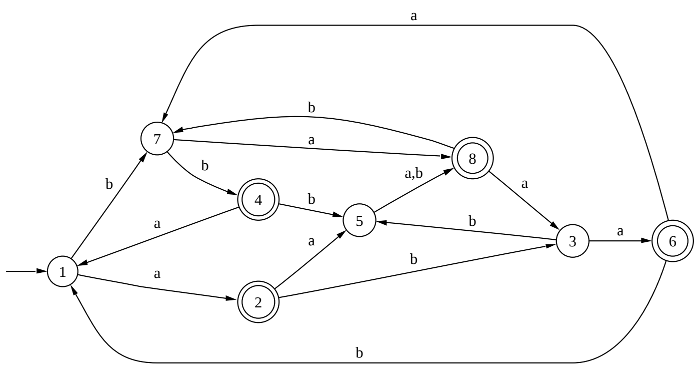
(a) Construct a regular expression $R.E.$ that generates the language over the alphabet $\{a, b, x\}$ constituted by all strings containing an odd number of a's, and in which between each pair of consecutive a's there is an even number of b's (and an arbitrary number of x's). E.g., $babbxbbxabbxaabxxbaxbx \in \mathcal{L}(A)$ and $bab \in \mathcal{L}(A)$, while $abba \notin \mathcal{L}(A)$ and $abbbaa \notin \mathcal{L}(A)$. [2 Points]

(b) Construct an NFA, A_N, that accepts the language specified above. [2 Points]

(c) Given two DFA's, said A and B, describe the notion of *Product Automaton*. Furthermore, show how this notion can be used to prove that regular languages are closed under difference (i.e., if L_1 and L_2 are regular languages, then so is $L_1 \setminus L_2$). [3 Points]

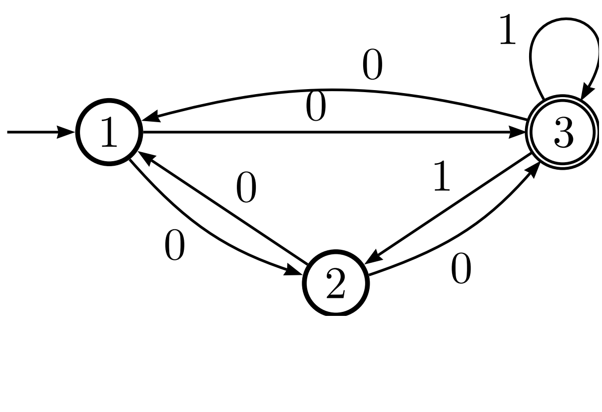
(d) Give the CFG for the language $L = \{0^n 1^{2n} \mid n \geq 1\}$. [2 Points]

Problem 3 [4 points] Consider the following DFA, A_D, over $\Sigma = \{a, b\}$:



(a) Construct a DFA A_m with minimal number of states such that $\mathcal{L}(A_m) = \mathcal{L}(A_D)$. The **algorithms** you have followed to construct A_m should become evident in your construction.

Problem 4 [6 points] Consider the following NFA, A_N, over $\{0, 1\}$:



(a) Construct a DFA, A_D, such that $\mathcal{L}(A_D) = \mathcal{L}(A_N)$. The **algorithms** you have followed to construct A_D should become evident in your construction. [4 Points]

(b) Show **all** possible sequences of states of A_N that are traversed for the string 0011. Show also the sequence of states of A_D for the same string. [2 Points]

Problem 5 [6 points]

(a) Apply the sequence of steps that are necessary to simplify a context free grammar and convert it into a Clean-up Form to the context free grammar $G = (\{S, A, B, C, D, E\}, \{a, b, c\}, P, S)$, where P consists of the following productions:

$$
\begin{array}{lcl@{\qquad\qquad}lcl}
S & \longrightarrow & aBbAb \mid bBa \mid AA & C & \longrightarrow & cC \mid aB \\
A & \longrightarrow & bS \mid B & D & \longrightarrow & aS \mid \varepsilon \\
B & \longrightarrow & aE \mid D & E & \longrightarrow & aEE \mid cEC
\end{array}
$$

The normalizations steps must be carried on in the **correct order** and the **algorithms** used for each step must become evident to get full mark.