INHN0013 Information Theory & Theory of Computation
Exercise Sheet 2

Assigned: Wednesday October 22 2025
Due: Wednesday October 29 2025

You should know how to solve these problems and ideally you will solve them in the allotted one week. Some of the specific problems below might appear on the optional or actual final exam. You will see the solutions for instances that appear on the optional exams. Many of the problems (although not necessarily every single subproblem of every problem) will be discussed in the tutorials.

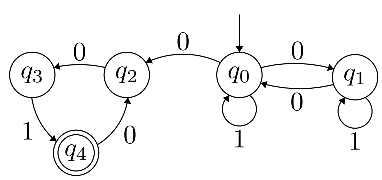
1. (a) Formally prove by construction that regular languages are closed under the complement operation. That is, given a DFA for some regular language, modify it so that it recognizes precisely the complement of the language, and argue the claim using the formal definition of computation for a DFA.

   (b) Formally prove by construction that regular languages are closed under the intersection operation. That is, given DFAs for two regular languages, create a new DFA that recognizes precisely the intersection of the two languages, and argue the claim using the formal definition of computation for a DFA.

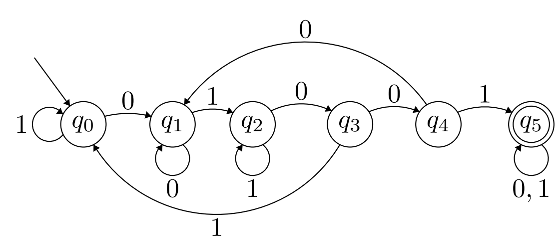
   (c) Consider a proof by construction that regular languages are closed under the complement operation, but this time based on modifying an NFA for the language, using the same method as that for DFAs above. Show that this approach does not lead to a correct proof.

2. For each of the following languages an NFA has been constructed, supposedly recognizing the language. Verify whether these are correct and if not identify the bugs, explain what the problems are, and show how to correct them. You may add, delete, or modify states and transitions, but do not construct an entirely new machine. The alphabet used for each language is $\{0, 1\}$.

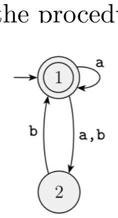
   (a) $L_1 = \{uv \mid u \text{ has an even number of 0's and is not empty, and } v \in (001)^+\}$.

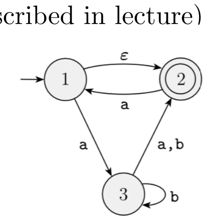


   (b) $L_2 = \{w \mid w \text{ has } 01001 \text{ as a substring}\}$.



3. Convert the NFAs below to DFAs using the procedure described in lecture) and show intermediate steps:





4. Build FAs for the following languages

   (a) $L_1 = \{xy | x \in A \text{ and } y \notin A\}$ where A is some regular language

   (b) $L_2 = \{w | w \in \{0 \cup 1\}^*$ with equal number of 0's and 1's and every prefix[1] of w has at most one more of either 0's or 1's $\}$.

   (c) $L_3 = \{w | w \in \{0 \cup 1\}^*$ with equal number of 0's and 1's and every prefix of w has at most two more of either 0's or 1's $\}$.

5. Prove or disprove the following claims:

   (a) Every subset of a regular language is a regular language.

   (b) Every regular language with infinite number of words contains a regular subset.

   (c) If $L_1 \subseteq L_2$ and L_1 is regular, then L_2 is also regular.

   (d) If $L_1 \subseteq L_2$ and L_2 is non-regular, then L_1 is also non-regular.

6. Convert the following regular expressions to NFAs, showing intermediate steps:

   (a) $0^* \cup (01)^+$

   (b) $(((11)^*(00)) \cup 101)^*$

   (c) $(0 \cup 1)^* 00 (0 \cup 1)^*$

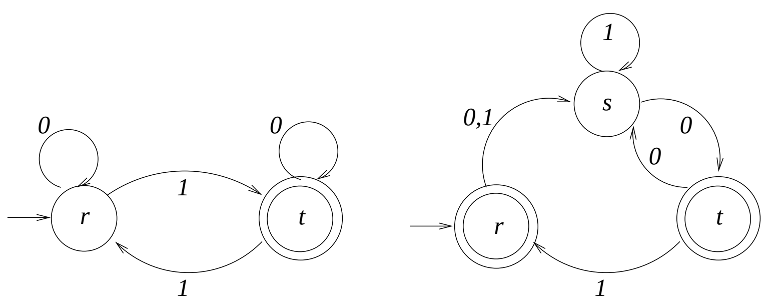
7. Produce regular expressions for the following languages.

   (a) $L_1 = \{w | w$ starts with a 1 and has at most one 0$\}$.

   (b) $L_2 = \{w | w$ has an odd number of 1s and an even number of 0s$\}$.

   (c) $L_3 = \{w | w$ has exactly three 1s and any number other than two of 0s$\}$.

8. Convert the DFAs below to regular expressions using the GNFA-based approach:



9. Prove or disprove each of the following for regular expressions r, s and t. Here, $r = s$ means $L(r) = L(s)$:

   (a) $r(s \cup t) = rs \cup rt$

   (b) $(\epsilon \cup r)^* = r^*$

   (c) $(r \cup s)^* = r^* \cup s^*$

   (d) $r \cup (st) = (r \cup s)(r \cup t)$

   (e) $(r \cup s)^* = (r \cup sr^*)^* = (r^* \cup sr^*)^*$

10. Prove or disprove that the languages over alphabet $\{0, 1\}$ below are regular.

   (a) $L_1 = \{w | w$ has the same number of 0's and 1's$\}$

   (b) $L_2 = \{w | w$ such that the number of 0's plus twice the number of 1's is equal to 5$\}$

   (c) $L_3 = \{w | w = uvu$ where u, v are nonempty strings over the alphabet$\}$

   (d) $L_4 = \{w | w$ is not a palindrome$\}$

[1] We say that x is a prefix of y if there exists z such that $xz = y$.