1. The “closed-book/notes” examination contains 3 problems. You have 90 minutes for 35 points.

2. Show all important steps in your work. Your answers will be graded on its correctness and clarity.

1. [10 points] Prove, or disprove by giving a counter-example with explanation, each of the following statements.

   (a) [2] For arbitrary languages $L_1$, $L_2$, and $L_3$ over an alphabet $\Sigma$, $(L_1 \cap L_2)L_3 = (L_1L_3) \cap (L_2L_3)$.

   (b) [2] If $L_1 \cup L_2$ is a regular language and $L_1$ is a regular language, then $L_2$ is also a regular language.
(c) [6 points]
i. [1 point] If $L_1, L_2, \ldots, L_{2017}$ are all regular languages, then the language $\bigcap_{i=1}^{2017} L_i$ is regular.

ii. [5 points] If $L_1, L_2, L_3, \ldots$ is an infinite sequence of regular languages, then the language $\bigcap_{i=1}^{\infty} L_i$ is regular.
2. [10 points]

(a) [4] Construct the state-transition diagram of a deterministic finite automaton accepting the following language. Note: \( #_u(v) \) denotes the number of occurrences of a substring \( u \) in a string \( v \).

\[ L = \{ x \in \{a, b\}^* \mid #_a(x) + 2#_b(x) \text{ is divisible by } 3 \}. \]

NOTE: No credits will be given for construction without brief and precise interpretations of the states of your machine.
Consider the following language:

\[ L = \{ x \in \{a, b\}^* \mid x \text{ has both } ab \text{ and } ba \text{ as substrings} \} \].

i. [4 points] Construct a deterministic finite automaton accepting the language \( L \). You may give its 5-tuple formal definition or its transition diagram.

**NOTE:** No credits will be given for construction without brief and precise interpretations of the states of your machine.

ii. [2 points] Give a regular expression denoting the language \( L \).

**NOTE:** No credits will be given for construction without brief and precise annotations.
(c) [15 points]

i. [5 points] Consider the language \( L \subseteq \{0, 1\}^* \) that consists of all strings whose every prefix has at least as many 0s as 1s, that is: for a string \( x = a_1a_2\cdots a_n \) of \( L \) (with \( a_i \in \{0, 1\} \) for all \( i \in \{1, 2, \ldots, n\} \)), each of the \( n+1 \) prefixes \( \epsilon, a_1, a_1a_2, \ldots, a_1a_2\cdots a_n \) has at least as many 0s as 1s. Prove that \( L \) is non-regular by a direct application of the Pumping Lemma for regular languages.
ii. [5 points] For any language $L$ over the alphabet $\Sigma = \{0, 1\}$, we define the operation \textit{unary} over languages as follows: for every string $x \in L$, replace $x$ by a string of 1s of the same length as $x$. Formally, 

$$\text{unary}(L) = \{1^{|x|} \mid x \in L\}.$$ 

Prove or disprove that regular languages are closed under the \textit{unary} operation.
iii. [5 points] For any string $x$ over the alphabet $\Sigma = \{1, 2, 3\}$, we define the operation $sort$ over strings as follows: $sort(x)$ is the string obtained by rearranging all the symbols in $x$ such that all the 1s appear before all the 2s, and all the 2s appear before all the 3s. That is, we sort the symbols in $x$ in the lexicographic order. For any language $L$ over the alphabet $\Sigma$, we define the language:

$$sort(L) = \{sort(x) \mid x \in L\}$$

as the set of “sorted” versions of all the strings of $L$. Prove or disprove that regular languages are closed under the unary operation.