Test 1, Automata, Spring 2025

Problem 1. Why do we need to study automata? Provide two main reasons.

Problems 2–4. Let us consider a 2-state automaton that accepts only words that end on letter a. This automaton has two states: start state s and final state f, and three symbols: a, b, and c. From both states, letter a leads to f, and letters b and c lead to s.

Problem 2. Trace, step-by-step, how this finite automaton will check that the word *baca* belongs to this language. Use the tracing to find the parts x, y, and z of the word *baca* corresponding to the Pumping Lemma. Check that the "pumped" word $xyyz$ will also be accepted by this automaton.

Problem 3. Write down the tuple $\langle Q, \Sigma, \delta, q_0, F\rangle$ corresponding to this automaton:

- Q is the set of all the states,
- Σ is the alphabet, i.e., the set of all the symbols that this automaton can encounter;
- $\delta : Q \times \Sigma \to Q$ is the function that describes, for each state q and for each symbol s, the state $\delta(q, s)$ to which the automaton that was originally in the state q moves when it sees the symbol s (you do not need to describe all possible transitions this way, just describe two of them);
- q_0 is the staring state, and
- F is the set of all final states.

Problem 4. Let A_1 be the automaton described in Problem 2. Let A_2 be an automaton that accepts only sequences that do not have letter b. This automaton has two states: the starting state s which is also final and the error state e. The transitions are as follows: from any state, a and c lead back to the same state and b leads to e. Use the algorithm that we had in class to describe the following two new automata:

- the deterministic automaton that recognizes the union $A_1 \cup A_2$ of the two corresponding languages, and
- the deterministic automaton that recognizes the intersection of the languages A_1 and A_2.

Problem 5. Use the general algorithm that we learned in class to design a non-deterministic finite automaton that recognizes the language $(a \cup b \cup c)^* a$ (that corresponds to all the words that end with a):

- first, describe automata for recognizing a, b, and c;
- then, use the automata for a, b, and c to design an automaton for recognizing $a \cup b \cup c$;
- then, transform the automaton for $a \cup b \cup c$ into an automaton for recognizing the Kleene $(a \cup b \cup c)^*$;
- then, use the automata for $(a \cup b \cup c)^*$ and a to design an automaton for concatenation $(a \cup b \cup c)^* a$.

Problem 6. Use the general algorithm to transform the resulting non-deterministic finite automaton into a deterministic one.

Problem 7-8. Use a general algorithm to transform the finite automaton from Problem 2 into the corresponding regular expression. Start with eliminating the state s.

Problem 9-10. Prove that the language L of all the words that have equal number of a's, b's, and c's is not regular.