Test 1

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

Problem 2–4. Let us consider the automaton that has two states: g (good student) and p (student on probation); g is the starting state, p is the final state. The only three symbols are A, B, and F.

- From g, A and B lead back to g, and F leads to p.
- From p, any symbol leads back to p.

Problem 2. Trace, step-by-step, how this finite automaton will check that the word ABF belongs to this language. Use the above tracing to find the parts x, y, and z of the word ABF 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. Use a general algorithm that we had in class to generate a context-free grammar corresponding to this automaton. Show how this grammar will generate the word ABF.

Problem 5. Let A_1 be the automaton described in Problem 2. Let A_2 be an automaton that accepts all the strings that do not contain As. This automaton has two states: the start state which is also final, and the sink state. The transitions are as follows:

- from the start state, B and F lead back to the start state, while A leads to the sink state;
- from the sink state, any symbol leads back to this state.

Use the algorithm that we had in class to describe the following two new automata:

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

Problem 6. Use the general algorithm that we learned in class to design a non-deterministic finite automaton that recognizes the language $ab \cup a^*$:

- first, describe the automata for recognizing a and b;
- then, combine them into the automata for recognizing the concatenation ab and the Kleene star a^*;
- finally, combine the automata for ab and a^* into an automaton for recognizing the desired union of the two languages.

Problem 7. Use the general algorithm to transform the resulting non-deterministic finite automaton into a deterministic one.

Problem 8–9. Use a general algorithm to transform the finite automaton A_2 from Problem 5 into the corresponding regular expression.

Problem 10. Prove that the language L of all the words that have fewer a's than b's is not regular.