CS 334 Fall 2020: Problem Set 1.

Problem 1. (30 points) Construct deterministic FSAs for each of the following languages over the alphabet $\{a, b\}$:

1. $L_1 = \{w: w\ contains\ the\ string\ aaa\ and\ the\ string\ bbb\}$.

2. $L_2 = \{w: w\ starts\ with\ an\ a\ and\ has\ at\ most\ one\ b.\}$
   Hint: Express L_2 as the intersection of two languages.

3. $L_3 = \{w: the\ 4^{th}\ symbol\ from\ the\ end\ of\ w\ is\ a\}$.
   So, for example, $abbbaaba \in L_3\ \ but\ \ abaababa \notin L_3$. To get started see how trivial it would be if the FSA had to check the last symbol of the input. Then try to design an FSA that must check only the 2nd last symbol. What does each state in this FSA represent? Now extend this idea to design an FSA that accepts L_3.

Problem 2. (10 points) Modify the proof of Theorem 1.25 in the textbook to cover the case when the machines M_1, M_2 have *different* input alphabets Σ_1, Σ_2. Hint: the machine M that recognizes the union of the languages of M_1, M_2 will have input alphabet $\Sigma = \Sigma_1 \cup \Sigma_2$. Take care when defining $\delta((r_1, r_2), a)$ as the symbol a could belong to one alphabet but not the other!

Problem 3. (10 points) Let D_k denote the set of binary strings that represent numbers divisible by k. For example, input strings $0, 00, 000, 10, 010, 0010, 0100010$ are all divisible by 2 (the least significant bit is the rightmost, or last symbol in the sequence), and therefore are all in the language D_2.

1. Construct a deterministic FSA to recognize the language D_3. Hint: Use states to represent remainders of the input seen thus far when divided by 3.
2. Prove that D_k is regular, for every $k \geq 1$.

Optional Exercise. (Present your ideas anytime this term during Sandeep's office hours.) You have been asked to design an FSA to operate a building elevator. How would you go about this design process? What would a state of the FSA correspond to? What should be the alphabet? What constraints would be reasonable to impose on the movement of the elevator? This is an open-ended problem – we don't expect a full design, but rather things you considered and what constraints you found difficult to design for. To get started, imagine a 2-storey building, then a 3-storey building, etc. How do your ideas scale for a very tall building?