CS172 Computability & Complexity, Spring 2021

Homework 1

Out: 22 Jan. Due: 29 Jan.

Instructions: *Submit your solutions in pdf format on Gradescope by* **5pm on Friday, January 29**. *Solutions may be written either in LaTeX (with either machine-drawn or hand-drawn diagrams) or* **legibly** *by hand. (The LaTeX source for this homework is provided in case you want to use it as a template.) Please be sure to begin the solution for each problem on a new page, and to tag each of your solutions to the correct problem! Per course policy, no late solutions will be accepted. Take time to write* **clear** *and* **concise** *answers; confused and long-winded solutions may be penalized. You are encouraged to form small groups (two to four people) to work through the homework, but you* **must** *write up all your solutions on your own. Depending on grading resources, we reserve the right to grade a random subset of the problems and check off the rest; so you are advised to attempt all the problems.*

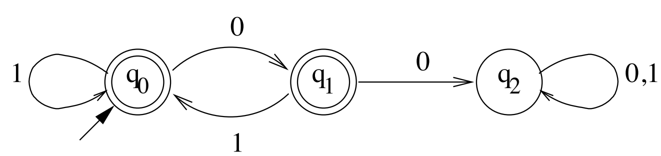
1. Construct *deterministic* finite automata (DFAs) that accept each of the following languages. In each case, specify the five components Q, Σ, δ, q_0 and F of your DFA by drawing a fully and clearly labeled transition diagram.

(a) The set of all 0-1 strings that begin with 0 and end with 1.

(b) The set of all words over the English alphabet $\{a, b, \ldots, z\}$ whose third-last letter is 'b'. [NOTE: Use the abbreviation Σ to denote the English alphabet.]

(c) The set of all 0-1 strings that are the binary encodings of integer multiples of 5. Leading zeros are allowed. (Thus, e.g., the strings 101 and 0101 belong to this language, while 110 does not.)

2. Prove carefully that the language accepted by the following DFA is the set of all 0-1 strings that do not contain a pair of consecutive 0's.



HINT: First, for each state q, write down a *precise* characterization of the set of input strings w that cause the DFA to end in state q. Then prove by induction on the *length* of w that these characterizations are correct.

3. Let L be a regular language over alphabet Σ. Prove that each of the following languages is also regular. In each case, you should assume the existence of a DFA $M = (Q, \Sigma, \delta, q_0, F)$ for L and explain precisely how to modify it to obtain an NFA (or a DFA) for the new language.

(a) $\text{prefix}(L) := \{w \in \Sigma^* \mid \exists z \in \Sigma^* : wz \in L\}$.

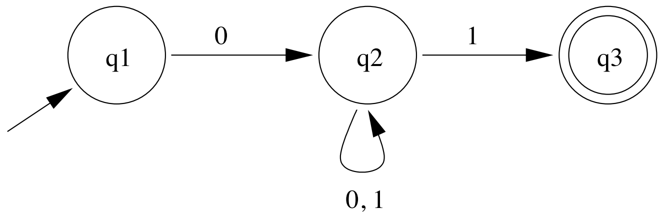
(b) $\text{suffix}(L) := \{w \in \Sigma^* \mid \exists z \in \Sigma^* : zw \in L\}$.

(c) $\text{extend}(L) := \{w \in \Sigma^* \mid \exists x \in L, \exists y \in \Sigma^* : w = xy\}$.

(d) $\text{prepend}(L) := \{w \in \Sigma^* \mid \exists x \in L, \exists y \in \Sigma^* : w = yx\}$.

[continued on next page]

4. Convert the following NFA into an equivalent DFA using the "subset construction" discussed in class.



What is the relationship between this DFA and the one you constructed for problem 1(a)?