CSE 105 - Theory of Computation (Winter 2026)
Homework 3: Nonregular Languages
Due Wednesday, January 28, 11:59pm

Instructions: There are two parts to this homework:

- Concept check (Question 1): Every student must complete this individually on Gradescope.
- Written Homework (Question 2-4): You may work individually or in a team of up to 3 people. Please ensure your name(s) and PID(s) are clearly visible on the first page of your submission, and then upload the PDF to Gradescope. If working in a group, submit only one submission per group: one partner uploads the submission through their Gradescope account and then adds the other group member to the Gradescope submission by selecting their name in the "Add Group Members" dialog box. You will need to re-add your group member every time you resubmit a new version of your assignment.

It is highly recommended (though not required) that you type your answers. It is your responsibility to make any handwriting clear and legible for grading. A LaTeX template for the homework is provided on Canvas. For ease of grading, please start each new problem on a separate page.

We will only be grading some of the problems below for correctness. However, because all of the concepts are important, we will not reveal which problems are being graded for correctness until after the assignment has been submitted. The remaining problems will be graded for completeness (i.e., does it look like there was a good-faith effort to solve the problem?).

Reading and extra practice problems: Sipser Section 1.4. Chapter 1 exercises 1.29, 1.30. Chapter 1 problems 1.49

Problems:

1. **Concept check**

   Complete the assignment "Homework 3 - Concept Check" on Gradescope.

2. **Regular or not?**

   Fix alphabet $\Sigma = \{0, 1\}$. For each of the languages listed below, prove that it is either regular or nonregular.

   For each language that is regular, a complete solution will include a precise definition of a DFA, NFA, or regular expression that recognizes or describes it, along with a brief justification of your construction by explaining the role each state plays in the machine or referring back to relevant definitions.

   For each language that is nonregular, a complete solution will use the pumping lemma to prove it, including appropriate justification related to the specific language.

(a) $L_1 = \{0^n x 1^n \mid n \geq 1, x \in \Sigma^*\}$, a language over Σ.

(b) $L_2 = \{0^n 1x01^n \mid n \geq 1, x \in \Sigma^*\}$, a language over Σ.

(c) $L_3 = \{xx \mid x \in \Sigma^*\}$.

3. **Properties of nonregular languages**

Prove or disprove each of the following statements. In other words, decide whether each statement is true or false and justify your decision. Let $\Sigma = \{0, 1\}$.

(a) For all languages L, K over Σ, if L is nonregular and K is finite, then $L - K$ is nonregular. Recall: $L - K = \{w \in \Sigma^* \mid w \in L \text{ and } w \notin K\}$.

(b) Every infinite language over Σ where each string in the language has an equal number of 0's and 1's is nonregular.

(c) Recall that for language K over Σ,

$$\textsc{Substring}(K) := \{w \in \Sigma^* \mid \text{there exist } a, b \in \Sigma^* \text{ such that } awb \in K\}.$$

For every nonregular language K over Σ, $\textsc{Substring}(K)$ is nonregular.

4. **Pumping dilemma**

Your friend claims that the Pumping Lemma is useless for proving that an infinite language $K \subseteq \Sigma^*$ is not regular. Their logic goes like this

(Step 1) Suppose that K is regular. It can be recognized by a DFA $M = (Q, \Sigma, \delta, q_0, F)$.

(Step 2) For arbitrary DFA M, the pumping length p is at least $|Q|$.

(Step 3) However, for every integer $n \geq |Q|$, there exists a machine $M' = (Q', \Sigma, \delta', q_0', F')$ such that $L(M') = L(M) = K$ and $|Q'| = n$.

(Step 4) Therefore, the Pumping Lemma cannot be used to pump any string of finite length since its pumping length might be arbitrarily large.

Below, we will examine the steps above in detail. Justify your answer to each part.

(a) (Step 1): Is this statement true? In other words, just because we're assuming that K is regular a regular language, does it mean we can assume there is a DFA that recognizes it?

(b) (Step 2): In general, it's true that the smallest the pumping length of a language recognized by a DFA with states Q can be is $|Q|$. Prove this by finding a specific infinite language K and a DFA recognizing it that cannot have pumping length smaller than $|Q|$.

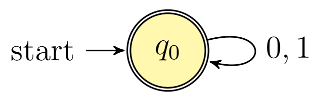
(c) (Step 3): This step is correct; prove the stated version of this statement: For every integer $n \geq |Q|$, there exists a machine $M' = (Q', \Sigma, \delta', q_0', F')$ such that $L(M') = L(M)$ and $|Q'| = n$.

(*Challenge; not graded*): Define a *cycle* to be a sequence of *distinct* states $q_1, q_2, \ldots, q_m$ such that

$$\delta(q_1, \sigma_1) = q_2, \qquad \delta(q_2, \sigma_2) = q_3, \qquad \ldots, \qquad \delta(q_m, \sigma_m) = q_1,$$

where $\sigma_1, \sigma_2, \ldots, \sigma_m \in \Sigma$ are symbols in the alphabet. An objection to the statement in (Step 3) is that the proof of the Pumping Lemma depends on the length of the cycles in the DFA rather than the number of states. That is, increasing the number of states in your DFA might not increase the pumping length because the length of the smallest cycle stays the same. Nevertheless, a version of your friend's statement is still true whenever you impose this additional cycle constraint: for every integer $n \geq |Q|$, there exists a machine $M' = (Q', \Sigma, \delta', q_0', F')$ such that $L(M') = L(M)$ and the length of the smallest cycle in the M' is at least n.

Your task is to show that even this more general statement is true for the simple language Σ^* recognized by the DFA below:



For all $n \geq 1$, define a DFA for this language where the length of the smallest cycle is n.

(d) (Step 4): Describe why this statement is true/false/misleading.