

INSTRUCTIONS

One member of the group should upload your group submission to Gradescope. During the submission process, they will be prompted to add the names of the rest of the group members. All group members' names and PIDs should be on **each** page of the submission.

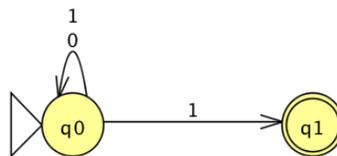
Your homework must be typed. We recommend that you submit early drafts to Gradescope so that in case of any technical difficulties, at least some of your work is present. You may update your submission as many times as you'd like up to the deadline.

Your assignments in this class will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should always explain how you arrived at your conclusions, using mathematically sound reasoning. Whether you use formal proof techniques or write a more informal argument for why something is true, your answers should always be well-supported. Your goal should be to convince the reader that your results and methods are sound.

READING Sipser Section 1.1, 1.2

KEY CONCEPTS Regular languages, closure of the class of regular languages under certain operations, nondeterministic finite automata (NFA), nondeterministic computation, spontaneous transitions (ε arrows).

1. (10 points) Let N be the NFA recognizing E_1 from this week's individual homework:



- Apply the construction defined in class to prove that the class of regular languages is closed under concatenation (and in the textbook in Theorem 1.47, page 60) to N to obtain a NFA for $E_1 \circ E_1$. Draw the resulting state diagram in JFLAP, export the image as a png or jpg file, and include it as part of your submission.
- Apply the construction defined in class to prove that the class of regular languages is closed under Kleene star (and in the textbook in Theorem 1.49, page 62) to N to obtain a NFA for E_1^* . Draw the resulting state diagram in JFLAP, export the image as a png or jpg file, and include it as part of your submission.

Note: You can use this example to justify why we need to add a new start state when forming the automaton for the Kleene star of the language. Can you see why?

- Apply the construction defined in class to prove that NFA can be determinized (and in the textbook in Theorem 1.39, page 55) to the NFA you obtained in b. to obtain a DFA that recognizes E_1^* . You do not need to include unreachable states. Draw the resulting state diagram in JFLAP, export the image as a png or jpg file, and include it as part of your submission.

2. Show* that the class of regular languages over the alphabet $\{0,1\}$ is closed under the operation $STUTTER(L)$, defined as

$$STUTTER(L) = \{w_1w_1w_2w_2 \cdots w_nw_n \mid n \geq 0, w_i \in \Sigma \text{ for each } i \text{ where } 1 \leq i \leq n, \text{ and } w_1w_2 \cdots w_n \in L\}.$$

Here's the setup: consider an arbitrary DFA $M = (Q, \{0,1\}, \delta, q_0, F)$ and call the language of this DFA L . Your job is to complete the construction stage: define a new DFA or NFA whose language is $STUTTER(L)$ by giving precise (but general) definitions for each parameter in the formal definition of $M' = (Q', \{0,1\}, \delta', q', F')$. It is up to you whether to use nondeterminism in your construction, but be careful about the **types** of objects in your formal definition and make them consistent with your choice.

(*) A full proof would have three stages: setup, construction, and proof of correctness; you do not need to hand in a proof of correctness (though it's a good idea to work through it to check your construction). To prove that the construction of correct, we would need to prove that $L(M') = STUTTER(L)$ for any L . Two claims are required

- (1) Assume that some string, call it w , is accepted by M' . Prove that w is in $STUTTER(L)$.
- (2) Assume that some string, call it y , is in $STUTTER(L)$. Prove that y is accepted by M' .

You can practice your proof techniques by carrying out this justification.

3. (10 points) Let

$$\Sigma_2 = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \right\}$$

Here, Σ_2 contains all columns of 0s and 1s of height two. A string of symbols in Σ_2 gives two rows of 0s and 1s of equal length. Consider each row to represent a binary number (potentially with leading 0s). For example, the string $\begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ represents the two binary numbers $(101)_2 = 5$ (top row) and $(011)_2 = 3$ (bottom row). Design a NFA over Σ_2 that recognizes the language

$$\{w \in \Sigma_2^* \mid \text{the number represented by the bottom row is twice that represented by the top row}\}$$

For example, $\begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ should not be accepted (because 3 is not two times 5) but $\begin{pmatrix} 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ should be accepted (because $4 = 2 \cdot 2$).

Draw the state diagram of your NFA in JFLAP, export the image as a png or jpg file, and include it as part of your submission. In JFLAP, you can represent elements in Σ_2 as two bits separated by a slash. You do not need to justify your construction for credit, **but** if you describe how your state diagram works by briefly describing the role of each state and the transitions between them we may be able to award partial credit if your answer is incorrect.

Note: you only need two states in your NFA.