

Theory of Computation
CS 378, Fall 2022

Smaller Assignment 6

Due **BY THE BEGINNING OF CLASS** Thursday, October 20

- Standard file naming and HW submission conventions apply: Please submit your type-written answers in a PDF file named

CS378_SA6_<userid>.pdf

where <userid> is replaced by your Colby userid—for example, my file would be called CS378_SA6_eaaron.pdf. Please use your full userid, including your class year. Place the properly named file in your **Private** directory in your CS378 filespace on **filer**.

- This smaller assignment is based on material initially introduced in our Oct. 13 class meeting, and continued in our Oct. 18 class meeting. This is due by the beginning of the next class meeting after Oct. 18, and it will be marked on a ✓ + /✓/✓ − /0 scale.
- *A general note for CS378:* As always, please present answers cleanly and **explain them clearly and thoroughly**, giving all details needed to make your answers easy to understand; typed-up (rather than handwritten) answers are required, and using L^AT_EX to typeset your answers is appreciated. (Feel free to talk with your Prof. about using L^AT_EX!) Graders may not award full credit to incomplete or illegible solutions. Clear communication *is* the point, on every assignment.

In general in CS378, unless explicitly specified otherwise, answers should be accompanied by explanations. Answers without explanations may not receive full credit. Please feel free to ask me any questions about explanations that might come up!

Exercises

1. **State Minimization and More Dynamic Programming!** Use the dynamic programming state minimization algorithm from class to construct a DFA that accepts the language accepted by the DFA in Figure 1, but has the minimum possible number of states. (*Note:* As noted in our lecture notes, pages from the HMU textbook that are posted on our Lecture Notes page can be a resource for this exercise; if there are differences between the method presented in class and the one in HMU, please be sure to follow the method from class.) Show your dynamic programming table with a **numerical index** in each “checked” cell indicating the length of the shortest distinguishing string for that pair of states. After you’ve built the table, draw the transition diagram for the new minimized DFA.

For this exercise, an explanation will be sufficient if it gives two illustrative examples of how a non-zero number was added to your dynamic programming table and gives two illustrative examples of how transitions were decided for the minimized DFA δ function. As with choosing test cases for code correctness, please be sure to choose the examples so that they are usefully illustrative of a correct answer, demonstrating that the answer shows full understanding of the state minimization algorithm. (That is, make sure they show what you know!)

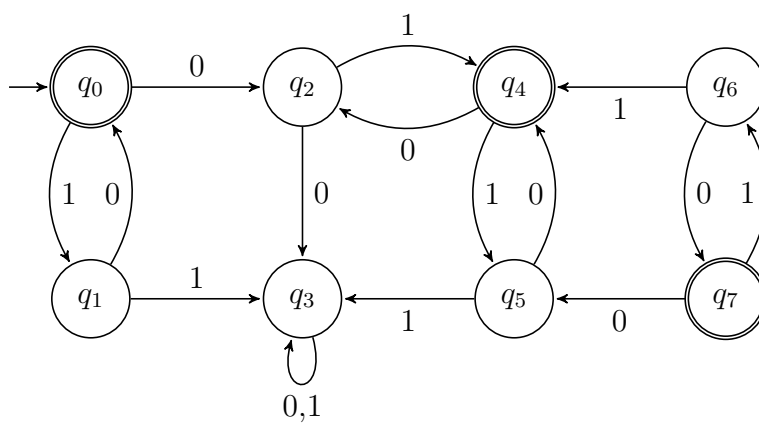


Figure 1: The DFA for the state minimization algorithm in Exercise 1.