Homework 2: DFAs and NFAs

Theory of Computing (CSE 30151), Spring 2026

Due 2026-01-30 5pm

Instructions

- Create a PDF file (or files) containing your solutions. You can write your solutions by hand, but please scan them into a PDF.
- Please name your PDF file(s) as follows to ensure that the graders give you credit for all of your work:
  - If you're making a complete submission, name it `*netid*-hw2.pdf`, where `*netid*` is replaced with your NetID.
  - If you're submitting some problems now and want to submit other problems later, name it `*netid*-hw2-part123.pdf`, where `123` is replaced with the problem numbers you are submitting at this time.
- Submit your PDF file(s) in Canvas.

Problems (10 points each)

1. **Divisibility tests.** Define, for all $k > 0$,

$$D_k = \{w \in \{\texttt{0}, \ldots, \texttt{9}\}^* \mid w \text{ is the decimal representation of a multiple of } k\}$$

   where ε is considered to represent the number 0. For example, the strings ε, `0`, `1234`, and `01234` all belong to D_2, but `99` and `099` do not.

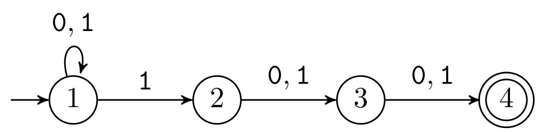
   (a) Prove that D_2 is regular by writing a DFA for D_2.

   (b) Prove that D_3 is regular by writing a DFA for D_3.

   (c) Prove that D_6 is regular. An explicit DFA is not necessary.

   (∗) Optional alternative: You can get full credit for all of the above if you can prove that for any $k > 0$, D_k is regular, by describing how to write the formal description of a DFA $M = (Q, \{\texttt{0}, \ldots, \texttt{9}\}, \delta, s, F)$ in terms of k.

2. **Nondeterminism.** Consider the following NFA N_2 (same as in Figure 1.31), which accepts a string iff the third-to-last symbol is a `1`:



   (a) Draw the computation of N_2 on input `1110100`, in the style of Sipser's Figure 1.29.

   (b) Use the subset construction (Theorem 1.39) to convert N_2 to a DFA M. You may omit curly braces and commas when naming states; for example, instead of $\{1, 2, 3, 4\}$ you may write 1234. (Hint: the DFA should be equivalent to the one in Figure 1.32.)

   (c) List, in order, the states that M goes through on input `1110100`.

   (d) Why do you think the states in Figure 1.32 are named the way they are?

3. **Procrustean closure properties.**

   (a) For any string $w = w_1 w_2 \cdots w_{n-1} w_n$, define

$$\text{STRETCH}(w_1 w_2 \cdots w_{n-1} w_n) = w_1 w_1 w_2 w_2 \cdots w_{n-1} w_{n-1} w_n w_n.$$

   For example,

$$\text{STRETCH}(\varepsilon) = \varepsilon$$
$$\text{STRETCH}(\texttt{abab}) = \texttt{aabbaabb}.$$

   This induces an operation on languages,

$$\text{STRETCH}(L) = \{\text{STRETCH}(w) \mid w \in L\}.$$

   For example,

$$\text{STRETCH}(\{\varepsilon, \texttt{abab}\}) = \{\varepsilon, \texttt{aabbaabb}\}.$$

   Prove that for any regular language L, STRETCH(L) is also regular.

   (b) For any $w = w_1 w_2 \cdots w_{n-1} w_n$ with $n \geq 2$, define

$$\text{CHOP}(w_1 w_2 \cdots w_{n-1} w_n) = w_2 \cdots w_{n-1}.$$

   For example,

$$\text{CHOP}(\texttt{ab}) = \varepsilon$$
$$\text{CHOP}(\texttt{abab}) = \texttt{ba}$$
$$\text{CHOP}(\texttt{bbaa}) = \texttt{ba}.$$

This induces an operation on languages,

$$\text{CHOP} = \{\text{CHOP}(w) \mid w \in L \text{ and } |w| \geq 2\}.$$

For example,

$$\begin{aligned}\text{CHOP}(\{\texttt{ab}, \texttt{abab}, \texttt{bbaa}\}) &= \{\varepsilon, \texttt{ba}\} \\ \text{CHOP}(\{\varepsilon, \texttt{a}\}) &= \emptyset.\end{aligned}$$

Prove that for any regular language L, $\text{CHOP}(L)$ is also regular.