Some important course policies

- **You may work in groups of up to three people.** However, each problem should be submitted by exactly one person, and the beginning of the homework should clearly state the names and NetIDs of each person contributing.

- **You may use any source at your disposal**—paper, electronic, or human—but you **must** cite every source that you use. See the academic integrity policies on the course web site for more details.

- **Submit your pdf solutions in Moodle.** See instructions on the course website and submit a separate pdf for each problem. Ideally, your solutions should be typeset in LaTeX. If you hand write your homework make sure that the pdf scan is easy to read. Illegible scans will receive no points.

- **Avoid the Three Deadly Sins!** There are a few dangerous writing (and thinking) habits that will trigger an automatic zero on any homework or exam problem. Yes, we are completely serious.
  - Give complete solutions, not just examples.
  - Declare all your variables.
  - Never use weak induction.

- Unlike previous editions of this and other theory courses we are not using the “I don't know” policy.

See the course web site for more information.

If you have any questions about these policies, please don't hesitate to ask in class, in office hours, or on Piazza.
1. (a) Draw an NFA with alphabet \( \{0, 1\} \), that accepts the language \( \{w \mid w \text{ contains exactly one maximal substring of 0s of odd length}\} \). (\( u \in \{0\}^+ \) is a maximal substring of 0s of \( w \) if \( u \) occurs in \( w \) such that there is no 0 immediately to the left or right of \( u \).) Note that the language does have strings with any number of maximal substrings of 0s of even length (but exactly one of odd length).

(b) i. Draw an NFA for the regular expression \((100)^* + 10^*\).

ii. Now apply the powerset construction (also called the subset construction) to your NFA, to obtain a DFA for the same language. Label the states of your DFA with names that are sets of states of your NFA. (Note that no points will be awarded for a DFA construction that is not derived from your NFA as directed.)

2. Let \( L \) be a regular language, accepted by a DFA \( M = (\Sigma, Q, \delta, s, F) \). Show that \( L^2 = \{ww \in L \mid w \in L\} \) is also a regular language by constructing an NFA \( N = (\Sigma, Q', \delta', s', F') \) that accepts it. You should formally specify (using precise mathematical notation) the components of \( N \) in terms of those of \( M \). Be sure to describe in English how your NFA works.

[Hint: Your NFA could “guess” which state \( M \) will be in when it finishes seeing \( w \).]

3. Suppose \( N_1 = (\Sigma, Q_1, \delta_1, s_1, F_1) \) and \( N_2 = (\Sigma, Q_2, \delta_2, s_2, F_2) \) are two NFAs (possibly with \( \epsilon \)-moves) such that \( F_1 = \{f_1\} \) and \( F_2 = \{f_2\} \). Assume that \( Q_1 \cap Q_2 = \emptyset \). Below \( N_{\text{comp}}, N_{\star}, N_{\text{cat}} \) and \( N_{\text{union}} \) are purported constructions for NFAs accepting \( L(N_1), L(N_1)^*, L(N_1) L(N_2) \) and \( L(N_1) \cup L(N_2) \) respectively. You should give counter-examples in each case.

(a) Let \( N_{\text{comp}} \) be obtained by swapping the sets of final and non-final states. Formally, let \( N_{\text{comp}} = (\Sigma, Q_1, \delta_1, s_1, Q_1 \setminus F_1) \).

Give an example of \( N_1 \) such that \( L(N_{\text{comp}}) \neq L(N_1) \). (Describe the languages \( L(N_{\text{comp}}) \) and \( L(N_1) \) explicitly.)

(b) Let \( N_{\star} \) be obtained from \( N_1 \) by adding an \( \epsilon \)-move from \( f_1 \) to \( s_1 \), and also setting the start state to be a final state. Formally, let \( N_{\star} = (\Sigma, Q_1, \delta, s_1, F) \), where \( F = \{f_1, s_1\} \) and \( \delta : Q \times (\Sigma \cup \{\epsilon\}) \rightarrow \mathcal{P}(Q) \) is defined as follows.

\[
\delta(q,a) = \begin{cases} 
\delta_1(f_1, \epsilon) \cup \{s_1\} & \text{if } (q,a) = (f_1, \epsilon) \\
\delta_1(q,a) & \text{otherwise}
\end{cases}
\]

Give an example of \( N_1 \) such that \( L(N_{\star}) \neq L(N_1)^* \). (Describe the languages \( L(N_{\star}) \) and \( L(N_1)^* \) explicitly.)
In the next two parts, we write replace\((S, q, q')\) to denote the set obtained by modifying a set \(S\) by replacing \(q\) by \(q'\) if it occurs in \(S\): i.e.,

\[
\text{replace}(S, q, q') = \begin{cases} 
S & \text{if } q \not\in S, \\
S \cup \{q'\} - \{q\} & \text{if } q \in S.
\end{cases}
\]

(c) Let \(N_{\text{cat}}\) be obtained by merging the only final state \(f_1\) of \(N_1\) with the start state \(s_2\) of \(N_2\) (to obtain a state denoted by \(f_1\)), and setting \(f_2\) to be the final state. Formally, let \(N_{\text{cat}} = (\Sigma, Q, \delta, s, F)\) where \(Q = Q_1 \cup Q_2 - \{s_2\}\), \(s = s_1, F = F_2\) and \(\delta : Q \times (\Sigma \cup \{\epsilon\}) \to \mathcal{P}(Q)\) is defined as follows.

\[
\delta(q, a) = \begin{cases} 
\delta_1(f_1, a) \cup \text{replace}(\delta_2(s_2, a), s_2, f_1) & \text{if } q = f_1 \\
\delta_1(q, a) & \text{if } q \in Q_1 - \{f_1\} \\
\text{replace}(\delta_2(q, a), s_2, f_1) & \text{if } q \in Q_2 - \{s_2\}
\end{cases}
\]

Give an example of \(N_1\) and \(N_2\) such that \(L(N_{\text{cat}}) \neq L(N_1) \cup L(N_2)\). (Describe the languages \(L(N_{\text{cat}})\) and \(L(N_1) \cup L(N_2)\) explicitly.)

(d) Let \(N_{\text{union}}\) be obtained by merging the start states of \(N_1\) and \(N_2\) into a single state (denoted by \(s_1\)), and setting \(f_1\) and \(f_2\) to be the final states. Formally, let \(N_{\text{union}} = (\Sigma, Q, \delta, s, F)\), where \(Q = Q_1 \cup Q_2 - \{s_2\}\), \(s = s_1, F = F_1 \cup F_2\) and \(\delta : Q \times (\Sigma \cup \{\epsilon\}) \to \mathcal{P}(Q)\) is defined as follows.

\[
\delta(q, a) = \begin{cases} 
\delta_1(s_1, a) \cup \text{replace}(\delta_2(s_2, a), s_2, s_1) & \text{if } q = s_1 \\
\delta_1(q, a) & \text{if } q \in Q_1 - \{s_1\} \\
\text{replace}(\delta_2(q, a), s_2, s_1) & \text{if } q \in Q_2 - \{s_2\}
\end{cases}
\]

Give an example of \(N_1\) and \(N_2\) such that \(L(N_{\text{union}}) \neq L(N_1) \cup L(N_2)\). (Describe the languages \(L(N_{\text{union}})\) and \(L(N_1) \cup L(N_2)\) explicitly.)