CS/ECE 374 A ✧ Spring 2026
Conflict Midterm 1 Problem 1 Solution

For each statement below, check "Yes" if the statement is ***always*** true and check "No" otherwise. In either case, give a brief (one short sentence) explanation of your answer. Read these statements very carefully—small details matter!

(a) There exists a DFA for every finite language.

[X] Yes [ ] No — Every finite language is regular.

(b) If L is regular, then $L \cap L'$ is regular for any language L'.

[ ] Yes [X] No — Consider $L = \Sigma^*$ and $L' = \{\mathtt{0}^n \mathtt{1}^n \mid n \geq 0\}$.

(c) If L is regular, then any $L' \subseteq L$ is regular.

[ ] Yes [X] No — Consider $L = \Sigma^*$ and $L' = \{\mathtt{0}^n \mathtt{1}^n \mid n \geq 0\}$

(d) If L has a fooling set of size 374, then every DFA for L requires at least 374 states.

[X] Yes [ ] No — Needed to send all members of fooling set to distinct states.

(e) If L has a fooling set of size 374, then there exists a DFA for L with 374 states.

[ ] Yes [X] No — There could be another larger fooling set.

(f) If F is a fooling set of a context-free language L, then F contains an infinite length string.

[ ] Yes [X] No — By definition, all strings have finite length.

(g) For language $\left\{\mathtt{0}^a \mathtt{1}^b \mathtt{0}^a \mid b > 0\right\}$, there exists a fooling set F such that F is regular.

[X] Yes [ ] No — Consider $F = \varnothing$.

(h) For language $\left\{\mathtt{0}^a \mathtt{1}^b \mathtt{0}^a \mid b > 0\right\}$, every fooling set is regular.

[ ] Yes [X] No — Consider $F = \{\mathtt{0}^a \mathtt{1}^a \mid a > 0\}$.

(i) If the *smallest* DFAs for regular languages L_1 and L_2 have k_1 and k_2 many states respectively, then the smallest DFA for language $L_1 \cap L_2$ has $(k_1 * k_2)$ states. Here, by *smallest* we mean with the fewest number of states.

[ ] Yes [X] No — Let $L_1 \left\{0^{2k} \mid k \geq 0\right\}$ and $L_2 = \left\{0^{2k+1} \mid k \geq 0\right\}$. Then, $k_1 = k_2 = 2$ but $L_1 \cap L_2 = \varnothing$ has a DFA with one state that rejects.

(j) If language L is regular then language $L' = \{y \mid xy \in L, |x| \geq 374\}$ is regular.

[X] Yes [ ] No — Guess *at least* 374 unseen symbols to feed to L's DFA and then continue simulating with the input string.

CS/ECE 374 A ✧ Spring 2026
Conflict Midterm 1 Problem 2 Solution

For an even length string $w \in \Sigma^*$, we define $\mathsf{Moderate}(w)$ to be the function that divides w into pairs of symbols and replaces each pair ab with the result of the NAND operation $a \barwedge b$.

(a) ***Prove*** that if $L \subseteq \Sigma^*$ is regular, then $\textsc{UnModerated}(L) := \{w \in \Sigma^* \mid |w| \text{ is even and } \mathsf{Moderate}(w) \in L\}$ is also regular.

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA for L. We construct a DFA $M' = (Q', S', A', \delta')$ for $\textsc{UnModerated}(L)$ as follows.

$$Q' = Q \times \{0, 1, \varepsilon\}$$
$$S' = (s, \varepsilon)$$
$$A' = \{(q, \varepsilon) \mid q \in A\}$$

$$\delta'((q, \varepsilon), a) = (q, a), \qquad \forall q \in Q, a \in \{0, 1\}$$
$$\delta'((q, a), b) = (\delta(q, a \barwedge b), \varepsilon) \qquad \forall q \in Q, a, b \in \{0, 1\}$$

State (q, ε) in M' indicates we are about to read a where $(a \barwedge b)$ is to be read in M. (q, a) state indicates we have read a and are about to read b. Once we read b at (q, a) in M', we transition in M along $(a \barwedge b)$ from q. ■

Rubric: 5 points: standard language-transformation rubric (scaled)

(b) ***Prove*** that if $L \subseteq \Sigma^*$ is regular, then $\textsc{Moderated}(L) := \{\mathsf{Moderate}(w) \mid |w| \text{ is even and } w \in L\}$ is also regular.

Solution: Let $M = (Q, s, A, \delta)$ be an arbitrary DFA for L. We construct an **NFA** $M' = (Q', S', A', \delta')$ for $\textsc{Moderated}(L)$ as follows.

$$Q' = Q$$
$$S' = s$$
$$A' = A$$

$$\delta'(q, 0) = \{\delta(\delta(q, 1), 1)\} \quad \forall q \in Q$$
$$\delta'(q, 1) = \{\delta(\delta(q, 0), 0), \delta(\delta(q, 0), 1), \delta(\delta(q, 1), 0)\} \quad \forall q \in Q$$

At every state q, to transition on reading a in M', we feed every possible combination of two characters to M from state q whose NAND is a. ■

Rubric: 5 points: standard language-transformation rubric (scaled)

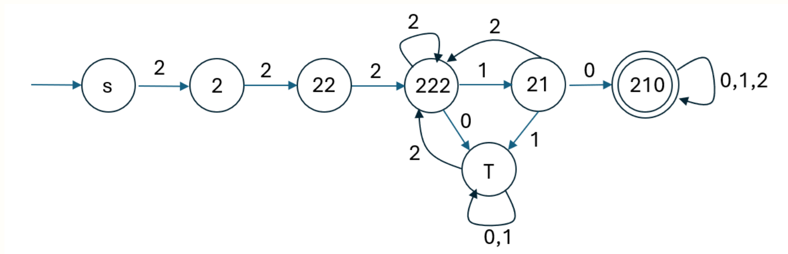
CS/ECE 374 A ✧ Spring 2026
Conflict Midterm 1 Problem 3 Solution

For each of the following languages over the alphabet $\Sigma = \{0, 1, 2\}$, describe both a regular expression that matches the language and a DFA that accepts the language. You do not need to prove that your answers are correct.

(a) All strings in Σ^* that start with 222 and have 210 as a substring.

Solution:

$$\big(222(0+1+2)^*210 + 22210\big)(0+1+2)^*$$



All missing transitions in the DFA go to a hidden dump state. The states are labeled as follows:

- 222: String starts with prefix 222.
- 21: Substring 21 encountered.
- 210: Substring 210 is encourntered.

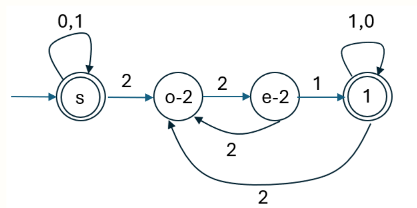
■

Rubric: 5 points = 2½ for regular expression + 2½ for DFA, standard rubrics (scaled). The state labels and explanations are not required for credit.

(b) All strings in Σ^* where each run of 2s contains the substring 22 an odd number of times and is followed by a run of 1s.

Solution: Note that a run of 2s has substring 22 odd number of times if and only if the run is of even length.

$$(0+1)^*((22)^*1(0+1)^*)^*$$



All missing transitions in the DFA go to a hidden dump state. The states are labeled with the last few symbols read:

- s: We are are reading $(0+1)^*$ or haven't read anything yet.
- o-2: We are in a run of 2s, and we have seen odd number of 2s so far within this run.
- e-2: We are in a run of 2s, and we have seen even number of 2s so far within this run.
- 1: We are in a run of 1s after a valid run of 2s.

■

Rubric: 5 points = 2½ for regular expression + 2½ for DFA, standard rubrics (scaled). The state labels and explanations are not required for credit.

CS/ECE 374 A ✧ Spring 2026
Conflict Midterm 1 Problem 4 Solution

Prove $\#(0, w) \mod 2 = |w| + \mathsf{NumOdd1s}(w) + \mathsf{NumEven1s}(w) \mod 2$ for all strings $w \in \{0, 1\}^*$ where the functions NumOdd1s and NumEven1s are defined in the question handout.

Solution: The statement must be interpreted to apply the mod 2 to the entire right-hand side of the equation; otherwise, you'd be trying to prove w has strictly more 0s than it has symbols in some cases. The proof includes extra parentheses to make the above clear.

Let w be an arbitrary string in Σ^*. Assume for all strings x shorter than w that $\#(0, x) \mod 2 = (|x| + \mathsf{NumOdd1s}(x) + \mathsf{NumEven1s}(x)) \mod 2$. There are three cases to consider:

- Suppose $w = \varepsilon$. Then

$$\begin{aligned}
\#(0, w) \mod 2 &= \#(0, \varepsilon) \mod 2 && w = \varepsilon \\
&= 0 \mod 2 && \text{def. } \# \\
&= (0 + 0 + 0) \mod 2 && \text{arithmetic} \\
&= (|\varepsilon| + \mathsf{NumOdd1s}(\varepsilon) + \mathsf{NumEven1s}(\varepsilon)) \mod 2 && \text{defs. } |\cdot|, \text{NO1s, NE1s} \\
&= (|w| + \mathsf{NumOdd1s}(w) + \mathsf{NumEven1s}(w)) \mod 2 && w = \varepsilon
\end{aligned}$$

- Suppose $w = a$ for some symbol $a \in \Sigma$. Then

$$\begin{aligned}
\#(0, w) \mod 2 &= \#(0, a) \mod 2 && w = a \\
&= ([a = 0]) \mod 2 && \text{def. } \# \\
&= ([a = 0] + [a = 1] + [a = 1] + 0) \mod 2 && \text{modular arithmetic} \\
&= (1 + [a = 1] + 0) \mod 2 && a \in \{0, 1\} \\
&= (|a| + \mathsf{NumOdd1s}(a) + \mathsf{NumEven1s}(a)) \mod 2 && \text{def. } |\cdot|, \text{NO1s, NE1s} \\
&= (|w| + \mathsf{NumOdd1s}(w) + \mathsf{NumEven1s}(w)) \mod 2 && w = a
\end{aligned}$$

- Suppose $w = abx$ for some string x and symbols $a, b \in \Sigma$. Then

$$\begin{aligned}
\#(\mathtt{0}, w) \mod 2 &= \#(\mathtt{0}, abx) \mod 2 && w = abx \\
&= ([a = \mathtt{0}] + [b = \mathtt{0}] + \#(\mathtt{0}, x)) \mod 2 && \text{def. } \#\text{, arithmetic} \\
&= ([a = \mathtt{0}] + [b = \mathtt{0}] + (\#(\mathtt{0}, x) \mod 2)) \mod 2 && \text{modular arithmetic} \\
&= ([a = \mathtt{0}] + [b = \mathtt{0}] + \\
&\qquad ((|x| + \mathsf{NumOdd1s}(x) + \mathsf{NumEven1s}(x)) \mod 2)) \mod 2 && \textbf{ind. hyp.} \\
&= ([a = \mathtt{0}] + [a = \mathtt{1}] + [b = \mathtt{0}] + [b = \mathtt{1}] + |x| + \\
&\qquad [a = \mathtt{1}] + \mathsf{NumOdd1s}(x) + \\
&\qquad [b = \mathtt{1}] + \mathsf{NumEven1s}(x)) \mod 2 && \text{modular arithmetic} \\
&= (2 + |x| + \\
&\qquad [a = \mathtt{1}] + \mathsf{NumOdd1s}(x) + \\
&\qquad [b = \mathtt{1}] + \mathsf{NumEven1s}(x)) \mod 2 && a \in \{\mathtt{0}, \mathtt{1}\} \\
&= (|abx| + \mathsf{NumOdd1s}(abx) + \mathsf{NumEven1s}(abx)) \mod 2 && \text{def. } |\cdot|\text{, NO1s, NE1s} \\
&= (|w| + \mathsf{NumOdd1s}(w) + \mathsf{NumEven1s}(w)) \mod 2 && w = abx
\end{aligned}$$

In all three cases, we conclude that. $\#(\mathtt{0}, w) \mod 2 = (|w| + \mathsf{NumOdd1s}(w) + \mathsf{NumEven1s}(w)) \mod 2$. ■

Rubric: 10 points: standard induction rubric. This solution is more detailed than necessary for full credit.

CS/ECE 374 A ✧ Spring 2026
Conflict Midterm 1 Problem 5 Solution

Let $L \subseteq \{0, 1, 2\}^*$ be a set of strings w in which 0s and 2s appear the same number of times and no prefix of w contains strictly more 2s than 0s.

(a) Prove that L is not a regular language.

Solution: Consider the set $F = 0^*$.

Let x and y be distinct strings in F.

Then $x = 0^i$ and $y = 0^j$ for some integers $i \neq j$.

Let $z = 2^i$.

- $xz = 0^i 2^i \in L$, because 0 and 2 each appear exactly i times, and no prefix of xz contains strictly more 2s than 0s.
- $yz = 0^j 2^i \notin L$, because 0 appears j times, 2 appears i times, and $i \neq j$.

We conclude that F is a fooling set for L.

Because F is infinite, L cannot be regular. ■

Rubric: 5 points: Standard fooling set rubric. This is not the only correct solution.

(b) Describe a context-free grammar for L.

Solution:

$$S \rightarrow A \mid A\,0\,S\,2\,S$$
$$A \rightarrow \varepsilon \mid 1\,A \qquad \text{i.e. } 1^*$$

L can be interpreted as the set of balanced parantheses except with 0s and 2s along with optional 1s scattered about. ■

Solution:

$$S \rightarrow A \mid A\,0\,S\,2\,A \mid SS$$
$$A \rightarrow \varepsilon \mid 1\,A \qquad \text{i.e. } 1^*$$

■

Rubric: 5 points: Standard CFG rubric. These are not the only correct solutions.