1. For each statement below, check "True" if the statement is ***always*** true and "False" otherwise. Each correct answer is worth +1 point; each incorrect answer is worth −½ point; checking "I don't know" is worth +¼ point; and flipping a coin is (on average) worth +¼ point. You do ***not*** need to prove your answer is correct.

   Read each statement *very* carefully. Some of these are deliberately subtle.

☒ Yes ☐ No — If zero is odd, then $2+2=5$.
The hypothesis is false, so the implication is true.

☒ Yes ☐ No — For every language L, and for every string $w \in L$, there is a DFA that accepts w.
Consider the DFA that accepts *every* string. Accepting a *language* L also requires rejecting strings that are not in L.

☐ Yes ☒ No — Two languages L and L' are regular if and only if $L \cap L'$ is regular.
Let $L = \{0^n 1^n \mid n \geq 0\}$ and $L = \{1^n 0^n \mid n \geq 0\}$.

☒ Yes ☐ No — For every language L, the language L^* is non-empty.
L^* always contains the empty string ε.

☒ Yes ☐ No — Every regular language is recognized by an NFA with exactly 374 accepting states.
Thompson's algorithm construct an NFA with exactly one accepting state t. Add ε-transitions from t to 373 new accepting states.

☒ Yes ☐ No — If L does not have a fooling set of size 374, then L is regular.
Every subset of a fooling set is another fooling set. So if L has no fooling set of size 374, then it has no infinite fooling set.

☒ Yes ☐ No — The language $\left\{0^{374n} \;\middle|\; n \geq 374\right\}$ is regular.
This is $(0^{374})^{374}(0^{374})^*$

☐ Yes ☒ No — The language $\left\{0^{37n}1^{4n} \;\middle|\; n \geq 374\right\}$ is regular.
Intuitively, any algorithm to detect strings in this language would need to count the number of 0s. More directly, $(0^{37})^*$ is a fooling set for this language.

☒ Yes ☐ No — The language $\left\{0^{3n}1^{74n} \;\middle|\; n \leq 374\right\}$ is regular.
This language is finite.

☒ Yes ☐ No — The empty language is context-free.
Consider the grammar $S \to S$, with no other production rules.

Rubric: Max 10 points = +1 for each correct answer — ½ for each incorrect answer + ¼ for each IDK. Round negative scores up to 0.

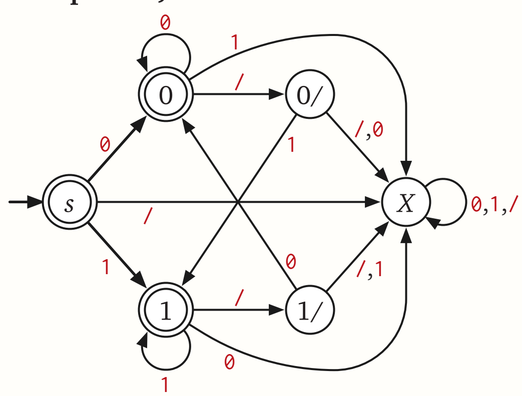
2. For any string $w \in \{\mathtt{0},\mathtt{1}\}^*$, let $slash(w)$ be the string in $\{\mathtt{0},\mathtt{1},\mathtt{/}\}^*$ obtained from w by inserting a new symbol $\mathtt{/}$ between any two consecutive symbols that are *not* equal. For example:

$$slash(\varepsilon) = \varepsilon$$
$$slash(\mathtt{00000}) = \mathtt{00000}$$
$$slash(\mathtt{000110111}) = \mathtt{000/11/0/111}$$

For any language $L \subseteq \{\mathtt{0},\mathtt{1}\}^*$, let $slash(L) = \{slash(w) \mid w \in L\}$.

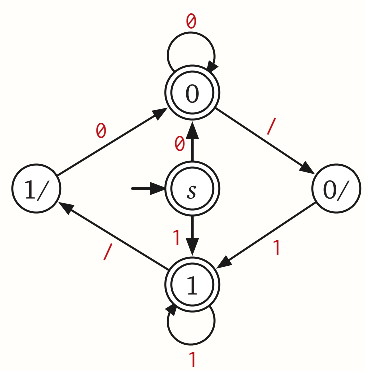
(a) Draw or describe a DFA that accepts the language $slash(\{\mathtt{0},\mathtt{1}\}^*)$.

Solution (explicit dump state):



■

Solution:



(All missing transitions go to a hidden dump state.) ■

Rubric: 2½ points, standard DFA rubric (scaled). These are not the only correct solutions. The state names are not necessary for credit.

(b) Give a regular expression for the language $slash(\{\mathtt{0},\mathtt{1}\}^*)$.

Solution: $\mathtt{1}^* + (\varepsilon + \mathtt{1}^+\mathtt{/})(\mathtt{0}^+\mathtt{/}\mathtt{1}^+\mathtt{/})^*\mathtt{0}^+(\varepsilon + \mathtt{/}\mathtt{1}^+)$ ■

Rubric: 2½ points, standard regular expression rubric (scaled). This is not the only correct solution.

(c) ***Prove*** that for any regular language L, the language *slash*(L) is also regular.

Solution (direct): Given a DFA $M = (Q, s, A, \delta)$ for L, we construct an NFA $N = (Q, s, A, \delta)$ for *slash*(L) as follows. The second component of each state indicates the last one or two symbols read.

$$Q' = Q \times \{0, 0/, 1, 1/\} \cup \{s'\}$$
$$s' \text{ is a new explicit state}$$
$$A' = Q \times \{0, 1\} \cup \{s'\}$$

$$\delta'(s', \varepsilon) = \{(s, 0/),\ (s, 1/)\}$$

$$\delta'((q,0),0) = \{(\delta(q,0),0)\} \qquad \delta'((q,1),1) = \{(\delta(q,1),1)\}$$
$$\delta'((q,0),/) = \{(q,0/)\} \qquad \delta'((q,1),/) = \{(q,1/)\}$$
$$\delta'((q,0/),1) = \{(\delta(q,1),1)\} \qquad \delta'((q,1/),0) = \{(\delta(q,0),0)\}$$

(All unspecified transitions go to the empty set $\varnothing$.) ■

Solution (almost-product): Let $M_1 = (Q_1, s_1, A_1, \delta_1)$ be an arbitrary DFA (over the alphabet $\{0, 1, /\}$) for the language *slash*$(\{0+1\}^*)$, and let $M_2 = (Q_2, s_2, A_2, \delta_2)$ be an arbitrary DFA (over the alphabet $\{0, 1\}$) for L. We construct a ***DFA*** $M = (Q, s, A, \delta)$ (over the alphabet $\{0, 1, /\}$) for *slash*(L) as follows:

$$Q = Q_1 \times Q_2$$
$$s = (s_1, s_2)$$
$$A = A_1 \times A_2$$
$$\delta((p,q),a) = \begin{cases} (\delta_1(p,a),\ q) & \text{if } a = / \\ (\delta_1(p,a),\ \delta_2(q,a)) & \text{otherwise} \end{cases}$$

■

Rubric: 5 points, standard language transformation rubric (scaled). These are not the only correct solutions.

3. Let L be the language $\{0^a 1^b 0^c \mid a + b = 2c\}$

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

Solution: Let $F = 0^*$.
Let x and y be arbitrary distinct strings in F.
Then $x = 0^i$ and $y = 0^j$ for some integers $i \neq j$.
Let $z = 1^i 0^i$.
Then $xz = 0^i 1^i 0^i \in L$ because $i + i = 2i$.
But $yz = 0^j 1^i 0^i \in L$ because $i + j \neq 2i$.
So F is an infinite fooling set for L. ■

Rubric: 5 points, standard fooling set rubric (scaled). This is not the only correct solution.

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

Solution:

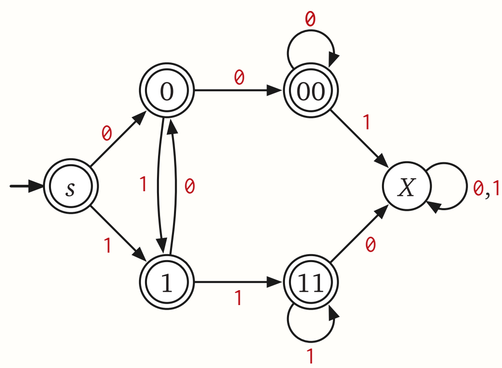
$$\begin{array}{ll} S \to 00S0 \mid 01B0 \mid B & \{0^a 1^b 0^c \mid a + b = 2c\} \\ B \to 11B0 \mid \varepsilon & \{1^b 0^c \mid b = 2c\} \end{array}$$ ■

Rubric: 5 points, standard CFG rubric (scaled). This is not the only correct solution. The language specifications for the non-terminals are not necessary for credit.

4. For each of the following languages L, give a regular expression that represents L ***and*** draw or describe a DFA that recognizes L.

(a) All strings in $\{0,1\}^*$ that do not contain either 001 or 110 as a substring

Solution: $(1+\varepsilon)(01)^*(0^*+1^*)$

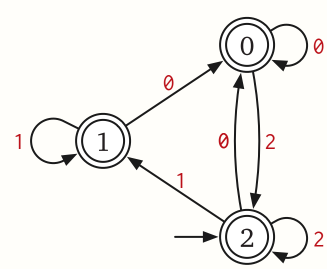


∎

Rubric: 5 points = 2½ for regular expression (standard rubric, scaled) + 2½ for DFA (standard rubric, scaled). These are not the only correct solutions.

(b) All strings in $\{0,1,2\}^*$ that do not contain either 01 or 12 as a substring

Solution: $(2^*1^*0^*02)^*2^*1^*0^*$



(All missing transitions go to a hidden dump state.) ∎

Rubric: 5 points = 2½ for regular expression (standard rubric, scaled) + 2½for DFA (standard rubric, scaled). These are not the only correct solutions.

5. For any string $w \in \{0, 1\}^*$, let $obliviate(w)$ denote the string obtained from w by removing every 1. Let L be an arbitrary regular language.

   (a) ***Prove*** that the language $\{obliviate(w) \mid w \in L\}$ is regular.

Solution: Let $M = (Q, s, A, \delta)$ be any DFA for L. We construct an NFA $N = (Q', s', A', \delta')$ for $obliviate(L)$ as follows:

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

$$\delta'(q, 0) = \{\delta(q, 0)\}$$
$$\delta'(q, 1) = \varnothing$$
$$\delta'(q, \varepsilon) = \{\delta(q, 1)\}$$

■

Rubric: 5 points: standard transformation rubric (scaled). This is not the only correct solution. No penalty for implicitly assuming the alphabet of M' is just $\{0\}$.

   (b) ***Prove*** that the language $\{w \in \{0, 1\}^* \mid obliviate(w) \in L\}$ is regular.

Solution: Let $M = (Q, s, A, \delta)$ be any DFA for L. We construct a DFA $M' = (Q', s', A', \delta')$ for $obliviate^{-1}(L)$ as follows:

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

$$\delta'(q, 0) = \delta(q, 0)$$
$$\delta'(q, 1) = q$$

■

Rubric: 5 points: standard transformation rubric (scaled). This is not the only correct solution.