CS/ECE 374 A ✧ Spring 2018
Midterm 1^ε Problem 1

For each statement below, check "Yes" if the statement is ***always*** true and "No" 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.

Yes ~~No~~ IDK — No infinite language is regular.

Solution: $\mathtt{0}^*$ is both infinite and regular. ■

~~Yes~~ No IDK — If L is regular, then for every string $w \in L$, there is a DFA that rejects w.

Solution: For every string w—in particular, for every string $w \in L$—there is a one-state DFA that rejects w. ■

Yes ~~No~~ IDK — If L is context-free and L has a finite fooling set, then L is not regular.

Solution: $\varnothing$ is a finite fooling set for *every* language, regular or not. $\mathtt{0}^*$ is both context-free and regular. ■

Yes ~~No~~ IDK — If L is regular and $L' \cap L = \varnothing$, then L' is not regular.

Solution: Consider $L' = \varnothing$, which is regular. ■

~~Yes~~ No IDK — The language $\{\mathtt{0}^i\mathtt{1}^j\mathtt{0}^k \mid i + j + k \geq 374\}$ is regular.

Solution: This language can be written as $\mathtt{0}^*\mathtt{1}^*\mathtt{0}^* \setminus L$, where L is a finite language (all binary strings of length less than 374). ■

Yes ~~No~~ IDK — The language $\{\mathtt{0}^i\mathtt{1}^j\mathtt{0}^k \mid i + j - k \geq 374\}$ is regular.

Solution: Consider the fooling set $F = \mathtt{01}^*$. ■

Yes ~~No~~ IDK — Let $M = (Q, \{\mathtt{0},\mathtt{1}\}, s, A, \delta)$ be an arbitrary DFA, and let $M' = (Q, \{\mathtt{0},\mathtt{1}\}, s, A, \delta')$ be the DFA obtained from M by changing every $\mathtt{0}$-transition into a $\mathtt{1}$-transition and vice versa. More formally, M and M' have the same states, input alphabet, starting state, and accepting states, but $\delta'(q,\mathtt{0}) = \delta(q,\mathtt{1})$ and $\delta'(q,\mathtt{1}) = \delta(q,\mathtt{0})$. Then $L(M) \cup L(M') = \{\mathtt{0},\mathtt{1}\}^*$.

Solution: If $L(M) = \varnothing$, then $L(M') = \varnothing$ as well. ■

~~Yes~~ No IDK

Let $M = (Q, \Sigma, s, A, \delta)$ be an arbitrary NFA, and $M' = (Q', \Sigma, s, A', \delta')$ be any NFA obtained from M by deleting some subset of the states. More formally, we have $Q' \subseteq Q$, $A' = A \cap Q'$, and $\delta'(q, a) = \delta(q, a) \cap Q'$ for all $q \in Q'$. Then $L(M') \subseteq L(M)$.

Solution: Every path from s to an accepting state in M' is also a path from s to an accepting state in M. Thus, every string accepted by M' is also accepted by M. ■

Yes ~~No~~ IDK

For every non-regular language L, the language $\left\{ 0^{|w|} \;\middle|\; w \in L \right\}$ is also non-regular.

Solution: If $L = \{0^{2^n} \mid n \geq 0\}$, then $L^* = 0^*$. ■

~~Yes~~ No IDK

For every context-free language L, the language $\left\{ 0^{|w|} \;\middle|\; w \in L \right\}$ is also context-free.

Solution: Replace each terminal symbol in the context-free grammar for L with a 0. ■

CS/ECE 374 A ✧ Spring 2018
Midterm 1^ε Problem 2

Prove that if L is a regular language, then $\textsc{StripFinal0s}(L)$ is also a regular language.

Solution: Let $M = (Q, s, A, \delta)$ be a DFA that accepts L. We construct an NFA $M' = (Q', s', A', \delta')$ with ε-transitions that accepts $\textsc{StripFinal0s}(L)$ as follows:

$$\begin{aligned} Q' &= Q \cup \{pass, strip\} \\ s' &= (s, pass) \\ A' &= A \times \{pass\} \end{aligned}$$

$$\begin{aligned} \delta'((q, pass),\, \varepsilon) &= \big\{(q, strip)\big\} \\ \delta'((q, pass),\, a) &= \big\{(\delta(q, a),\, pass)\big\} \\ \delta'((q, strip),\, \varepsilon) &= \big\{(\delta(q, 0), strip)\big\} \\ \delta'((q, strip),\, a) &= \varnothing \end{aligned}$$

Less formally: Add a shadow copy of M with 0-transitions replaced by ε-transitions and 1-transitions deleted, which guesses the deleted suffix of 0s. ■

Solution: Let $M = (Q, s, A, \delta)$ be a DFA that accepts L. We construct a new **DFA** $M' = (Q, s, A', \delta)$ with that accepts $\textsc{StripFinal0s}(L)$ by modifying the accepting states as follows:

$$A' = \big\{ q \in Q \;\big|\; \delta^*(q, 0^n) \cap A \neq \varnothing \text{ for some } n \geq 0 \big\}$$

Less formally: State q is an accepting state in M' if and only if there is a path of 0-transitions from q to an accepting state of M. ■

Rubric: 10 points =

- \+ 4 for a formal, complete, and unambiguous description of the output automaton.
  - − 2 for omitting $\varnothing$ transitions without declaring that omission.
- \+ 6 for correctness =
  - \+ 1½ for accepting ε when L contains at least one string in 0^*
  - \+ 1½ for accepting every non-empty string in $\textsc{StripFinal0s}(L)$
  - \+ 1½ for rejecting ε when $L \cap 0^* = \varnothing$
  - \+ 1½ for rejecting every non-empty string not in $\textsc{StripFinal0s}(L)$
  - − 1 for a single typo or similar mistake
- English explanation is not required, but may help us give you partial credit.

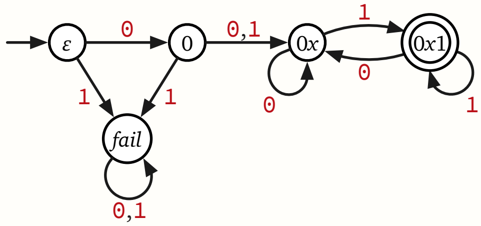
CS/ECE 374 A ✧ Spring 2018
Midterm 1^ε Problem 3

For each of the following languages L over the alphabet $\Sigma = \{0, 1\}$, give a regular expression that represents L ***and*** describe a DFA that recognizes L.

(a) $\{0^n w 1^n \mid n \geq 1 \text{ and } w \in \Sigma^+\}$

Solution: $0(0+1)(0+1)^*1$ ■

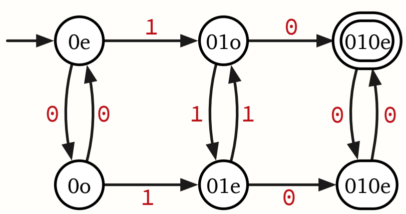
Solution: $0(0+1)^+1$ ■

Solution: 

■

(b) All strings in $0^*1^*0^*$ whose length is even.

Solution: $(00)^*(11)^*(00)^* + (00)^*(0+1)(11)^*(0+1)(00)^*$ ■

Solution: 

All missing transitions in the figure go to a hidden fail state. Each non-fail state is labeled with a prefix of 010, indicating which symbols have been read so far, and a letter e or o, indicating whether the number of symbols read is even or odd. ■

Rubric: 10 points = 2½ for each regular expression + 2½ for each DFA. Grade as four separate subproblems, each with IDK partial credit available.

— ½ for one typo
— 1 for incorrectly including/excluding a finite number of strings
— 2½ for incorrectly including/excluding an infinite number of strings
— 1 for omitting fail state from DFA without declaring that omission explicitly.

Explanations are not required for full credit. No penalty for leaving DFA states unlabeled. These are not the only correct answers.

CS/ECE 374 A ✧ Spring 2018
Midterm 1^ε Problem 4

(a) Give a *self-contained*, formal, recursive definition of the *parity* function. (In particular, do ***not*** refer to # or other functions defined in class.)

Solution: $parity(w) = \begin{cases} 0 & \text{if } w = \varepsilon \\ a \oplus parity(x) & \text{if } w = ax \end{cases}$

Here $\oplus$ is exclusive-or: $0 \oplus 0 = 1 \oplus 1 = 0$ and $0 \oplus 1 = 1 \oplus 0 = 1$. ■

Solution: $parity(w) = \begin{cases} 0 & \text{if } w = \varepsilon \\ 0 & \text{if } w = parity(x) \cdot x \text{ for some string } x \\ 1 & \text{otherwise} \end{cases}$ ■

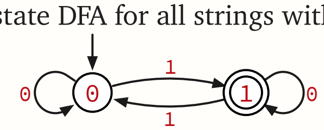
Rubric: 2 points = ½ for base case + 1½ for recursive case. These are not the only correct solutions. No penalty for using arithmetic or logical functions (such as $\neg a$ or $\sim a$ or $\overline{a}$ or $!a$ or $1 - a$ or $(a + 1) \bmod 2$ for logical negation). No credit for solutions that directly refer to "the number of 1s".

(b) Let L be an arbitrary regular language. Prove that $OddParity(L) := \{w \in L \mid parity(w) = 1\}$ is also regular.

Solution: $OddParity(L) = L \cap (0^*10^*1)^*0^*10^*$, where the latter regular expression describes *all* binary strings with odd parity. ■

Solution: $OddParity(L) = L \cap 0^*1(0 + 10^*1)^*$, where the latter regular expression describes *all* binary strings with odd parity. ■

Solution: We can construct a DFA for $OddParity(L)$ as the product of an arbitrary DFA for L and the following two-state DFA for all strings with odd parity:



A state (p, q) in the product DFA is accepting if and only if both p and q are accepting states of their respective DFAs. ■

Solution: Given a DFA $M = (Q, s, A, \delta)$ for L, we construct a new DFA $M' = (Q', s', A', \delta')$ for $EvenParity(L)$ as follows:

$$\begin{aligned} Q' &= Q \times \{0, 1\} \\ s' &= (s, 0) \\ A' &= \{(q, 1) \mid q \in A\} \\ \delta'((q, p), a) &= (\delta(q, a),\ p \oplus a) \end{aligned}$$

In each state (q,p) of M', q is the current state of M and p is the parity of the input read so far. ■

Rubric: 4 points:

- For closure argument: −1 for single mistake in regular expression for odd-parity strings
- For product construction: −1 for not specifying the accepting states of the final DFA.
- For explicit DFA construction: ½ for start state + 1½ for accepting states + 2 for transition function

(c) Let L be an arbitrary regular language. Prove that $AddParity(L) := \{parity(w) \cdot w \mid w \in L\}$ is also regular.

Solution: $AddParity(L) = ((\mathtt{0}+\mathtt{1}) \bullet L) \setminus OddParity((\mathtt{0}+\mathtt{1}) \bullet L)$.

The string $parity(w) \cdot w$ always has even parity. ■

Solution: $AddParity(L) = ((\mathtt{0}+\mathtt{1}) \bullet L) \cap (\mathtt{0}+\mathtt{1}\mathtt{0}^*\mathtt{1})^*$. ■

Solution: $AddParity(L) = \mathtt{1} \bullet OddParity(L) + \mathtt{0} \bullet \big(L \setminus OddParity(L)\big) \bullet \mathtt{1}$. ■

Solution: $AddParity(L) = \mathtt{1} \bullet OddParity(L) + \mathtt{0} \bullet EvenParity(L)$, where $EvenParity(L) = L \setminus OddParity(L)$ ■

Solution: $AddParity(L) = \mathtt{1} \bullet OddParity(L) + \mathtt{0} \bullet EvenParity(L)$, where $EvenParity(L) = L \cap (\mathtt{0}^*\mathtt{1}\mathtt{0}^*\mathtt{1})^*\mathtt{0}^*$ ■

Solution: $AddParity(L) = \mathtt{1} \bullet OddParity(L) + \mathtt{0} \bullet EvenParity(L)$, where $EvenParity(L)$ is accepted by the same product DFA as $OddParity(L)$, but with accepting states $A \times \{\mathtt{0}\}$ instead of $A \times \{\mathtt{1}\}$. ■

Solution: Let $M = (Q, s, A, \delta)$ be a DFA that accepts L. We define a new DFA $M' = (Q', s', A', \delta')$ that accepts $AddParity(L)$ as follows:

$$\begin{aligned}
Q' &= (Q \times \{\mathtt{0},\mathtt{1}\}) \cup \{s'\} \\
s' &\text{ is an explicit state in } Q' \\
A' &= \{(q,\mathtt{0}) \mid q \in A\} \\
\delta'(s',a) &= (s,a) \\
\delta'((q,p),a) &= (\delta(q,a), p \oplus a)
\end{aligned}$$

Each state (q,p) in M' indicates that M is in state q, and the parity of the input read so far is p. The extra start state reads the initial bit and initializes the parity accordingly; the rest of M' reads the rest of the input string and simulates M; fianlly, M' accepts if and only if the parity of the *entire* input string is even. ■

Rubric: Max 4 points. These are not the only correct solutions. A correct solution for (c) that assumes (b) is worth full credit, even if the solution to (b) is incorrect.

CS/ECE 374 A ✧ Spring 2018
Midterm 1^ε Problem 5

Let L be the language $\left\{\mathtt{0}^i\mathtt{1}^j\mathtt{0}^k \mid 2i = k \text{ or } i = 2k\right\}$.

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

Solution: Consider the set $F = \mathtt{0}^*$.

Let x and y be arbitrary distinct strings in F.

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

Let $z = \mathtt{0}^i\mathtt{1}\mathtt{0}^i$.

Then $xz = \mathtt{0}^{2i}\mathtt{1}\mathtt{0}^i \in L$, but $yz = \mathtt{0}^{j+i}\mathtt{1}\mathtt{0}^i \notin L$, because $j + i \neq 2i$ and $2(j + i) \neq i$.

Thus, F is a fooling set for L.

Because F is infinite, L cannot be regular. ■

Solution: Consider the set $F = \mathtt{0}^*$.

Let x and y be arbitrary distinct strings in F.

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

Without loss of generality, assume $i < j$.

Let $z = \mathtt{1}\mathtt{0}^{2j}$.

Then $xz = \mathtt{0}^i\mathtt{1}\mathtt{0}^{2j} \notin L$, because $2i < 2j$ and $i < j < 4j$.

But $yz = \mathtt{0}^j\mathtt{1}\mathtt{0}^{2j} \in L$.

Thus, F is a fooling set for L.

Because F is infinite, L cannot be regular. ■

Solution: Consider the set $F = (\mathtt{00})^*$.

Let x and y be arbitrary distinct strings in F.

Then $x = \mathtt{0}^{2i}$ and $y = \mathtt{0}^{2j}$ for some positive integers $i \neq j$.

Without loss of generality, assume $i < j$.

Let $z = \mathtt{1}\mathtt{0}^i$.

Then $xz = \mathtt{0}^{2i}\mathtt{1}\mathtt{0}^i \in L$.

But $yz = \mathtt{0}^{2j}\mathtt{1}\mathtt{0}^i \notin L$ because $2j > 2i$ and $4j > i$.

Thus, F is a fooling set for L.

Because F is infinite, L cannot be regular. ■

Rubric: 5 points: standard fooling set rubric (see HW2). These are not the only correct answers.

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

Solution:

$$
\begin{aligned}
S &\to A \mid B \\
A &\to \mathtt{0}A\mathtt{00} \mid C & [2i = k] \\
B &\to \mathtt{00}B\mathtt{0} \mid C & [i = 2k] \\
C &\to \mathtt{1}C \mid \varepsilon & [\mathtt{1}^*]
\end{aligned}
$$

■

Rubric: 5 points:

- − 1 for single typo or similar mistake
- − 1 for incorrectly omitting ε
- − 2 for incorrectly omitting/including a finite number of non-empty strings
- − 5 for incorrectly omitting/including a infinite number of strings

This is not the only correct answer. Explanations of nonterminals are not required, but they may help us give you partial credit.