

CS/ECE 374 A ♦ Spring 2018
Midterm 1^δ 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 $-\frac{1}{2}$ point; checking “I don’t know” is worth $+\frac{1}{4}$ point; and flipping a coin is (on average) worth $+\frac{1}{4}$ point.

Yes No IDK

Every infinite language is regular.

Solution: $\{0^n 1^n \mid n \geq 0\}$ is infinite but not regular. ■

Yes No IDK

If L is not regular, then for every string $w \in L$, there is a DFA that accepts w .

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

Yes No IDK

If L is context-free and L has a finite fooling set, then L is regular.

Solution: \emptyset is a finite fooling set for *every* language, regular or not. $\{0^n 1^n \mid n \geq 0\}$ is context-free but not regular. ■

The next question was asked inconsistently in the question handout and the answer booklet.
All students will receive full credit for this question.

Yes No IDK

Question handout: If L is regular and $L' \cap L = \emptyset$, then L' is regular.

Solution: $L = 000000(000)^*$ is regular, $L' = \{0^p \mid p \text{ is prime}\}$ is not regular, and $L' \cap L = \emptyset$. ■

Yes No IDK

Answer booklet: If L is regular and $L' \cap L = \emptyset$, then L' is **not** regular.

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

Yes No IDK

The language $\{0^i 1^j 0^k \mid i + j + k \geq 374\}$ is not regular.

Solution: This language is finite and therefore regular. ■

Yes No IDK

The language $\{0^i 1^j 0^k \mid i + j - k \geq 374\}$ is not regular.

Solution: Consider the fooling set $F = 01^*$. ■

Yes	No	IDK
-----	---------------	-----

Let $M = (Q, \{0, 1\}, s, A, \delta)$ be an arbitrary DFA, and let $M' = (Q, \{0, 1\}, s, A, \delta')$ be the DFA obtained from M by changing every 0-transition into a 1-transition and vice versa. More formally, M and M' the same states, input alphabet, starting state, and accepting states, but $\delta'(q, 0) = \delta(q, 1)$ and $\delta'(q, 1) = \delta(q, 0)$. Then $L(M) \cap L(M') = \emptyset$.

Solution: If $L(M) = (0 + 1)^*$, then $L(M') = (0 + 1)^*$ as well. ■

Yes	No	IDK
----------------	----	-----

Let $M = (Q, \Sigma, s, A, \delta)$ be an arbitrary NFA, and $M' = (Q', \Sigma, s, A', \delta')$ be a 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 regular language L , the language $\{0^{|w|} \mid w \in L\}$ is also regular.

Solution: Replace each symbol in the regular expression for L with a 0. ■

Solution: Given a DFA $M = (Q, \Sigma, s, A, \delta)$ for L , build an NFA $M' = (Q, \{0\}, s, A, \delta')$ for L' by defining $\delta'(q, 0) = \{\delta(q, a) \mid a \in \Sigma\}$. ■

Yes	No	IDK
----------------	----	-----

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

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

Rubric: max 10 points = +1 for each correct answer $-1/2$ for each incorrect answer $+1/4$ for each IDK; round negative scores up to zero. Explanations are not required (and will be ignored if provided).

CS/ECE 374 A ♦ Spring 2018
Midterm 1^δ Problem 2

Prove that if L is a regular language, then $\text{STRIPINIT}\Theta\text{s}(L)$ 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 $\text{STRIPINIT}\Theta\text{s}(L)$ as follows:

$$\begin{aligned} Q' &= Q \cup \{\text{strip}, \text{pass}\} \\ s' &= (s, \text{strip}) \\ A' &= A \times \{\text{pass}\} \\ \delta'((q, \text{strip}), \varepsilon) &= \{(\delta(q, \Theta), \text{strip}), (q, \text{pass})\} \\ \delta'((q, \text{strip}), a) &= \emptyset \\ \delta'((q, \text{pass}), \varepsilon) &= \emptyset \\ \delta'((q, \text{pass}), a) &= \{(\delta(q, a), \text{pass})\} \end{aligned}$$

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

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 $\text{STRIPINIT}\Theta\text{s}(L)$ as follows:

$$\begin{aligned} Q' &= Q \cup \{s'\} \\ s' &\text{ is an explicit state in } Q' \\ A' &= A \\ \delta'(s', \varepsilon) &= \bigcup_{n \geq 0} \delta^*(s, \Theta^n) \\ \delta'(s', a) &= \emptyset \\ \delta'(q, \varepsilon) &= \emptyset \\ \delta'(q, a) &= \{\delta(q, a)\} \end{aligned}$$

Less formally: Add a new start state s' with ε -transitions to every state reachable from s by Θ -transitions alone. ■

Rubric: 10 points =

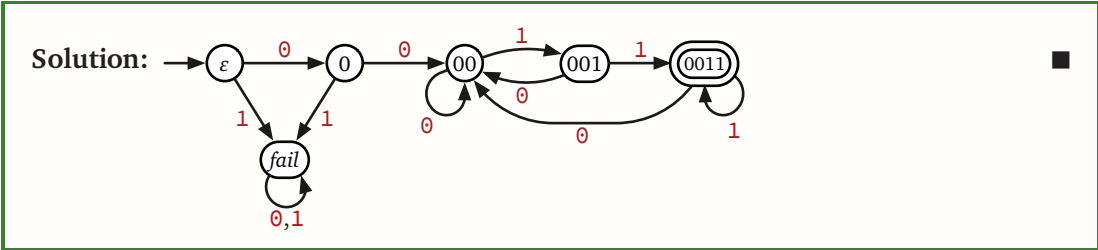
- + 4 for a formal, complete, and unambiguous description of the output automaton.
 - 2 for omitting \emptyset transitions without declaring that omission.
- + 6 for correctness =
 - + 1½ for accepting ε when L contains at least one string in Θ^*
 - + 1½ for accepting every non-empty string in $\text{STRIPINIT}\Theta\text{s}(L)$
 - + 1½ for rejecting ε when $L \cap \Theta^* = \emptyset$
 - + 1½ for rejecting every non-empty string not in $\text{STRIPINIT}\Theta\text{s}(L)$
 - 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^δ 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) $\{0^n w 1^n \mid n > 1 \text{ and } w \in \Sigma^*\}$

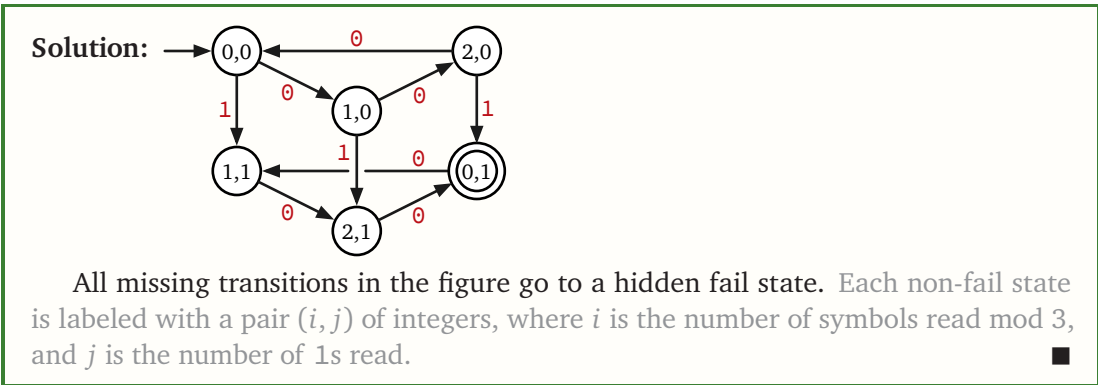
Solution: $00(0+1)^*11$ ■



(b) All strings in 0^*10^* whose length is a multiple of 3.

Solution: $(000)^*(001 + 010 + 100)(000)^*$ ■

Solution: $(000)^*1(000)^*00 + 0(000)^*1(000)^*0 + 00(000)^*1(000)^*$ ■



Rubric: 10 points = $2\frac{1}{2}$ for each regular expression + $2\frac{1}{2}$ for each DFA. Grade as four separate subproblems, each with IDK partial credit available.

- $\frac{1}{2}$ for one typo
- 1 for incorrectly including/excluding a finite number of strings
- $2\frac{1}{2}$ 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^δ 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:
$$\text{parity}(w) = \begin{cases} 0 & \text{if } w = \varepsilon \\ a \oplus \text{parity}(x) & \text{if } w = ax \end{cases}$$

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

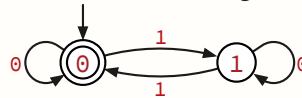
Solution:
$$\text{parity}(w) = \begin{cases} 0 & \text{if } w = \varepsilon \\ 0 & \text{if } w = \text{parity}(x) \cdot x \text{ for some string } x \\ 1 & \text{otherwise} \end{cases}$$
 ■

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 \bar{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 $\text{EvenParity}(L) := \{w \in L \mid \text{parity}(w) = 0\}$ is also regular.

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

Solution: We can construct a DFA for $\text{EvenParity}(L)$ as the product of an arbitrary DFA for L and the following two-state DFA for all strings with even 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 $\text{EvenParity}(L)$ as follows:

$$Q' = Q \times \{0, 1\}$$

$$s' = (s, 0)$$

$$A' = A \times \{0\}$$

$$\delta'((q, p), a) = (\delta(q, a), p \oplus a)$$

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 “even” regular expression
- For product construction: -1 for not specifying the accepting states of the final DFA.
- For explicit DFA construction: 1 for complete explicit formal DFA description + 1½ for correct acceptances + 1½ for correct rejections

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

Solution: $AddParity(L) = EvenParity(L) \cdot (0 + 1)$.

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

Solution: $AddParity(L) = (L \cdot (0 + 1)) \cap (0^*10^*1)^*0^*$. ■

Solution: $AddParity(L) = EvenParity(L) \cdot 0 + (L \setminus EvenParity(L)) \cdot 1$. ■

Solution: $AddParity(L) = EvenParity(L) \cdot 0 + OddParity(L) \cdot 1$, where $OddParity(L) = L \setminus EvenParity(L)$ ■

Solution: $AddParity(L) = EvenParity(L) \cdot 0 + OddParity(L) \cdot 1$, where $OddParity(L) = L \cap (0^*10^*1)^*0^*10^*$ ■

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

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^δ Problem 5

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

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

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

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 $z = 1^i 0^{i+1}$.

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

Thus, F is a fooling set for L .

Because F is infinite, L cannot be regular. ■

Solution: Consider the set $F = 11^*$.

Let x and y be arbitrary distinct strings in F .

Then $x = 1^i$ and $y = 1^j$ for some positive integers $i \neq j$.

Let $z = 1^i$.

Then $xz = 1^i 0^i = 0^0 1^i 0^i \in L$, but $yz = 1^j 0^i = 0^0 1^j 0^i \notin L$ because $0 \neq j$ and $j \neq i$.

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:

$$S \rightarrow AB \mid BC$$

$$A \rightarrow 0A1 \mid \varepsilon$$

$$[0^n 1^n]$$

$$B \rightarrow 0B \mid \varepsilon$$

$$[0^*]$$

$$C \rightarrow 1C0 \mid \varepsilon$$

$$[1^n 0^n]$$

■

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.