CS/ECE 374 A ✧ Fall 2021 Fake Midterm 1 Problem 1	Name: Dakshita

For each statement below, check "Yes" if the statement is ***ALWAYS*** true and "No" otherwise, and give a ***brief*** explanation of your answer.

(a) Every integer in the empty set is prime.

☑ Yes ☐ No

vacuous / empty set contains no integers

(b) The language $\{0^m 1^n \mid m + n \leq \cancel{374}\,5\}$ is regular.

☑ Yes ☐ No

$0^0 1^{374} + 0^1 1^{373} + \dots\dots 0^{373} 1 + 0^{374} 1^0$

(c) The language $\{0^m 1^n \mid m - n \leq 374\}$ is regular.

☐ Yes ☑ No

FOOLING Set : 0^*

(d) For all languages L, the language L^* is regular.

☐ Yes ☑ No

$\{0^n 1^n \mid n \geq 0\}$

(e) For all languages L, the language L^* is infinite.

☐ Yes ☑ No

NOT TRUE when $L = \{\varepsilon\} \Rightarrow L^* = \{\varepsilon\}$. (Another: $L = \emptyset$)

(f) For all languages $L \subset \Sigma^*$, if L can be represented by a regular expression, then $\Sigma^* \setminus L$ is recognized by a DFA.

if L is regular

is regular

☑ Yes ☐ No

Flip accepting and rejecting states in DFA for L.

(g) For all languages L and L', if $L \cap L' = \emptyset$ and L' is not regular, then L is regular.

☐ Yes ☑ No

$0^n 1^n$ is not regular, $1^n 0^n$ is not regular $n \geq 1$.

(h) Every regular language is recognized by a DFA with exactly one accepting state.

☐ Yes ☑ No

No ε- transitions

(i) Every regular language is recognized by an NFA with exactly one accepting state.

☑ Yes ☐ No

ε- transition from all acc. states to a single acc. state.

(j) Every language is either regular or context-free.

☐ Yes ☑ No

$0^n 1^n 2^n$

CS/ECE 374 A ✧ Fall 2021 Fake Midterm 1 Problem 2	Name:

For each of the following languages over the alphabet $\Sigma = \{0, 1\}$, either ***prove*** that the language is regular or ***prove*** that the language is not regular. ***Exactly one of these two languages is regular.*** Both of these languages contain the string 00110100000110100.

1. $\{0^n w 0^n \mid w \in \Sigma^+ \text{ and } n > 0\}$

$0\underbrace{1}_{=w}0$, $0\underbrace{0}_{=w}0$, $0\boxed{\ \ }0$,

$00\boxed{\ \ }00$, $000\_\_\_\_000$

$$0(0+1)^+0$$

Let $z \in \{0^n w 0^n \mid w \in \Sigma^+, n > 0\}$.

Then $z = \underbrace{0\ldots0}_{n} w \underbrace{0\ldots0}_{n} = 0\,\underbrace{000..w\ \ .000}\,0 \in 0(0+1)^+0$

Let $z \in 0(0+1)^+0$.

Then $z = 0^1 w 0^1$ where $w \in \Sigma^+ \in \{0^n w 0^n \mid w \in \Sigma^+, n > 0\}$

2. $\{w0^n w \mid w \in \Sigma^+ \text{ and } n > 0\}$

$0 0^n 0$, $1 0 1$, $1^2 0 1^2$, $1^3 0 1^3$, $1^4 0 1^4$ . .

$F = \{1^n 0 : n > 0\}$.

Let x, y be ANY distinct strings in F.

$x = 1^i 0$, $y = 1^j 0$, $i \neq j$, $i, j > 0$.

$z = 1^i$

$xz = 1^i 0 1^i \in L$, $yz = \underbrace{1^j}_{w_1} \underbrace{0}_{} \underbrace{1^i}_{w_2}$ $n=1$ $w_1 \neq w_2$ $\therefore yz \notin L$.

F is infinite fooling set. So, L is not regular.

CS/ECE 374 A ✧ Fall 2021 Fake Midterm 1 Problem 3	Name:

The *parity* of a bit-string w is $\texttt{0}$ if w has an even number of $\texttt{1}$s, and $\texttt{1}$ if w has an odd number of $\texttt{1}$s. For example:

$$parity(\varepsilon) = \texttt{0} \qquad parity(\texttt{0010100}) = \texttt{0} \qquad parity(\texttt{00101110100}) = \texttt{1}$$

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

$$parity(w) = \begin{cases} 0 & \text{if } w = \varepsilon \\ parity(x) & \text{if } w = 0 \cdot x \\ 1 \oplus parity(x) & \text{if } w = 1 \cdot x \end{cases}$$

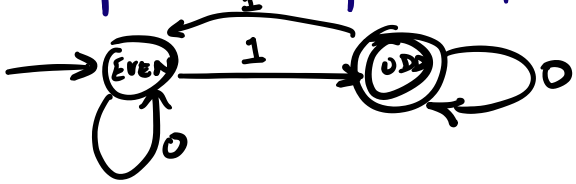
(b) Let L be an arbitrary regular language. Prove that the language $OddParity(L) := \{w \in L \mid parity(w) = \texttt{1}\}$ is also regular.

strings in L that have odd number of 1s.

Product construction of M and M',

$M = (\Sigma, s, Q, A, \delta)$ is the DFA for L, (exists because L is regular)

$M' = (\Sigma', s', Q', A', \delta')$ is the DFA that on input w, computes parity(w).



Accepting states (in product) $= \{(a, a')$ s.t. $a \in A, a' \in A'\}$

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

OddParity(L) is regular.

Similarly EvenParity(L) is regular. [Change M' to flip acc, reject states]

$= \{w \in L \mid parity(w) = 0\}$

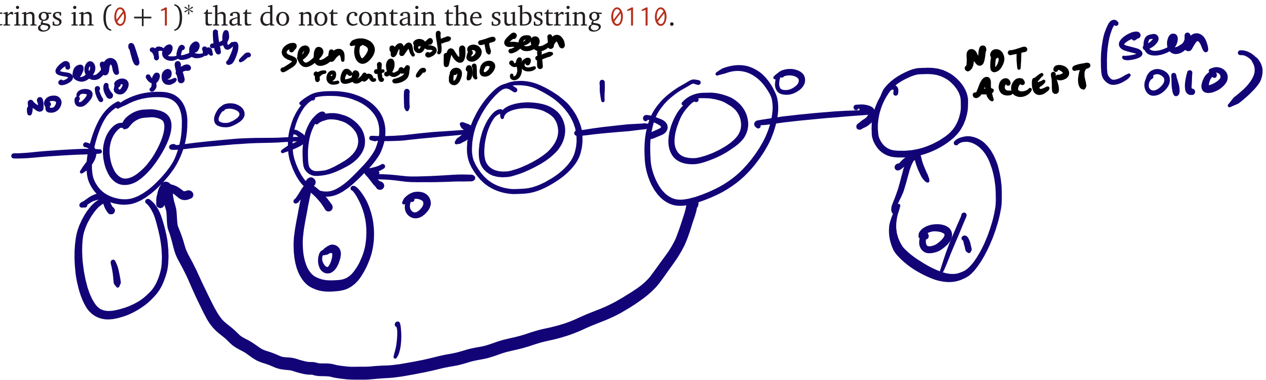
$$AddParity(L) = 0 \cdot Evenparity(L) + 1 \cdot OddParity(L)$$

[Hint: Yes, you have enough room.]

CS/ECE 374 A ✧ Fall 2021 **Fake Midterm 1 Problem 4**	Name:

For each of the following languages L, give a regular expression that represents L ***and*** describe a DFA that recognizes L. You do ***not*** need to prove that your answers are correct.

(a) All strings in $(0+1)^*$ that do not contain the substring 0110.

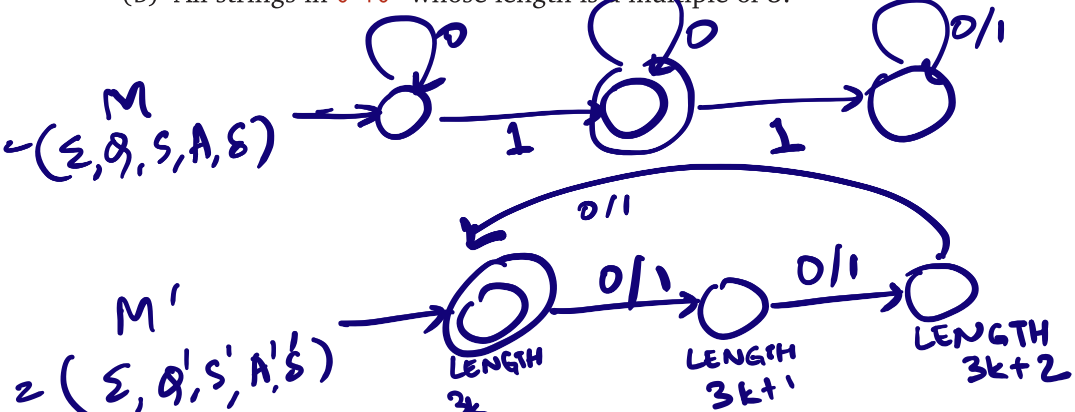


0001

$$1^*\Big(0\underbrace{(\varepsilon+1+1111^*)}_{}\Big)^*1^*$$

any run of 1s that follows a 0 has length 0 or 1 or ≥ 3.

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



0^*10^*

length 3k.

Product construction with Acc states. $A \times A'$.

$(000)^*010(000)^*$

$+(000)^*100(000)^*$

$+(000)^*001(000)^*$

CS/ECE 374 A ✧ Fall 2021 Fake Midterm 1 Problem 5	Name:

For any string $w \in \{0, 1\}^*$, let *obliviate*(w) denote the string obtained from w by removing every 1. For example:

$$obliviate(\varepsilon) = \varepsilon$$
$$obliviate(000000) = 000000$$
$$obliviate(111111) = \varepsilon$$
$$obliviate(010001101) = 00000$$

Let L be an arbitrary regular language.

1. ***Prove*** that the language $\textsc{Obliviate}(L) = \{obliviate(w) \mid w \in L\}$ is regular.

notes. [$x \in OBL(L)$ iff $x = obliviate(w)$, $w \in L$.
$w = 010011$
$x = 000$ $w = 010011 \in L$.

Let M be DFA for L, $M = (\Sigma, Q, s, A, \delta)$

M' is NFA for $OBL(L)$, $M' = (\Sigma, Q', s', A', \delta')$

$Q' = Q$ $\delta'(q, 0) = \delta(q, 0)$ Changing $\xrightarrow{1}$ arrows in M to $\xrightarrow{\varepsilon}$ arrows in M'.

$s' = s$ $\delta'(q, 1) = \emptyset$

$A' = A$ $\delta'(q, \varepsilon) = \delta(q, 1)$

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

notes. [$w \in UNOBL(L)$ iff $OBliviate(w) \in L$.
M' gets $w \to$ first $OBLIVIATE(w)$, i.e. remove 1s $\longrightarrow$ run M on the remaing string of 0's.

Let M be a DFA for L, $M = (\Sigma, Q, s, A, \delta)$

M' is NFA for $OBLV(L)$, $M' = (\Sigma, Q', s', A', \delta')$

$Q' = Q$ $\delta'(q, 0) = \delta(q, 0)$ changing $O \xrightarrow{1} O$

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

$A' = A$ to $O \circlearrowleft 1$ O