1. For each of the following statements, select true if the statement is always true. Briefly justify your answer choices.

a. For any language **L**, the language **L*** is regular. (True/False)

$(NR)^*$ ww^R ww^R ww^R

is regular

b. For all languages **L** ∈ **Σ***, if L is ~~recognized by a DFA~~, then **Σ*\L** can be ~~represented by a regular expression~~. (True/False)

regular

c. Let **L** and **L'** be arbitrary languages. It is known that **L** ∩ **L' = Ø**. If it is knows that **L'** is not regular, **L** is regular. (True/False)

$\boxed{?} \cap \boxed{NR} = \boxed{\varnothing}$ $\quad (0^n 1^n \cap 1^n 0^n)$

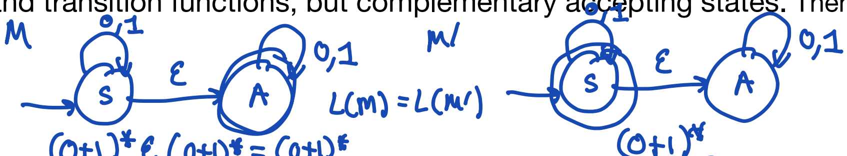
$L \cap L' \quad \varnothing \quad n > 0$

d. For all languages **L,** if **L** is not regular, **L** does not have a finite fooling set. (True/False)

$0^n 1^n$ take any ∞ fooling set for L, and pick 5 elems, we get a finite fooling set.

e. Let **M = (Σ, Q, s, A, δ),** and **M' = (Σ, Q, s, Q\A, δ)** be arbitrary DFAs with identical alphabets start states, and transition functions, but complementary accepting states. Then **L(M) ∩ L(M') = Ø.** (True/False)

f. Let **M = (Σ, Q, s, A, δ),** and **M' = (Σ, Q, s, Q\A, δ)** be arbitrary NFAs with identical alphabets start states, and transition functions, but complementary accepting states. Then **L(M) ∩ L(M') = Ø.** (True/False)



g. For all context free languages **L**, **L'**, **L** • **L'** is also context free. (True/False)

h. Every non-context free language is not regular. (True/False)

2. The following two languages are defined over the alphabet **Σ = {0,1}.** Decide if each language is regular or non-regular. Either way, prove it!

extrems $n=1$

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

Yes: $\underbrace{1}_{w}0\underbrace{1}_{w}$, $\underbrace{011}_{w}00000\underbrace{011}_{w}$

No: 11000111, 11 0 1111, 1 0 11111, 111 0 1111111

$\forall\, i \neq j$, $i>0$, $j>0$

$x = 1^i 0 1^i \in L$

$y = 1^j 0 1^i \notin L$

$1^i 0 0 1^i$

$1^j 0 0 1^i$

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

Yes: $0\underbrace{1}_{w}0$, $0\underbrace{110}_{w}0$, $\underbrace{0110110000}_{w}0$ pattern!!

No: 0101

$0\underbrace{110010010100}_{w}0$

$\cancel{0^+ \Sigma^+ 0^+}$

$\cancel{0^+ (0+1)^+ 0^+}$

$\Rightarrow \boxed{0(0+1)^+0}$

3. For all strings **w** ∈ **{0,1}***, ***tiny(w)*** is the string obtained from w by replacing each run of consecutive **0**'s with a single **0** each run of consecutive **1**'s with a single **1.**

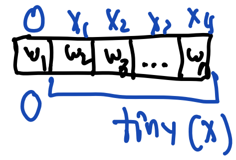
$tiny(0000111) = 01$

a. Define ***tiny(w)*** recursively. [Hint: How are strings defined?]

$w = \varepsilon$

$w = ax$ $\quad a \in \Sigma$, x is a string

$$tiny(w) = \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ 0 & w = 0 \\ 1 & w = 1 \\ 0 \cdot tiny(x) & w = 00x \\ 1 \cdot tiny(x) & w = \underline{11}x \\ 0 \cdot tiny(1x) & w = 01x \\ 1 \cdot tiny(0x) & w = \underline{10x} \end{cases}$$



B

$big(w) := \{w \in \{0,1\}^* \mid tiny(w) \in L\}$

b. For any regular language L, prove that big(w):={ w∈{0,1}* | tiny(w) ∈ L} is also regular. [Hint this is a language transformation task.]

Let M be a DFA $(Q, \Sigma, \delta, s, A)$ that accepts L.

Let B be an NFA $(Q', \Sigma', \delta', s', A')$ that accepts B.

0|00 1111
0 1

$Q': Q \times \{0,1\}^{\swarrow \text{last}} \cup \{s'\}$

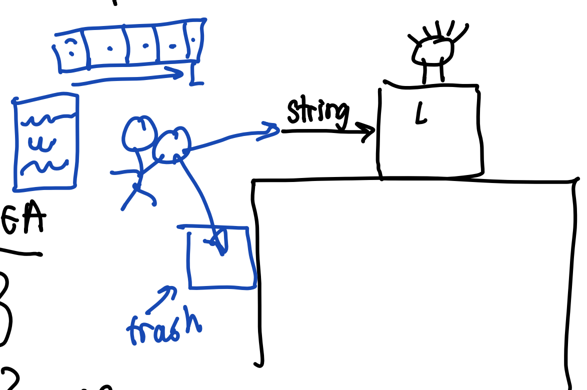
$s': s'$

$A': A \times \{0,1\} \cup \{s'\}$ if $s \in A$

$\delta': \delta(s', a) = \{(\delta(s,a), a)\}$

$$\delta((q,a), b) = \begin{cases} \{(q,a)\} & \text{if } b = a \\ \{(\delta(q,b), b)\} & \text{else} \end{cases}$$

$q \in Q$



$x \in \Sigma^*$

2 Language Transformation - 20 points

Let $\Sigma = \{0, 1\}$ and let L be an arbitrary regular language over Σ.

Define the operation TwoIsWild(L) as follows:

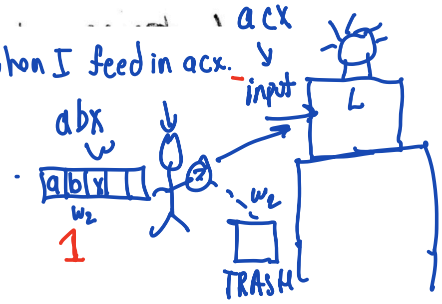
$$\text{TwoIsWild}(L) = \{abx \mid a \in \Sigma, b \in \Sigma, acx \in L \text{ for some symbol } c \in \Sigma\}.$$

To summarize in words, every string (of length 2 or longer) in L is also in TwoIsWild(L). Additionally, you can take any string w from L, change the 2nd character to anything you want, and the resulting string will be in TwoIsWild(L).

Show that TwoIsWild(L) is regular by constructing an NFA. You may assume a DFA for L exists as $(Q, \Sigma, \delta, s, A)$.

Intuition

- I have a blackbox on the table that lights up when I feed in acx.
- Feed 1^{st} char of w into box!
- I throw the second character that I read into the trash can and make a guess for the second char. I feed that guess into the blackbox.
- I feed the rest of X into L.
- If the light on the box is on, accept!



Let M be a dFA for L.

Build an NFA for TWOISWILD(w) as follows.

$M' = (Q, \Sigma, \delta', s', A')$

Q': $Q \times Q \cup \{s'\}$

s': s'

A': $\bigcup_{\substack{q_i \in A \\ q_j \in Q}} (q_i, q_j) = Q \times A$ → The box on the table lights up → its in an accepting state!

δ:

$\delta'(s', a) = \{(\delta(s,a), s)\}$, $a \in \Sigma$ → Record the state of the blackbox, then feed the first character a into my blackbox.

$\delta'((q_i, s), c) = \{(\delta(q_i, d), q_i) \mid d \in \Sigma\}$, $q_i \in Q$, $c \in \Sigma$ → If the previous state of my blackbox was S, I'm now reading the second charcter, throw it onto the floor, guess the second charcter & feed it into the box

$\delta'((q_i, q_j), c) = \{\delta(q_i, c), q_i\}$, $q_i \in Q$, $q_j \neq S$, $c \in \Sigma$ → After taking care of the 2^{nd} char, feed the rest of x into the blackbox and see if it lights up!