CSCE 355
2/17/2025

More closure properties:
Principal prefix
Homomorphisms
Pumping Lemma (?)

Recall: Every nonempty string $w = xa$ for unique $x \in \Sigma^*$ and $a \in \Sigma$. x is the principal prefix of w.

Def: Let $L \subseteq \Sigma^*$ be a language.

$$pp(L) = \{\text{principal prefixes of strings from } L\}$$

$$= \{x \in \Sigma^* : (\exists a \in \Sigma)\ xa \in L\}$$

Prop: If L is regular, then $pp(L)$ is regular.

Proof: Method 1: Let D be a DFA s.t. $L(D) = L$.

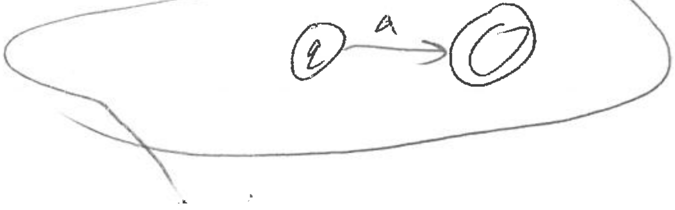
$D := \langle Q, \Sigma, \delta, q_0, F \rangle$.

Define a DFA $pp(D)$ such that $L(pp(D)) = pp(L)$.

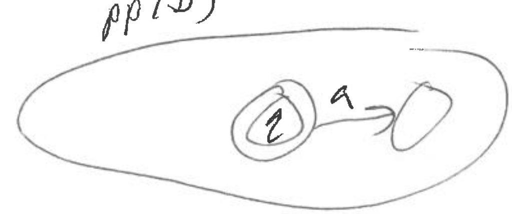
$pp(D) = \langle Q, \Sigma, \delta, q_0, F' \rangle$ where

$$F' := \{q \in Q : \exists a \in \Sigma,\ \delta(q,a) \in F\}$$

D:



pp(D)



Proof of correctness: $\forall x \in \Sigma^*$

$$x \in L(pp(D)) \iff \underbrace{\hat{\delta}(q_0, x)}_{q} \in F' \quad \text{(def of acceptance in } pp(D))$$

$$\underset{\text{def of } F'}{\iff} \exists a \in \Sigma,\ \delta(\underbrace{\hat{\delta}(q_0, x)}_{q}, a) \in F$$

$$\iff \exists a \in \Sigma,\ \hat{\delta}(q_0, xa) \in F$$

$$[\hat{\delta}(q_0, xa) = \delta(\hat{\delta}(q_0, x), a)]$$

$$\iff \exists a \in \Sigma,\ xa \in L(D) \quad [= L]$$

$$\underset{\text{def of } pp(L)}{\iff} x \in pp(L)$$

$\therefore$ $L(pp(D)) = pp(L)$

$\therefore$ $pp(L)$ is regular $\blacksquare$ Method 1

Method 2: Let r be a regex such that $L(r) = L$. Give rules for transforming r to a regex r' such that $L(r') = pp(L)$:

	r	r'	
	$\emptyset$	$\emptyset$	
$(a \in \Sigma)$	a	$\varepsilon \quad (= \emptyset^*)$	
	$s + t$	$s' + t'$	Ind hyp: s' & t' are already defined
s, t regexes	st	$\begin{cases} st' & \text{if } \varepsilon \notin L(t) \\ st' + s' & \text{otherwise} \end{cases}$	
	s^*	$s^* s'$	

Proof of correctness omitted.

$L \subseteq \Sigma^*$.

DROP-ONE(L) := set of strings obtained from (nonempty) strings in L by removing a single symbol somewhere in the string.

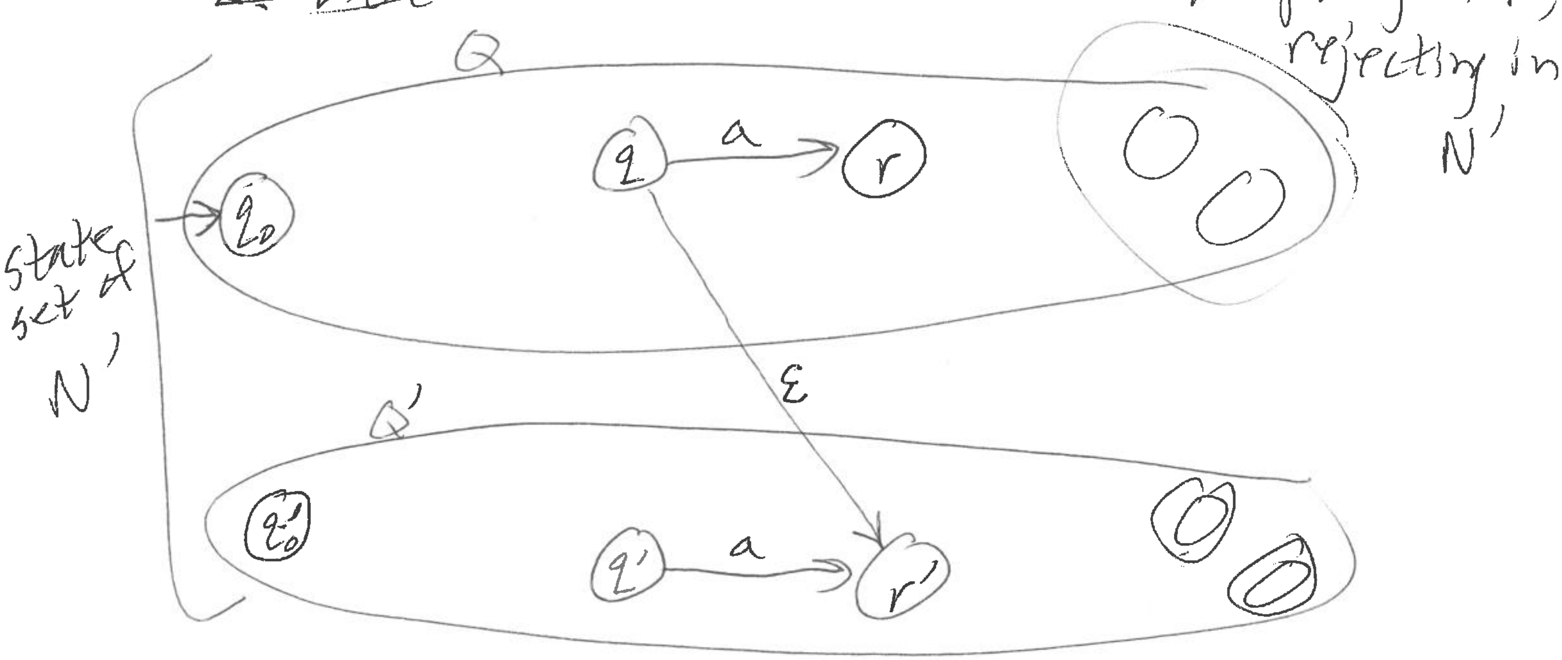
$= \{ wx : w, x \in \Sigma^* \text{ and } \exists a \in \Sigma, \; wax \in L \}$

<u>Prop</u>: If L is regular, then DROP-ONE(L) is regular.

Proofs: Method 1. Let N be an NFA such that $L = L(N)$. Define an ε-NFA N' such that $L(N') = \text{DROP-ONE}(L)$:

1) Make 2 disjoint copies Q, Q' ~~Ea~~ of the state set Q of N.

~~2. Let~~



2) start state of N' is the start state of N in Q (top ~~copy~~ copy)

3) accepting states are those in Q' (bottom copy)

4) Transitions within a copy are ~~both~~ the same as in N

5) For every transition $q \xmapsto{a} r$ in N, add an ε-transition $q \xmapsto{\varepsilon} r'$ //

<u>Method 2</u>: Let r be a regex s.t. $L = L(r)$. (5)

Rules to build r' regex s.t. $L(r') = \text{DROP-ONE}(L)$

	r	r'
	$\emptyset$	$\emptyset$
$(a \in \Sigma)$	a	$\varepsilon \; (= \emptyset^*)$
	$s + t$	$s' + t'$
	st	$\underbrace{s't}_{\text{drop from the } s} + \underbrace{st'}_{\text{drop from the } t}$
	s^*	$s^* s' s^*$

<u>String Homomorphisms</u>

<u>Def</u>: Let Σ, Γ be alphabets. A string homomorphism from Σ^* to Γ^* is a map

$$\varphi : \Sigma^* \longrightarrow \Gamma^*$$

such that $\forall w, x \in \Sigma^*$

$$\underbrace{\varphi(\underbrace{wx}_{\substack{\text{concat in} \\ \Sigma^*}})}_{} = \underbrace{\varphi(w)\varphi(x)}_{\substack{\text{concat} \\ \text{in } \Gamma^*}}$$

Basic facts: $\varphi : \Sigma^* \to \Gamma^*$ string homom.

1) $\varphi(\varepsilon) = \varepsilon$

2) φ is uniquely determined by input strings of length 1.

Proof:

1) $$\underbrace{\varphi(\varepsilon)}_{\text{length } n} = \varphi(\varepsilon\varepsilon) = \underbrace{\varphi(\varepsilon)\varphi(\varepsilon)}_{\text{length } 2n}$$

$n = 2n$ $\Rightarrow$ $n = 0$

$\therefore |\varphi(\varepsilon)| = 0 \quad \therefore \varphi(\varepsilon) = \varepsilon$

2) Write a string $w \in \Sigma^*$ as

$$w = w_1 w_2 \cdots w_n \qquad (\text{each } w_i \in \Sigma)$$

Then

$$\varphi(w) = \varphi(w_1 \cdots w_{n-1} w_n) = \varphi(w_1 \cdots w_{n-1})\varphi(w_n)$$

$$= \varphi(w_1 \cdots w_{n-2})\varphi(w_{n-1})\varphi(w_n) = \cdots$$

$$= \varphi(w_1)\varphi(w_2)\cdots\varphi(w_n)$$

Example: $\Sigma = \{a, b, c\}$, $\Gamma = \{0, 1\}$

$\varphi(a) = 011$
$\varphi(b) = 10$
$\varphi(c) = \varepsilon$

$$\varphi(acbcab) = 011\underset{\uparrow \varepsilon}{\,}10\underset{\uparrow \varepsilon}{\,}01110$$

Def: Let Σ, Γ be alphabets and $\varphi : \Sigma^* \to \Gamma^*$ a string homom.

1) For any lang. $L \subseteq \Sigma^*$, define

$$\varphi(L) = \{\varphi(w) : w \in L\} \subseteq \Gamma^*$$

("image of L under φ")

2) For any lang. $A \subseteq \Gamma^*$, define

$$\varphi^{-1}(A) := \{w \in \Sigma^* : \varphi(w) \in A\}$$

("inverse image of A under φ")

Theorem: Let $\varphi : \Sigma^* \to \Gamma^*$ be a string homom.

1) If $L \subseteq \Sigma^*$ is regular, then $\varphi(L)$ is regular

2) If $A \subseteq \Gamma^*$ is regular, then $\varphi^{-1}(A)$ is regular.

Proof:

1) Transform a regex r over Σ to a regex r' over Γ such that

$$\varphi(L(r)) = L(r')$$

	r	r'
	$\emptyset$	$\emptyset$
$(a \in \Sigma)$	a	$\varphi(a) \in \Gamma^*$ (concatenation of single-symbol regexes)
	$s+t$	$s'+t'$
	st	$s't'$
	s^*	$(s')^*$

Ex: $r = (ab + bc)^*$

$r' = (0110 + 10)^*$

// Proof of (1)

Proof of (2) next time