Section: Properties of Regular Languages

Example

$L = \{a^n b a^n \mid n > 0\}$ Not regular!

$= \{aba, aabaa, \ldots\}$

$abaa \in L$ No

Can you build a DFA? or regexpr?

Closure Properties

A set is closed over an operation if

$$\mathbf{L_1, L_2 \in class}$$
$$\mathbf{L_1 \ op \ L_2 = L_3}$$
$$\mathbf{\Rightarrow L_3 \in class}$$

L={x | x is a positive even integer}

L is closed under

addition? yes
multiplication? yes
subtraction? no $6-10=-4 \notin L$
division? no

Closure of Regular Languages

Theorem 4.1 If L_1 and L_2 are regular languages, then

$$\mathbf{L}_1 \cup \mathbf{L}_2$$
$$\mathbf{L}_1 \cap \mathbf{L}_2$$
$$\mathbf{L}_1\mathbf{L}_2$$
$$\bar{L}_1$$
$$\mathbf{L}_1^*$$

are regular languages.

Proof(sketch)

L_1 and L_2 are regular languages
$\Rightarrow \exists$ reg. expr. r_1 and r_2 s.t.
$L_1 = L(r_1)$ and $L_2 = L(r_2)$
$r_1 + r_2$ is r.e. denoting $L_1 \cup L_2$
$\Rightarrow$ closed under union
$r_1 r_2$ is r.e. denoting $L_1 L_2$
$\Rightarrow$ closed under concatenation
r_1^* is r.e. denoting L_1^*
$\Rightarrow$ closed under star-closure

complementation:

L_1 is reg. lang.

$\Rightarrow \exists$ DFA M s.t. $L_1 = L(M)$

Construct M' s.t.

final states in M
are nonfinal state in M'
nonfinal states in M
are final states in M'

show $w \in L(M') \Leftrightarrow w \in \overline{L}$

$\Rightarrow$ closed under complementation

intersection:

$\mathbf{L}_1$ **and** $\mathbf{L}_2$ **are reg. lang.**

$\Rightarrow \exists$ **DFA** $\mathbf{M}_1$ **and** $\mathbf{M}_2$ **s.t.**

$\mathbf{L}_1 = \mathbf{L}(\mathbf{M}_1)$ **and** $\mathbf{L}_2 = \mathbf{L}(\mathbf{M}_2)$

$\mathbf{M}_1 = (\mathbf{Q}, \Sigma, \delta_1, q_0, \mathbf{F}_1)$

$\mathbf{M}_2 = (\mathbf{P}, \Sigma, \delta_2, p_0, \mathbf{F}_2)$

Construct $\mathbf{M'} = (\mathbf{Q'}, \Sigma, \delta', (q_0, p_0), \mathbf{F'})$

$\mathbf{Q'} = Q \times P$

δ': $((q_i, p_j), a) = (q_k, p_l)$ if

$\delta_1(q_i, a) = q_k \in M_1$ and

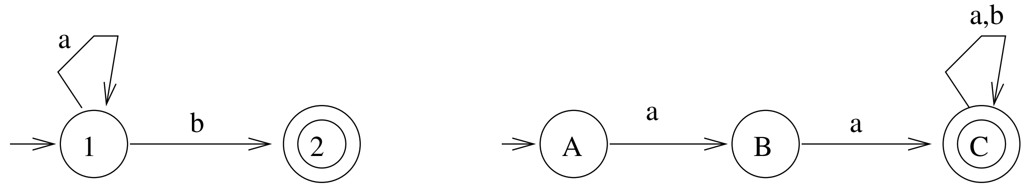
$\delta_2(p_j, a) = p_l \in M_2$

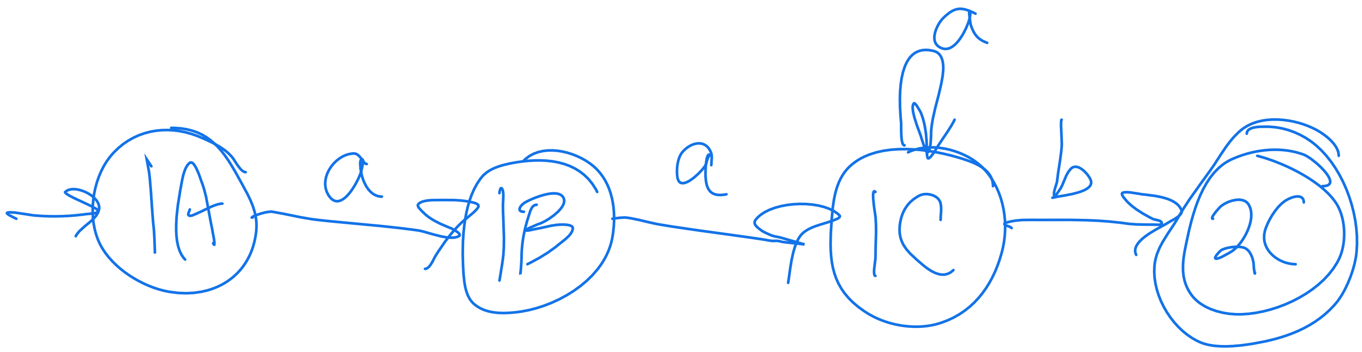
$F' = \{(q_i, p_j) \in Q' \mid q_i \in F_1 \text{ and } p_j \in F_2\}$

show $w \in L(M') \Longleftrightarrow w \in L_1 \cap L_2$

$\Rightarrow$ closed under intersection

Example:





Regular languages are closed under

reversal	$\mathbf{L}^R$
difference	$\mathbf{L}_1$-$\mathbf{L}_2$
right quotient	$\mathbf{L}_1/\mathbf{L}_2$
homomorphism	$\mathbf{h}(\mathbf{L})$

Right quotient

Def: $\mathbf{L}_1/\mathbf{L}_2 = \{x | xy \in \mathbf{L}_1$ **for some** $y \in \mathbf{L}_2\}$

Example:

$\mathbf{L}_1 = \{a^*b^* \cup b^*a^*\}$
$\mathbf{L}_2 = \{b^n | n$ **is even,** $n > 0\}$
$\mathbf{L}_1/\mathbf{L}_2 =$ a^*b^*

$aabb \in L_1/L_2$? yes

$\underbrace{aabb}_{L_1/L_2}\underbrace{bb}_{L_2}$

Theorem If L_1 and L_2 are regular, then L_1/L_2 is regular.

Proof (sketch)

$\exists$ DFA M=(Q,Σ,δ,q_0,F) s.t. $L_1 =$ L(M).

Construct DFA M'=(Q,Σ,δ,q_0,F')

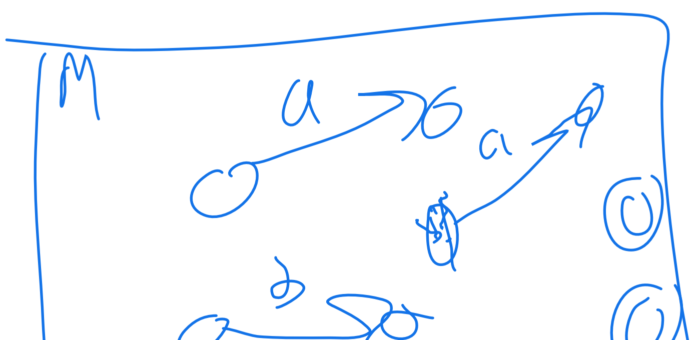
For each state i do

Make i the start state (representing L_i')

if $L_i' \cap L_2 \neq \emptyset$

put q_i in F' in M'

Changing start state to i



QED.

Homomorphism

Def. Let Σ, Γ be alphabets. A homomorphism is a function

$$\mathbf{h}{:}\Sigma \to \Gamma^*$$

Example:

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

$$\mathbf{h(a)=11}$$
$$\mathbf{h(b)=00}$$
$$\mathbf{h(c)=0}$$

$\mathbf{h(bc)} =$ 000

$\mathbf{h(ab^*)} =$ $11(00)^*$

Questions about regular languages :

L is a regular language.

- **Given L, Σ, w$\in \Sigma^*$, is w$\in$L?**

Construct a DFA and test
if it accepts w

- **Is L empty?**

$L = \{a^n b^m \mid n \geq 0, m \geq 0\} \cap \{b^n a^m \mid n > 1, m > 1\}$

DFS Construct NFA $\Rightarrow$ DFA
is there path from start state to final state

- **Is L infinite?**

Construct DFA
is there a cycle?
DFS should know by n steps if n states

- **Does $L_1 = L_2$?**

Construct $L_3 = (L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2)$

If $L_3 = \phi$ then $L_1 = L_2$

Identifying Nonregular Languages

If a language L is finite, is L regular? yes

If L is infinite, is L regular? it depends

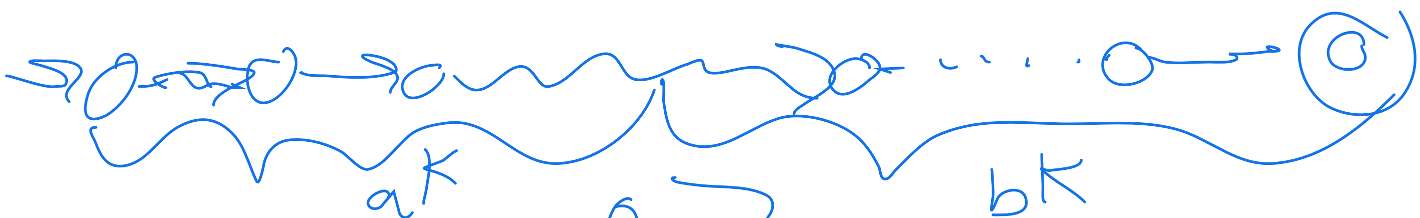
- $L_1 = \{a^n b^m | n > 0, m > 0\} =$ aa^*bb^*
- $L_2 = \{a^n b^n | n > 0\}$ NOT regular

Prove that $L_2 = \{a^n b^n | n > 0\}$ is ?

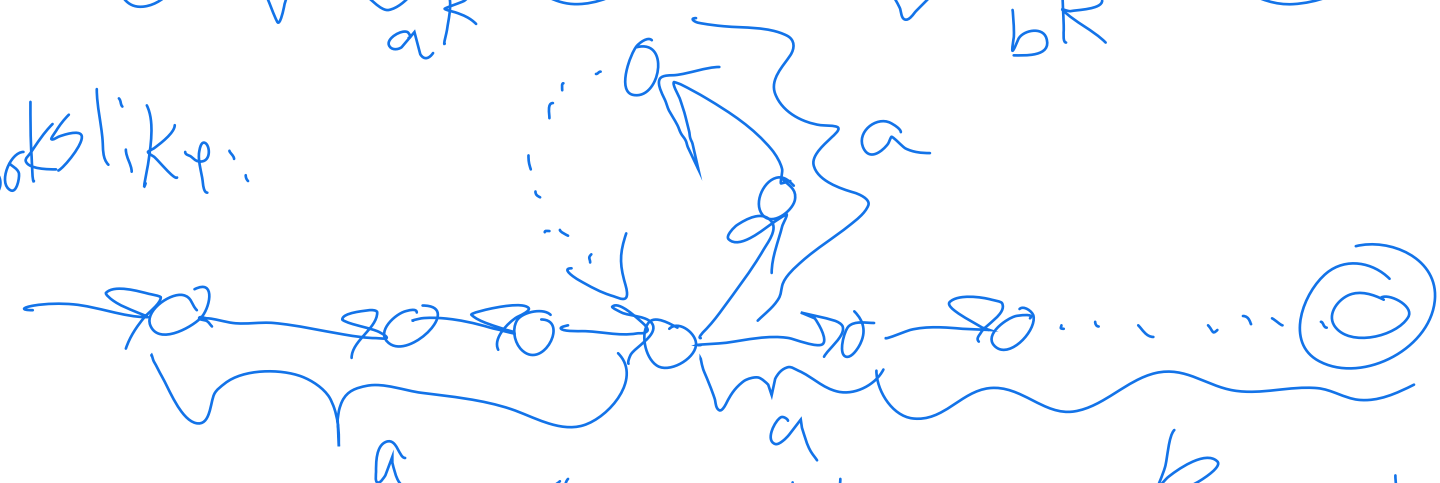
- **Proof: Suppose L_2 is regular.**
  $\Rightarrow \exists$ DFA M that recognizes L_2

M has finite no. of states, k states

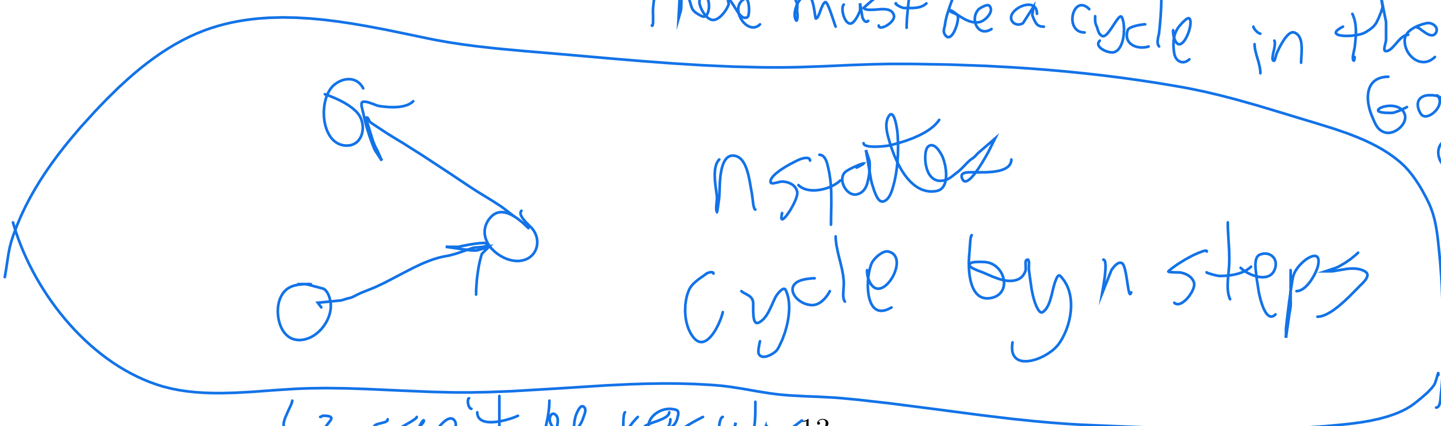
Consider a long string $a^k b^k \in L_2$



looks like:



There must be a cycle in the a's.



Go thru cycle one extra time. that string not in L

L_2 can't be regular

∴ there can't be a dfa ⟶ contradiction

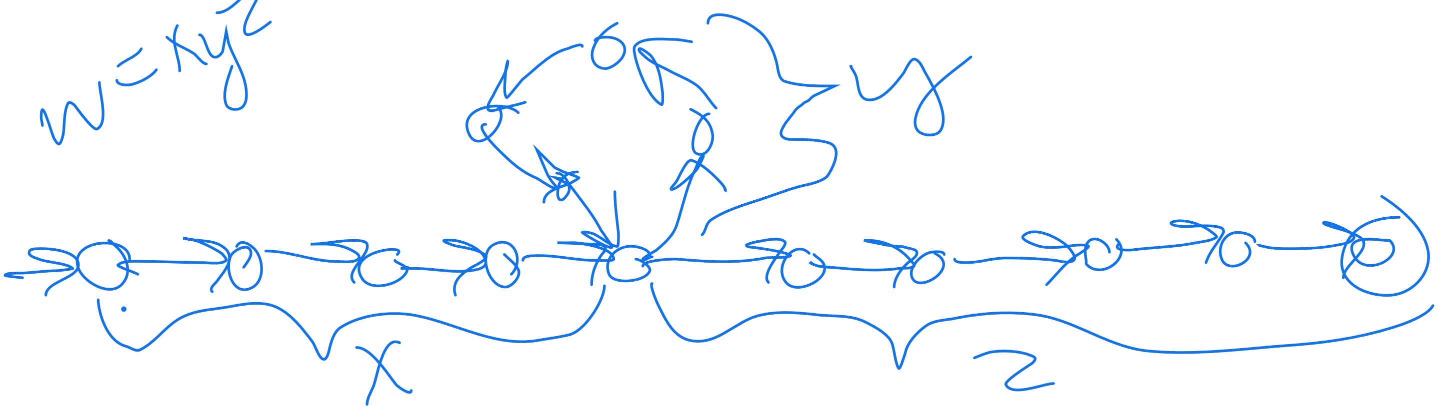
K from before

Pumping Lemma: Let L be an infinite regular language. $\exists$ a constant $m > 0$ such that any $w \in L$ with $|w| \geq m$ can be decomposed into three parts as $w = xyz$ with

no of states in DFA if existed

$$|xy| \leq m$$
$$|y| \geq 1$$
$$xy^i z \in L \text{ for all } i \geq 0$$

$w = xyz$



To Use the Pumping Lemma to prove L is not regular:

- **Proof by Contradiction.**

  Assume L is regular.

  $\Rightarrow$ L satisfies the pumping lemma.

  Choose a long string w in L, $|w| \geq m$.

  Show that there is NO division of w into xyz (must consider all possible divisions) such that $|xy| \leq m$, $|y| \geq 1$ and $xy^iz \in$L $\forall\ i \geq 0$.

  The pumping lemma does not hold. Contradiction!

  $\Rightarrow$ L is not regular. QED.

Example L=$\{a^n c b^n | n > 0\}$

L is not regular.

- **Proof:**

  Assume L is regular.

  $\Rightarrow$ the pumping lemma holds.

  Choose $w =$ $a^m c b^m$

only way partition

$x = a^k \quad y = a^j \quad z = a^{m-k-j} c b^m$

should be true $\quad xy^i z \in L \ \forall i$

pick $i=0$

$xy^0z = xz = a^{m-j} c b^m \notin L$

Contradiction

$\Rightarrow L$ is not regular

$i=2$ also works

STOPPED HERE

Example L=$\{a^n b^{n+s} c^s | n, s > 0\}$

L is not regular.

- **Proof:**

  Assume L is regular.

  $\Rightarrow$ the pumping lemma holds.

  Choose w= $a^m b^{2m} c^m = xyz$ $\quad a^n b^{m+1} c^{\text{any ok}}$

  So the partition is:

all possible partitions $\quad x = a^k \quad y = a^j, j > 0 \quad z = a^{m-k-j} b^{2m} c^m$

$\forall i \; xy^iz \in L$

$i = 0 \quad xz = a^{m-j} b^{2m} c^m \notin L$

$n(a) + n(c) \neq n(b)$

Contradiction!

$i = 2$ also works

L is not regular!

Example $\Sigma = \{a, b\}$,
L=$\{w \in \Sigma^* \mid n_a(w) > n_b(w)\}$

L is not regular.

- **Proof:**

  Assume L is regular.

  $\Rightarrow$ the pumping lemma holds.

  Choose $w=$ $a^{m+1} b^m$

  So the partition is:

all the possible partitions of this string

$x = a^k$ $\quad y = a^j$ $\quad z = a^{m+1-k-j} b^m$

$\forall i \; xy^iz \in L$

$i = 2 \quad xyyz = a^{m+1+j} b^m \in L$

not a contradiction

$i = 0 \quad xy^0z = xz = a^{m+1-j} b^m \notin L$

$j > 0 \quad n_a \ngtr n_b$

Contradiction!

L is not regular

Example L=$\{a^3b^nc^{n-3}|n > 3\}$

(shown in detail on handout)

L is not regular.

Prove several cases

aaabbb . . . . bc c

uvw

$\underbrace{\quad}_{x}\underbrace{\quad}_{y}\underbrace{\qquad\qquad\qquad\qquad}_{z}$

$\underbrace{\qquad}_{x}\underbrace{\qquad}_{y}\underbrace{\qquad\qquad\qquad\qquad}_{z}$

look at handout for details

$\boxed{\text{stop here}}$

To Use Closure Properties to prove L is not regular:

- **Proof Outline:**

  Assume L is regular.

  Apply closure properties to L and other regular languages, constructing L' that you know is not regular.

  closure properties $\Rightarrow$ L' is regular.

  Contradiction!

  L is not regular. QED.

Example L=$\{a^3b^nc^{n-3}|n > 3\}$

L is not regular.

- **Proof: (proof by contradiction)**
  Assume L is regular.
  Define a homomorphism $h : \Sigma \to \Sigma^*$
  $h(a) = a \;\; h(b) = a \;\; h(c) = b$
  $h(L) =$

Example L=$\{a^n b^m a^m | m \geq 0, n \geq 0\}$

L is not regular.

- **Proof: (proof by contradiction)**
  Assume L is regular.

Example: $L_1 = \{a^n b^n a^n | n > 0\}$

L_1 **is not regular.**