— CBN6-280 —

Regular expression over an alphabet Σ are all strings over alphabet $\Sigma \cup \{(,), \emptyset, \cup, *\}$ that can be obtained by

1) $\emptyset$ and each member of Σ is RE
2) If α and β are RE then so does $(\alpha \cup \beta)$ and $(\alpha \cdot \beta)$
3) If α is a RE then so does α^*

nothing else is RE.

Formally given a RE, the language it represents formed by a function $\mathcal{L}$

1) $\mathcal{L}(\emptyset) = \emptyset \qquad \mathcal{L}(a) = \{a\} \quad a \in \Sigma$
2) $\mathcal{L}(\alpha \cup \beta) = \mathcal{L}(\alpha) \cup \mathcal{L}(\beta)$
3) $\mathcal{L}(\alpha\beta) = \mathcal{L}(\alpha)\,\mathcal{L}(\beta)$
4) $\mathcal{L}(\alpha^*) = \mathcal{L}(\alpha)^*$

Σ. $L \subseteq \Sigma^*$ is regular if $L = \mathcal{L}(\alpha)$ for some regular exp α

$$\mathcal{L}((a \cup b)^* a) = \mathcal{L}(a \cup b)^* \mathcal{L}(a)$$
$$= (\mathcal{L}(a) \cup \mathcal{L}(b))^* \mathcal{L}(a)$$
$$= \{a, b\}^* \{a\}$$

— all strings over $\{0,1\}$ that does not contain 11 as a substring

$$\underbrace{0^*}_{\text{no 1's}} \cup \underbrace{(0^* 1 (0 0^* 1)^* 0^*)}_{\text{no consecutive 1's}}$$

— all strings over $\{0,1\}$ that does not contain 111 as a substring.

$$0^* \cup 0^* (1 \cup 11)(0 0^* (1 \cup 11)^*)^* 0^*$$

$$\boxed{\begin{aligned} \mathcal{L}(\emptyset^*) &= \mathcal{L}(\emptyset)^* \\ &= \{\ \}^* \\ &= \{e\} = \text{empty string} \end{aligned}}$$

not in $\emptyset$ but in $\emptyset^*$

$$\underbrace{\underbrace{(a^* b)^*}_{\text{all strings end with } a^* b} \cup \underbrace{(b^* a)^*}_{\text{all strings end with an } a}}_{\text{all strings with } \{a,b\}}$$

$(a \cup b)^*$

Scanned by CamScanner

$\underbrace{(a \cup b)^* \, a \, (a \cup b)^*}$ $\qquad \Sigma = \{a, b\}$

$\rightarrow L = \{ w \in \Sigma^* : w$ contains at least $1\ a \}$

$= \Sigma^* \, a \, \Sigma^*$

$\rightarrow$ all strings in Σ^* that does not contain more than 3 a's

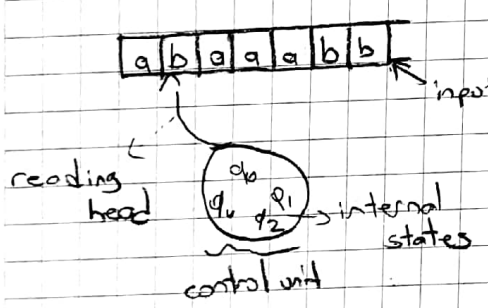
$b^* \cup b^* a b^* \cup b^* a b^* a b^* \cup b^* a b^* a b^* a b^*$

$\Sigma = \{a, b\}$

a	$\{a\}$
e	$\{\}$
ab	$\{ab\}$
$a \cup b$	$\{a, b\}$
a^*	$\{e, a, aa, \ldots\}$

Deterministic Finite State Automate (FSA)

- restricted model of computers



- read one symbol
- update internal state

input
- yes $\Rightarrow$ word is accepted
- No $\Rightarrow$ " " not accepted

The set of words it accepts is the language accepted by the automate.

A FSA is a quintuple: $M = (K, \Sigma, \delta, s, F)$

- K is a finite set of states
- Σ is an alphabet
- $\delta : K \times \Sigma \rightarrow K$ transition function
- $s \in K$ initial state
- $F \subseteq K$ is the set of final states (accepting)

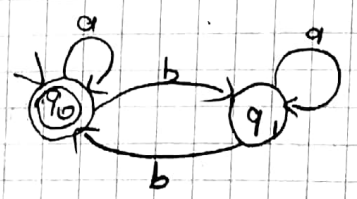
at state q, when $a \in \Sigma$ is read M switches to $q' = \delta(q, a)$

ex)

$K = \{q_0, q_1\}$ $\qquad s = q_0$

$\Sigma = \{a, b\}$ $\qquad F = \{q_0\}$

q	α	$\delta(q, \alpha)$
q_0	a	q_0
q_0	b	q_1
q_1	a	q_1
q_1	b	q_0



Scanned by CamScanner

$(q, w) \in K \times \Sigma^*$: configuration represents the status of the machine

q: state of the machine
w: unread part of the input

$(q_0, abb) \vdash_M (q_0, bb) \vdash_M (q_1, b) \vdash_M (q_0, e)$

$(q_0, ba) \vdash_M (q_1, a) \vdash_M (q_1, e)$

Binary relation (move) $\vdash_M \subseteq ((K, \Sigma^*) \times (K, \Sigma^*))$

$(q, w) \vdash_M (q', w')$ iff $w = aw'$ $a \in \Sigma$, $\delta(q, a) = q'$

$\rightarrow (q, w)$ yields (q', w') in one step

(q, e) : shows that M consumed all input.

$\vdash_M^*$: reflexive transitive closure of $\vdash_M$

$(q, w) \vdash_M^* (q', w')$

A word w is accepted by the automaton iff there is a state $q \in F$ s.t. $(s, w) \vdash_M^* (q, e)$

The language accepted by M is one set of words accepted by M and denoted by $\mathcal{L}(M)$

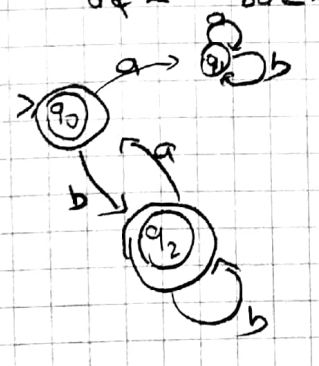
A sequence of configuration is called a computation.

$\mathcal{L}(M) = \{ w \in \Sigma^* : (s, w) \vdash_M^* (q, e), \text{ for some } q \in F \}$

$\mathcal{L}(M) = \{ w \in \Sigma^* : (q_0, w) \vdash_M (q_1, w_1) \vdash_M \dots \vdash_M (q_k, e) \text{ for some } k \geq 0, q_k \in F, s = q_0 \}$

ex) $L = \{ w \in \{a, b\}^* : \text{each } a \text{ is immediately preceeded by a } b \}$

$a \notin L$ $ba \in L$ $baa \notin L$

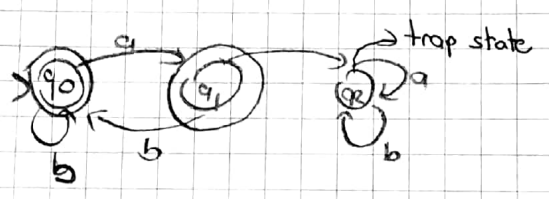


$\mathcal{L}(M) = L$

GIPTA

Scanned by CamScanner

ex) $L = \{w \in \{a,b\}^* ;$ w does not include aa as a substring$\}$



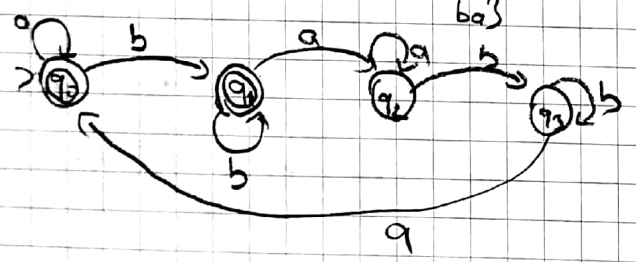
$$\boxed{\begin{array}{l} e \in \mathcal{L}(M) \text{ iff } s \in F \\ (s,e) \overset{*}{\underset{M}{\vdash}} (q,e) \ , q \in F \end{array}}$$

$\mathcal{L}(M) = \emptyset$
$\{\ \}$

ex)
$\Sigma = \{a,b\}$
$K = \{q_0, q_1\}$
$M = (K, \Sigma, \delta, s, F)$ how many different automata?

$\delta : K \times \Sigma \to K$
4 2 2^4
$s \in K$ 2
$F \subseteq K$ 4

ex)
Construct a DFA $L \{w \in \{a,b\}^* ;$ w contains even # of occurences of substring ba$\}$



Nondeterministic Finite State Automata (NFA)

$M = (K, \Sigma, \Delta, s, F)$ $\Delta \subseteq K \times (\Sigma \cup \{e\}) \times K$

transition relation
$(q, a, q') \in \Delta$ $q' \neq q''$
$(q, a, q'') \in \Delta$

"several options for next states"

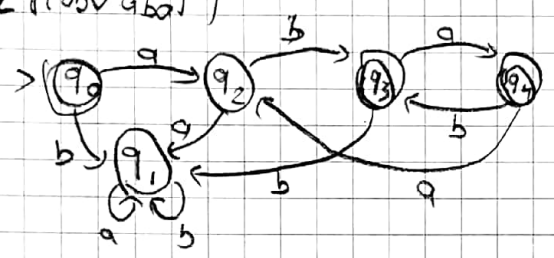
Scanned by CamScanner

$(q,w) \vdash_M (q',w')$ iff
$w = aw'$ $\quad a \in \Sigma \cup \{e\}$
$(q,a,q') \in \Delta$

$\vdash_M^*$ reflexive transitive closure of $\vdash_M$ w is accepted by M iff there exist $q \in F$ s.t. $(s,w) \vdash_M^* (q,e)$

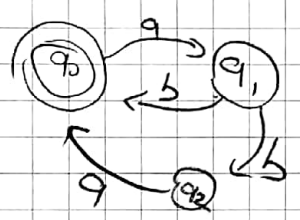
M s.t $\Sigma = \{\}$ $\quad L(M) \neq \emptyset$ $\quad > \textcircled{0}$

$\mathcal{L}((ab \vee aba)^*)$



← DFA

← NFA

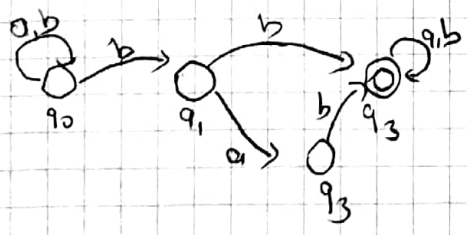


"aba":
$(q_0, aba) \vdash_M (q_1, ba) \vdash_M (q_0, a) \vdash_M (q_1, e)$ −
$\rightarrow \vdash_M (q_2, a) \vdash_M (q_0, e)$ +

NFA:
If all possibilities end in not acceptable states, word is not in language. Else if, it has at least one possible computation ends in acceptable state, the word is in language.

$M = (K, \Sigma, \Delta, s, F)$
$\Delta \subseteq K \times \Sigma \times K$
$\mathcal{L}(M) = \{w \in \Sigma^* : (s,w) \vdash_M^* (q,e)$ for some $q \in F\}$

ex $L = \{w \in \{a,b\}^* : w$ contains bb or bab$\}$



GIPTA

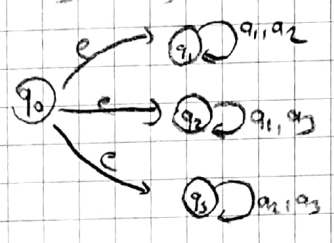
Scanned by CamScanner

$\Sigma = \{a_1, \dots a_n\}$

$L = \{ w \in \Sigma^* : \text{there is a symbol } a_i \in \Sigma \text{ that does not appear in } w \}$

for $n=3$:

$\Sigma = \{a_1, a_2, a_3\}$



Theorem: Two finite automata (deterministic or nondeterministic) A_1 and A_2 are equivalent iff $\mathcal{L}(A_1) = \mathcal{L}(A_2)$

$M = (K, \Sigma, \delta, s, F)$ $\qquad$ $M = (K, \Sigma, \Delta, s, F)$

$\delta : K \times \Sigma \to K$ $\qquad$ $\Delta \subseteq K \times \Sigma \times K$

$\Delta = \{ (q, a, q') : (q, a) \in K \times \Sigma,\ q' = \delta(q, a) \}$

For any language accepted by a DFA there exists a NFA that accepts the same language.

(same above)

Theorem: For each NFA, there exists an equivalent DFA

Constructive Proof

NFA $M = (K, \Sigma, \Delta, s, F)$

DFA $M = (K, \Sigma, \delta, s, F)$

1) $K' = 2^K$

2) $F' = \{ f \in K' : f \cap F \neq \emptyset \}$

States of DFA represent set of state of NFA that can be reached by the input so far.

$K = \{ q_0, q_1 \}$

$K' = \{ \underset{(?)}{\{\}}, \{q_0\}, \{q_1\}, \{q_0, q_1\} \}$

$M = (K, \Sigma, \delta, s, F)$

$M = (K, \Sigma, \Delta, s, F) \qquad \Delta \subseteq K \times (\Sigma \cup \{e\}) \times K$

Theorem

For each NFA, M', there exists a DFA M that accepts the same language, $L(M) = L(M')$

$M = (K, \Sigma, \delta, s, F)$ (DFA)

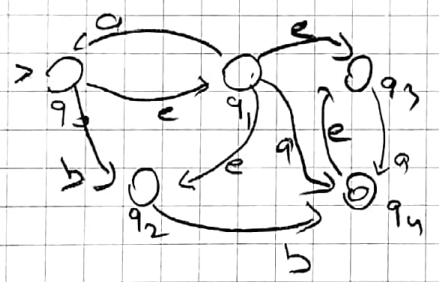
$M' = (K', \Sigma, \Delta, s', F')$ (NFA)

1) $K' = 2^K$

2) $F' = \{f \in K' : f \cap F \neq \emptyset\}$

$E(q) = \{p \in K : (q, e) \overset{*}{\vdash}_M (p, e)\}$

$E(q)$: the set of states of M that can be reached from q without reading any symbol from the input tape.



$K = \{q_0, q_1, q_2, q_3, q_4\}$

$s = q_0 \qquad F = \{q_4\}$

$E(q_0) = \{q_0, q_1, q_2, q_3\}$

$\delta(Q, a) = \bigcup \{E(p) : p \in K$ and $(q, a, p) \in \Delta$ for some $q \in Q\}$

the set of states that are reachable from Q by reading symbol a.

$\bigcup_{q \in Q} \{p : (q, a) \overset{*}{\vdash}_M (p, e)\}$

$E(q_1) = \{q_1, q_2, q_3\}$

$E(q_2) = \{q_2\}$

$E(q_3) = \{q_3\}$

$E(q_4) = \{q_3, q_4\}$

$\delta(\{q_0, q_1, q_2, q_3\}, b)$

$= E(q_2) \cup E(q_4)$

$= \underline{\{q_2, q_3, q_4\}}$

$\delta(\{q_0, q_1, q_2, q_3, q_4\}, a)$

$= E(q_0) \cup E(q_4)$

$= \underline{\{q_0, q_1, q_2, q_3, q_4\}}$

$\delta(\{q_0, q_1, q_2, q_3, q_4\}, b)$

$= E(q_4) \cup E(q_2)$

$= \{q_2, q_3, q_4\}$

Scanned by CamScanner

$\delta(\{q_2, q_3, q_4\}, a)$

$= E(q_4)$

$= \{q_3, q_4\}$

$\delta(\{q_2, q_3, q_4\}, b)$

$= E(q_4)$

$= \{q_3, q_4\}$

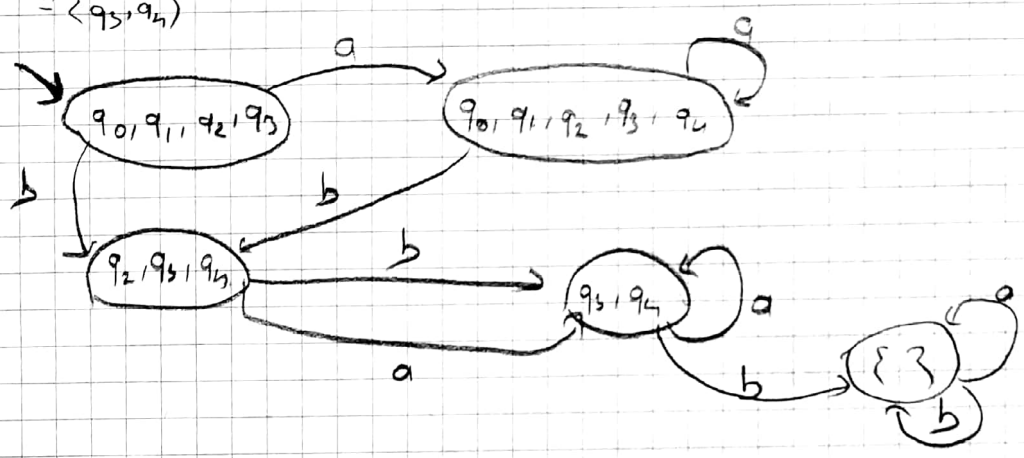
$\delta(\{q_3, q_4\}, a)$

$= E(q_4)$

$= \{q_3, q_4\}$

$\delta(\{q_3, q_4\}, b)$

$= \{\ \}$



Prove that

a) M' is DFA

b) $L(M) = L(M')$

a) S is single value and well defined for any $(q, a) \in Q' \times \Sigma$

To show that

$L(M) = L(M')$ consider any string $w \in \Sigma^*$

- $w \in L(M)$ iff $(s, w) \vdash_M^* (f, e)$ for some $f \in F$

$(\underbrace{E(s)}_{\text{initial state of } M'}, w) \vdash_{M'}^* (T, e)$ final state s.t. $f \in T$, $T \in F' \Rightarrow w \in L(M')$

prove the claim by induction on $|w|$ Basis step $|w| = 0$ $w = e$

$\underline{(q, e) \vdash_M^* (p, e)}$ iff

$p \in E(q)$ $(E(q), e) \vdash_{M'}^* (P, e)$ for some $p \in P$ $P = E(q)$

Scanned by CamScanner

$\Sigma+1$ Claim holds for all w with $|w| \leq k$

Show that claim holds for w of length $k+1$

$w = va$, $a \in \Sigma$

$|w| = k$, $v \in \Sigma^*$

<u>Suppose that</u>

$(q, w) \vdash_M^* (p, e)$

$r_1, r_2 \in K$ s.t. $\underbrace{(q, va) \vdash_M^* (r_1, a)}_{\text{read } v} \underbrace{\vdash_M (r_2, e) \vdash_M^* (p, e)}_{\text{read } a}$

$(q, r) \vdash_M^* (r, e)$ by IHR, $(E(q), v) \vdash_M^*, (R_1, e)$ $r_1 \in R_1$

By construction of M' $r_2 \in \delta(R_1, a)$ $p \in E(r_2)$, $p \in \delta(R_1, a)$

$R_e \in K'$

$R_2 = \delta(R_1, a)$

$r_2, p \in R_2$

$(E(q), va) \vdash_M^* (R_1, a) \vdash_M (R_2, e)$ $p \in R_2$

<u>To complete the proof: show</u>

$(E(q), w) \vdash_M^* (P, e)$ implies $(q, w) \vdash_M^* (p, e)$

$(E(q), va) \vdash_M^* (R_1, a) \vdash_M^* (P, e)$

read the rest

Finite Automata and Regular Expressions

RL: the closure of finite languages under concatenation $(\{\}, \{a\}$ $a \in \Sigma\}$ union Kleene star.

The class of languages accepted by finite automaton is closed under

(a) union
(b) concatenation
(c) Kleene star
(d) complementation
(e) intersection

GIPTA

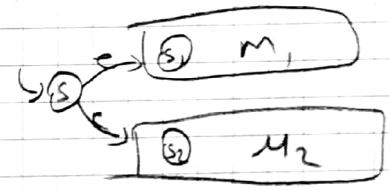
Scanned by CamScanner

a) L_1, L_2 $\quad L_1 = L(M_1)$ $\quad L_2 = L(M_2)$

M_1, $\quad L(M) = L_1 \cup L_2$

$M_1 = (K_1, \Sigma, \Delta_1, s_1, F_1)$ $\quad M_2 = (K_2, \Sigma, \Delta_2, s_2, F_2)$

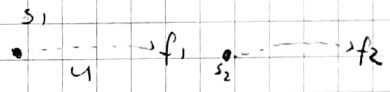
$M = (K, \Sigma, \Delta, s, F)$ st $L(M) = L(M_1) \cup L(M_2)$



the class of languages accepted by FA is the same as the class of regular languages

2) $L(M) = L(M_1) \cdot L(M_2)$

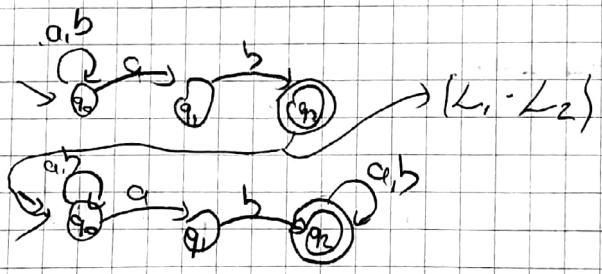
$w = uv$
$u \in L(M_1)$
$v \in L(M_2)$



$\Delta = \Delta_1 \cup \Delta_2 \cup \{(f, \varepsilon, s_2) : f \in F_1\}$

$s = s_1$
$F = F_2$

$L_1 = \{w \in \{a,b\}^* : w$ ends with $ab\}$
$L_2 = \{w \in \{a,b\}^* : w$ contains $ab\}$



Scanned by CamScanner

3) $L(M) = L(M_1)^*$

$e, a, \in L(M_1)$

$a, b \quad , \quad a \in L(M_1)$

$b \in L(M_1)$

$K = K_1 \cup \{s\}$

$\Delta = \Delta \cup \{(s, e, s_1)\} \cup \{(f, e, s_1) : f \in F_1\}$

$F = F_1 \cup \{s\}$

4) Complementation

$L(M) = \overline{L(M_1)}$

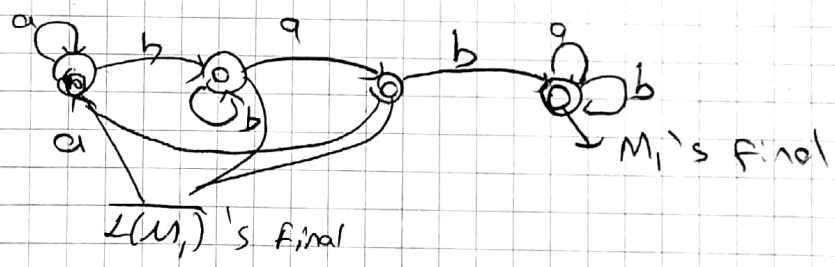
DFA

$w : (q_0, w) \vdash_M \cdots \vdash_M (q_f, e)$

$M_1 = (K, \Sigma, \delta_1, s, F_1)$

$K = K_1, \quad \delta = \delta_1, \quad s = s_1$

$L(M_1) = \{w \in \{a,b\} : w$ contains "bab" as a substring$\}$



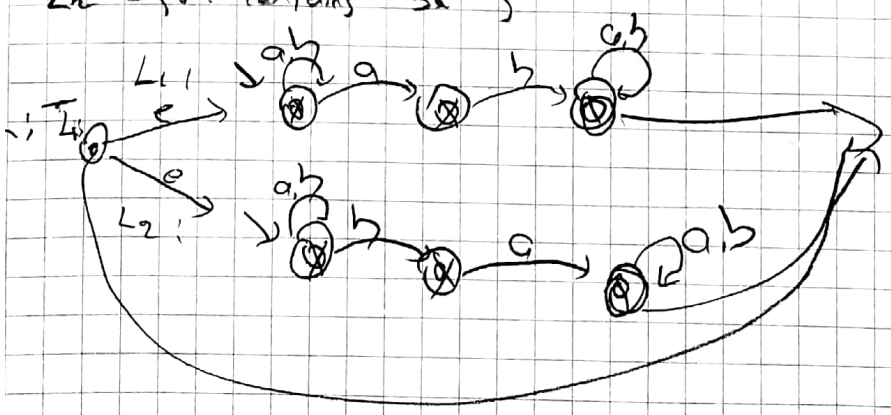
5 $L(M) = L(M_1) \cap L(M_2)$

$L(M_1) \cap L(M_2) = \Sigma^* - \left((\Sigma^* - L(M_1)) \cup (\Sigma^* - L(M_2)) \right) = \overline{\left(\overline{L(M_1)} \cup \overline{L(M_2)} \right)}$

$L_1 = \{w : w$ contains "ab"$\}$

$L_2 = \{w : $ contains "ba"$\}$

$L(M) = L_1 \cap L_2$



Scanned by CamScanner

Thm: A language is regular iff it is accepted by a finite automaton.

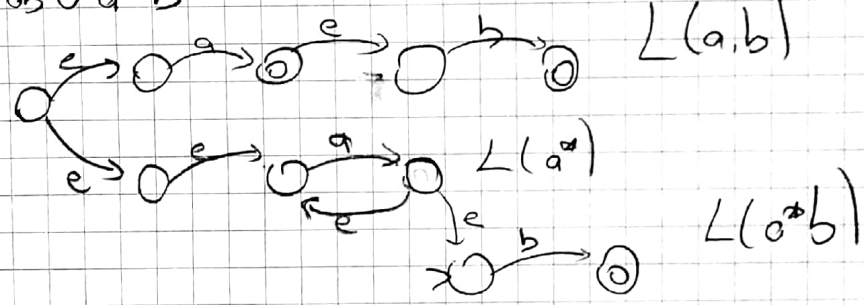
(only if) For any RE α, there is a FA M such that $L(\alpha)=L(M)$

$\underbrace{\emptyset, \{\{a\} : a \in \Sigma\}}_{\text{contain these and closed under union, concatenation, Kleene star}}$

The languages accepted by FA is closed under union, concatenation, Kleene star.

For every RL there is a FA that recognize it.

α) $ab \cup a^*b$



$L(a,b)$

$L(a^*)$

$L(a^*b)$

if) $M=(K,\Sigma,\Delta,s,F)$, write α

$L(\alpha)=L(M)$

$K=\{q_1,\dots,q_n\}$ $s=q_1$

$R(i,j,k)=\{w \in \Sigma^* : (q_1,w) \vdash_M (q_j,e)$ without passing through any intermediate stated numbered $k+1$ or higher$\}$

$L(M)=\bigcup_{q_j \in F} R(1,j,n)$ } If $R(i,j,k)$ is regular, then $L(M)$ is regular

prove $R(i,j,k)$ is regular)

Induction on k

Scanned by CamScanner

$R(i,j,0)$

$$= \begin{cases} \{a \in \Sigma \cup \{e\} : (q_i, a, q_j) \in \Delta\} & i \neq j \\ \{e\} \cup \{a \in \Sigma : (q_i, a, q_j) \in \Delta\} & i = j \end{cases}$$

↳ Finite sets $R(i,j,0)$ is regular.

Assume that $R(i,j,m)$ is regular for any i,j and $m \leq k-1$

Show that $R(i,j,k)$ is regular

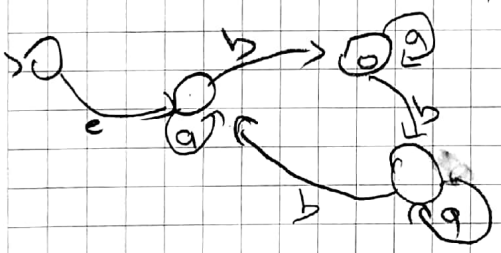
$R(i,j,k) = R(i,j,k-1) \cup$ no passing through q_k

$R(i,k,k-1)\, R(k,k,k-1)^* R(k,j,k-1)$

$\Rightarrow$ all of them regular so $R(i,j,k)$ is regular

ex) $L = \{w \in \{a,b\} \mid w \text{ has } 3k+1 \text{ b's for some } k \in N\}$

show that L is regular
write a RE for L



Scanned by CamScanner