Chapter 1 - Regular expressions and automata over finite words

I] Words and languages

- alphabet, ex: $\Sigma_1 = \{a, b, c\}$, $\Sigma_{fr} = \{a, b, c, d, e, é, f, \dots, y, z, ì, í\}$
- word, written $w = a_0 a_1 \cdots a_{n-1}$, n length
  ex: $w = bbacabc$ $\quad |w| = 7$
- empty word, $a \equiv a$
- $|w|_a$

Fact: $|w| = \sum_{a \in \Sigma} |w|_a$

- Σ^*, Σ^n $\quad$ convention: $\emptyset^* = \{\varepsilon\}$

Fact: $|\Sigma|^n$

- a^n, $w_1 \cdot w_2$ $\quad w\varepsilon = \varepsilon w = w$ $\quad$ Fact: $|w_1 + w_2| = |w_1| + |w_2|$

Harmless facts: $(w_1 \cdot w_2) \cdot w_3 = w_1 \cdot (w_2 \cdot w_3)$

- w^n $\quad$ Fact: $|w^n| = |w| \times n$
- language $\quad$ ex: $L_\emptyset$, L_ε, $L_1 = \{abc, aaac\}$, $L_{|a|=|b|}$, $L_{a^*b^*c^*}$
- III Fact: lang infinite iff words of unbounded (sizes) / length

II] Regular expressions

α) inductive def°

- a $\quad$ • $\emptyset$ $\quad$ • E^c, E^* $\quad$ • $E \cup F$ ($E+F$), $E \cap F$, $E \cdot F$

  ex: $[(a \cup b)\cdot(a+b)]^* \cap (a \cup b)^* \cdot a \cdot a \cdot (a+b)^* \cap [(a \cup b)^* \cdot b \cdot b \cdot (a \cup b)^*]^c$

β) ~~satisfaction~~ regular language

- $a \models a$, $w \not\models \emptyset$ ...
- $\mathcal{L}(E)$, example above $\quad$ ex: $a \cdot (a \cup b)^* \cdot c$
- regular L

III] Automata

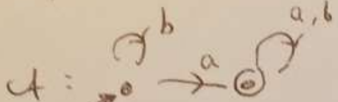
1) def and example

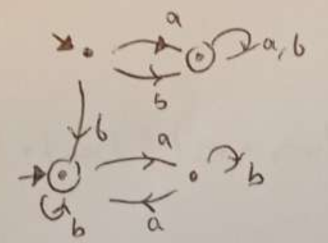
- $\langle Q, \delta, I, F \rangle$ $\qquad p \xrightarrow{a} q$ for $\langle p, a, q \rangle \in \delta$ $\qquad$ if $\mathcal{A}$ then $Q_{\mathcal{A}}$..
- ex : $Q_{\mathcal{A}} = \{0, \dots, 4\}$, $\delta_{\mathcal{A}} = \{ \langle 0, a, 1 \rangle, \langle 0, a, 2 \rangle, \langle 0, b, 3 \rangle, \langle 1, b, 0 \rangle, \langle 2, a, 0 \rangle, \langle 2, b, 2 \rangle, \langle 2, b, 3 \rangle, \langle 3, a, 4 \rangle, \langle 4, a, 4 \rangle \}$

$I_{\mathcal{A}} = \{0, 3\}$ $\qquad F_{\mathcal{A}} = \{3, 4\}$

1 4
0
3
2

- run from p to q via w $\qquad p \xrightarrow[\mathcal{A}]{w} q$
- accepting run
- $\mathcal{A}$ reco w if..
- $\mathcal{L}(\mathcal{A})$

ex : $\mathcal{A}$:  $\mathcal{L}(\mathcal{A}) = ?$

$\mathcal{B}$ =  $\mathcal{L}(\mathcal{B}) = ?$ everything

- infinite but finite (like expr)

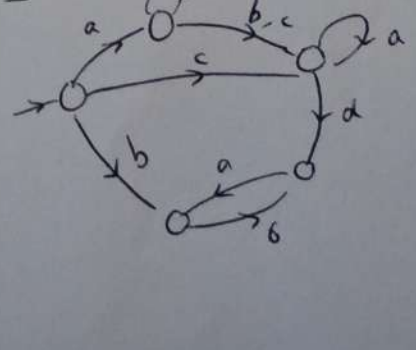
IV] Automata and regular languages

- theorem

α) from reg expr to automata

- ε transitions

β) 

ex : 

Determinisation

- def

Fact : for every $p \in Q_{\mathcal{A}}$, $w \in \Sigma^*$, unique $q \in Q_{\mathcal{A}}$ s.t. $p \xrightarrow[\mathcal{A}]{w} q$, $q = \delta_{\mathcal{A}}(p, w)$

Prop : for every $\mathcal{A}$, exists $\mathcal{D}$ DFA s.t. $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{D})$.

proof : states of $\mathcal{D}$ subsets of $Q_{\mathcal{A}}$

DET $(\mathcal{A})$:=

$i_{\mathcal{D}} := I_{\mathcal{A}}$;

$Q_{\mathcal{D}} := \{ i_{\mathcal{D}} \}$; $(TBD = \{ i_{\mathcal{D}} \})$

while new states P :

for every $a \in \Sigma$

$P_a := \{ q \in Q_{\mathcal{A}} \mid \delta_{\mathcal{A}}(p, a) = q$ for some $p \in P \}$;

$\delta_{\mathcal{D}} \leftarrow \langle P, a, P_a \rangle$;

$Q_{\mathcal{D}} \leftarrow P_a$

endfor

endwhile

$F_{\mathcal{D}} := \{ P \in Q_{\mathcal{D}} \mid P \cap F_{\mathcal{A}} \neq \emptyset \}$ □

Prop : it is decidable whether a regular language is the empty language or not

VI] Pumping lemma

α) Pumping lemma / statement

- Pumping lemma $|x \cdot y| \leq N$, $|y| \geq 1$
- proof

β) Application

Prop : $L_{|a|=|b|} = \{ u \in \{a,b\}^* \mid |u|_a = |u|_b \}$ not regular

VII] Myhill-Nerode right congruence, and minimal automaton

- $u \sim_L v$ iff for every z..

Prop: eq + for all u, v, w s.t. $u \sim_L v$, $uw \sim_L vw$.

- eq class

- $\mathcal{A}$ DFA, $u \sim_{\mathcal{A}} v$

Prop: if $u \sim_{\mathcal{A}} v$ then $u \sim_{\mathcal{L}(\mathcal{A})} v$

Theorem: L regular iff $\sim_L$ finitely many classes

ex: $L_{a<b} = \{w \mid \exists a < b\}$