GNFA

Regular expressions over Σ

$$G = (Q, \Sigma, \delta, q_0, q_f)$$

$Q, \Sigma, q_0, q_f \in Q$ as usual

$$\delta : (Q - \{q_f\}) \times (Q - \{q_0\}) \to R_\Sigma$$

Defn

- G can be in state $q \in Q$ after reading $x \in \Sigma^*$ if $\exists k \geq 0$,

  $\exists\, r_0, r_1, \cdots, r_k \in Q$

  $\exists\, x_1, \cdots, x_k \in \Sigma^*$

  such that

  (i) $x = x_1 \cdot x_2 \cdot \cdots \cdot x_k$

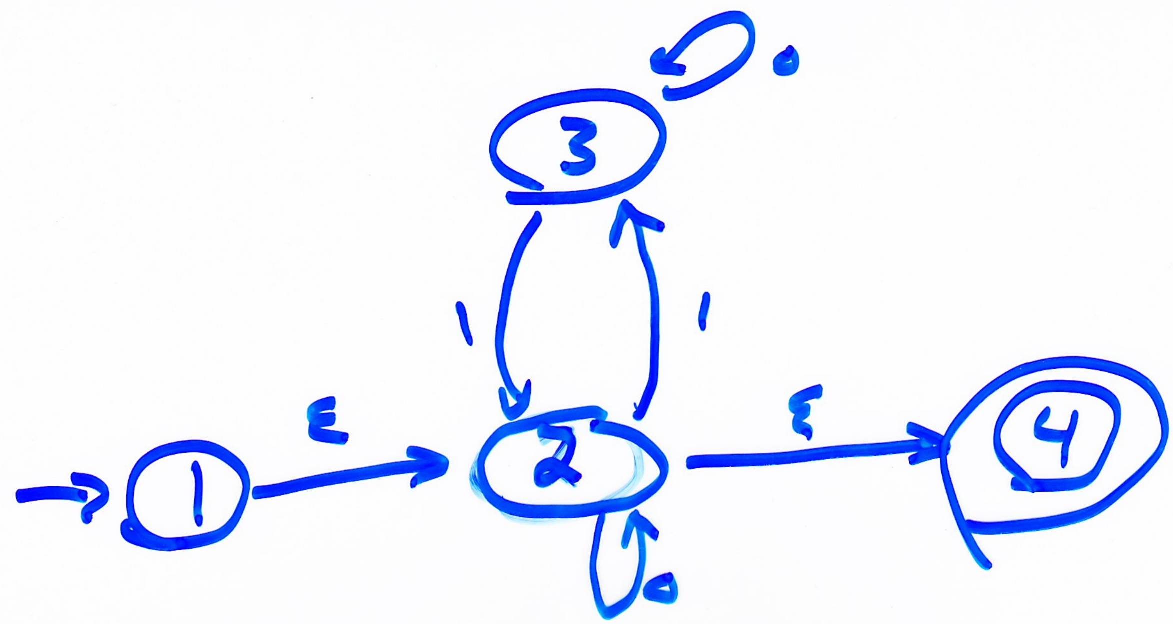
  (ii) $r_0 = q_0$

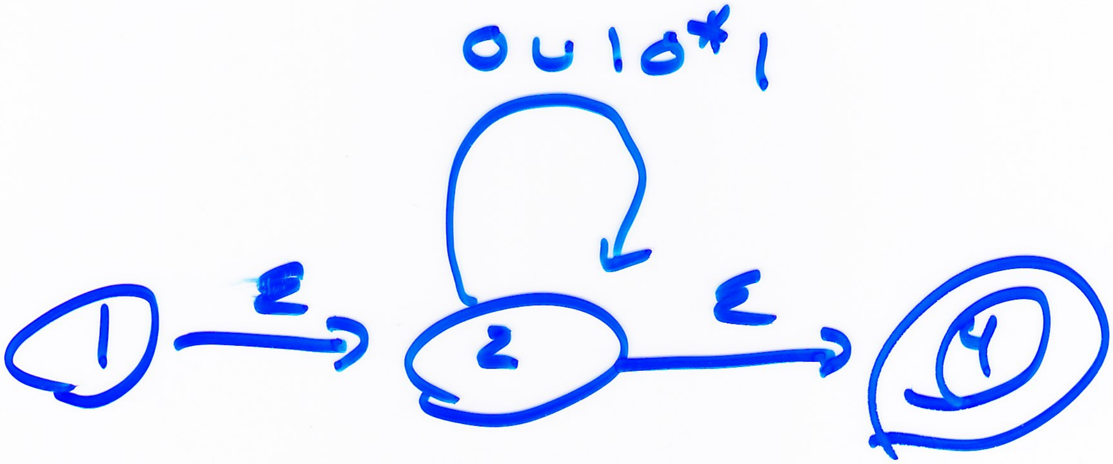
  (ii) $r_k = q$

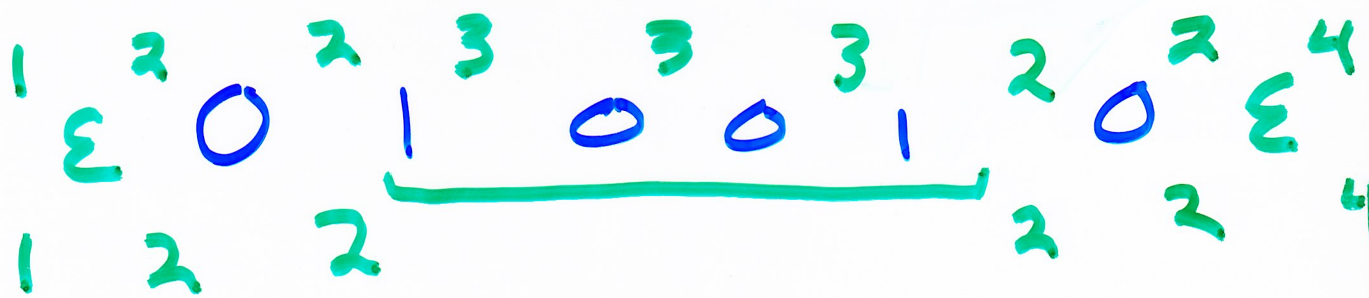
  (iii) $\forall\, 1 \leq i \leq k,\ x_i \in L(\delta(r_{i-1}, r_i))$

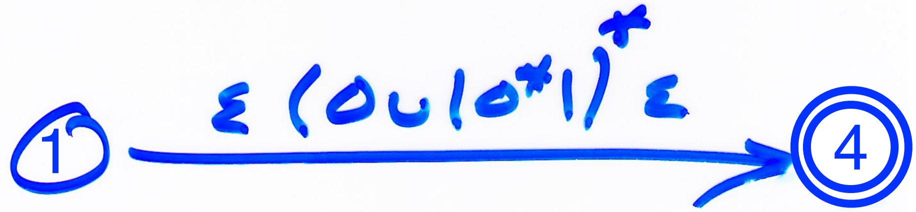
- $L(G) = \{ x \mid G \text{ can be in state } q_f \cdots \}$

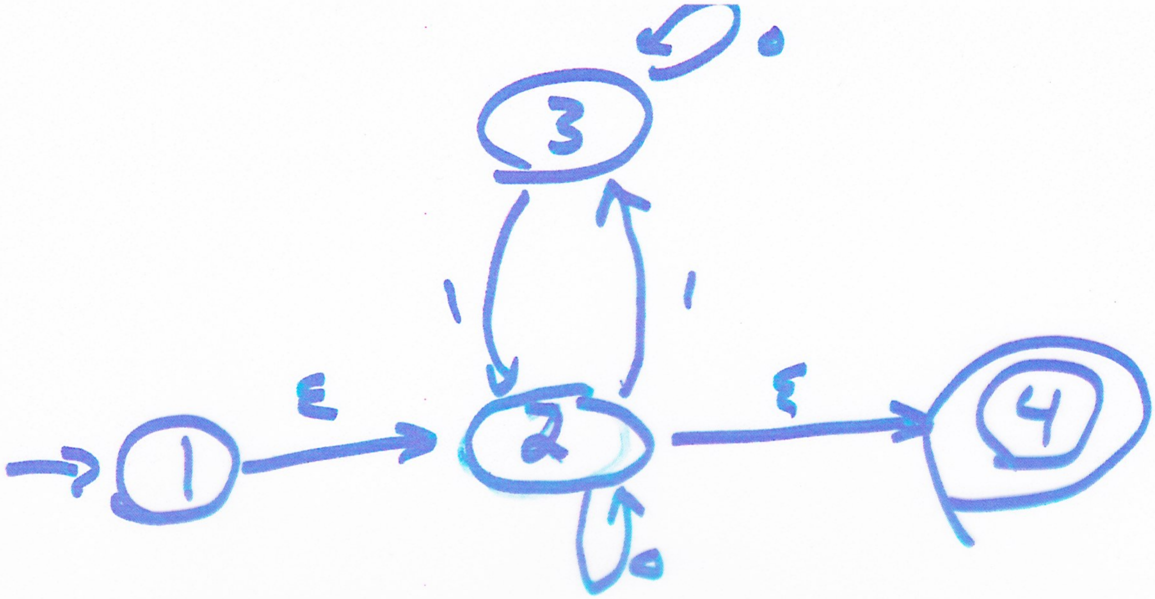
Note: δ syntax a little different; maps state pair to label (reg. exp.) rather than state $\times$ symbol $\to$ new state

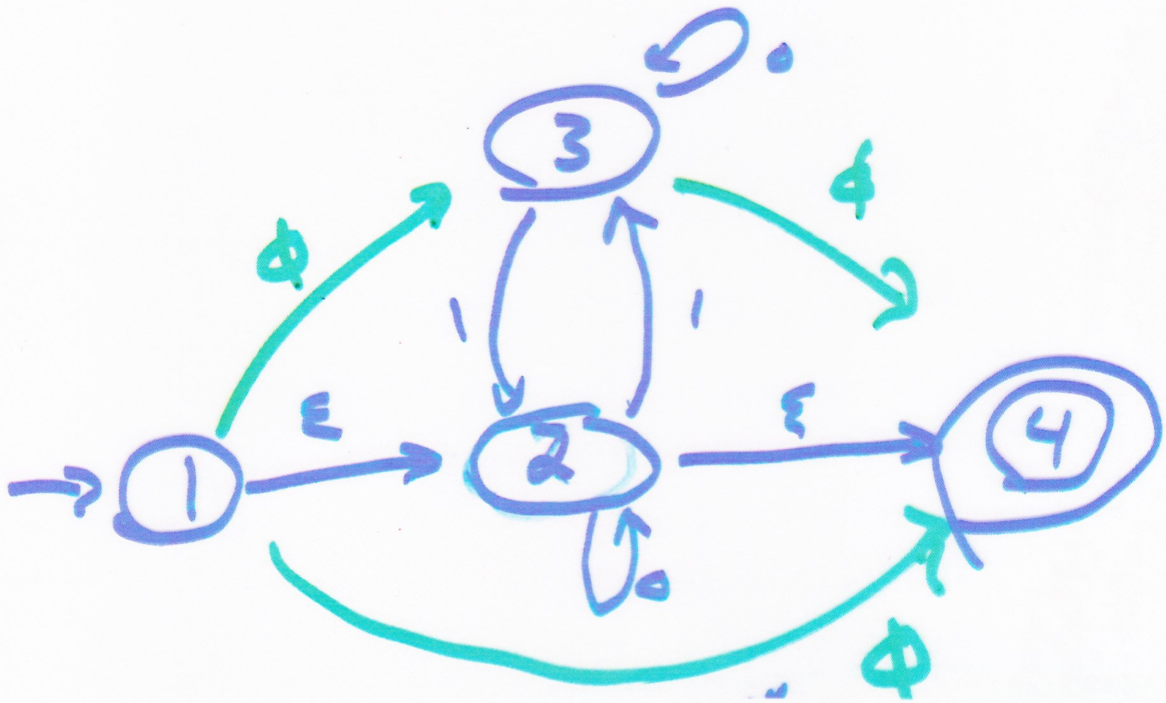

3
0
1
1
1
ε
2
ε
4
0

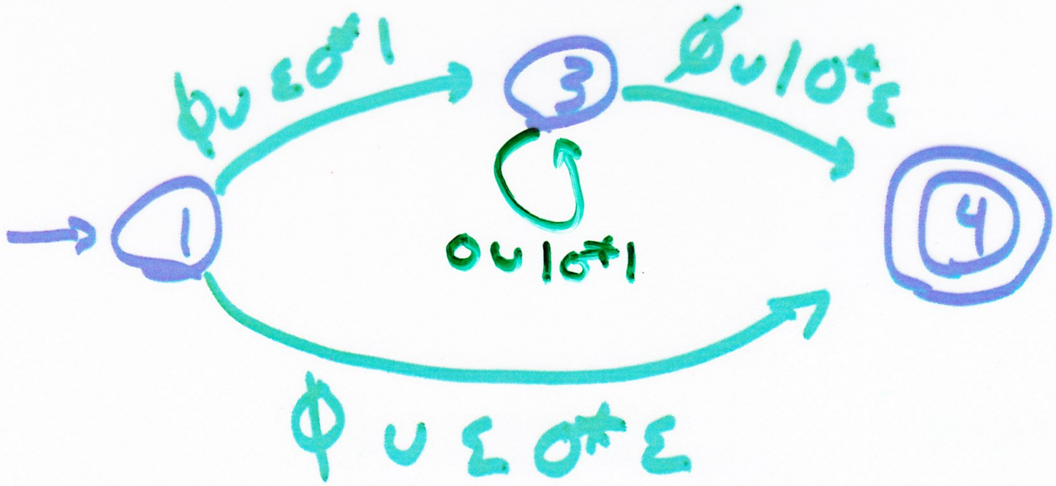

0∪10*1
1
ε
2
ε
4


1 2 2 3 3 3 2 2 4
ε 0 1 0 0 1 0 ε
1 2 2 2 2 4


ε(0∪10*1)*ε
1
4







$$\to (1) \xrightarrow{(\phi \cup \varepsilon 0^* \varepsilon)\ \cup\ (\phi \cup \varepsilon 0^* 1)(0 \cup 1 0^* 1)^* (\phi \cup 1 0^* \varepsilon)} (4)$$

Given GNFA $G = (Q, \Sigma, \delta, q_0, q_f)$ with > 2 state } "the old machine"

Notation $\forall q_i \neq q_f, q_j \neq q_0$

$$r_{ij} = \delta(q_i, q_j)$$

Pick anystate $q_k \neq q_0, q_f$

Build GNFA $G' = (Q', \Sigma, \delta', q_0, q_f)$ with one less state as follows: } "the new machine"

$$Q' = Q - \{q_k\}$$

$$\delta'(q_i, q_j) = r'_{ij} = r_{ij} \cup r_{ik} r_{kk}^* r_{kj}$$

Claim! G & G' are equivalent

To prove this, it is useful to focus on a sub problem: how do <u>edges</u> in G' relate to <u>paths</u> in G?

Relating edges of G′ to paths of G

A path in G : any sequence of states

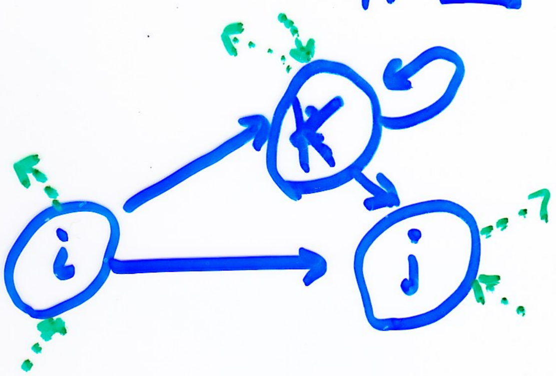
A simple path in G : any sequence of ≥ 2 states s.t. 1st & last are not k, and all intermediate ones (if any) are k.

$i \to j$

$i \to k \to j$

$i \to k \to k \to j$

$\vdots$



Other edges ignored/forbidden

The Point:

(a) every path in G can be decomposed into simple paths

(b) every edge in G′, say $i \to j$, corresponds to the set of all simple paths in G with those endpoints

Claim 2

$L(r'_{ij}) = \{ w \mid G$ can move from i to j reading w and passing through no intermediate states except possibly k. $\}$

Equivalently:

$L(r'_{ij}) = \{ w \mid G$ can move from i to j reading w along a simple path $\}$

Suppose w accepted by G

Then $\exists$ states $s_0\ s_1 \dots s_n \in Q$

and strings $w_1, w_2, \dots, w_n \in \Sigma^*$

st $w_i \in L(r_{i-1}, r_i)$

$w = w_1 \cdot w_2 \cdots w_n$

$s_0 = q_s$

$s_n = q_f$

$w_6 \in r_{6K}$

$w_7 \in r_{K7}$

$\therefore w_6 \cdot w_7 \in L(r_{6K} \cdot r_{KK}^* \cdot r_{K7})$

G

$$\underset{q_s}{\overset{\parallel}{s_0}} \quad \underbrace{s_1 \quad s_2}_{w_2} \quad \underset{K}{\overset{\parallel}{s_3}} \quad \underset{K}{\overset{\parallel}{s_4}} \quad s_5 \quad \underset{K}{\overset{\parallel}{s_6}} \quad \overbrace{s_7 \quad s_8}^{w_8} \quad \underset{q_f}{\overset{\parallel}{s_8}}$$

$w_1 \in L(r_{s_0 s_1})$

G'

$$\underbrace{s_0 \quad s_1}_{w_1} \quad \underbrace{s_1 \quad s_2}_{w_2} \quad \underbrace{\overbrace{s_2 \quad s_5}^{r_{2K} r_{KK}^* r_{K5}}}_{w_3 \cdot w_4 \cdot w_5} \quad \overbrace{s_5 \quad s_7}^{w_6 \cdot w_7} \quad \underbrace{s_7 \quad s_8}_{w_8}$$