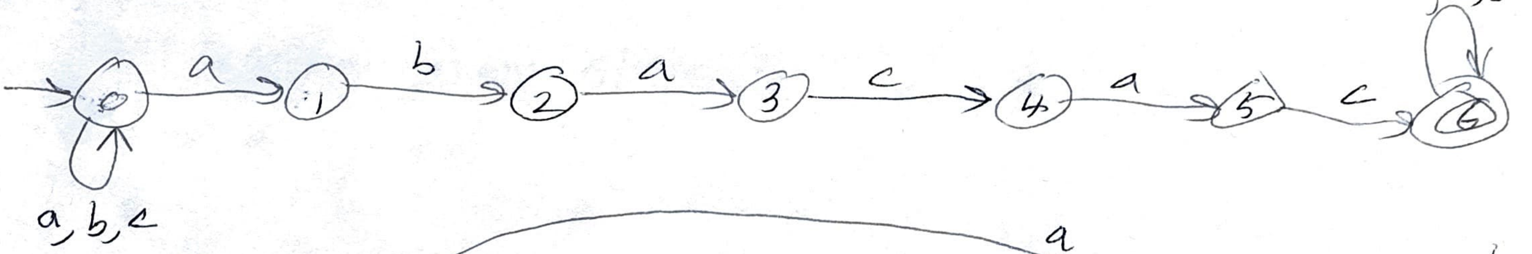
CSCE 355
1/28/2026

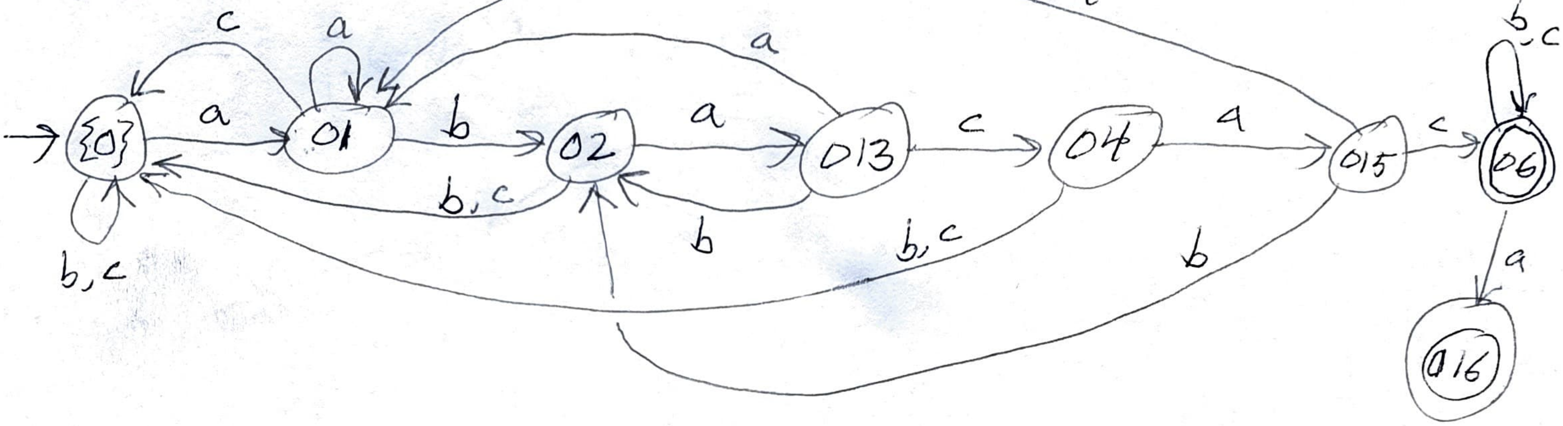
NFA $\longrightarrow$ DFA example(s)

ε-transitions

Text search, for substring abacac

NFA

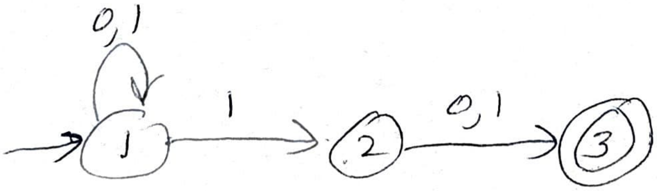




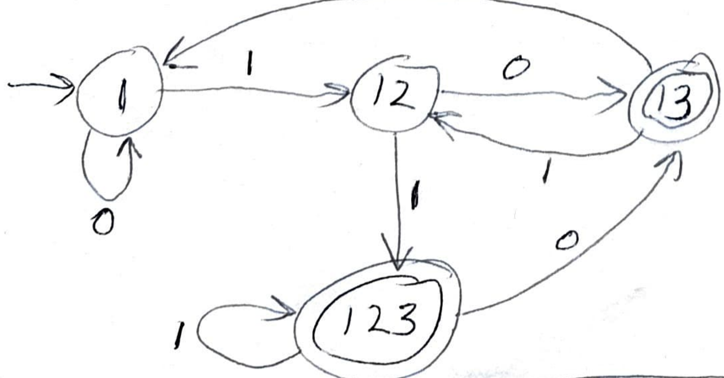
Another example

$L = \{ w \in \{0,1\}^* : \text{2nd last symbol of } w \text{ is } 1 \}$

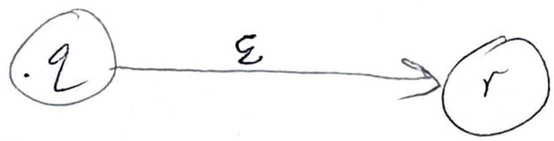
NFA



Equivalent DFA:

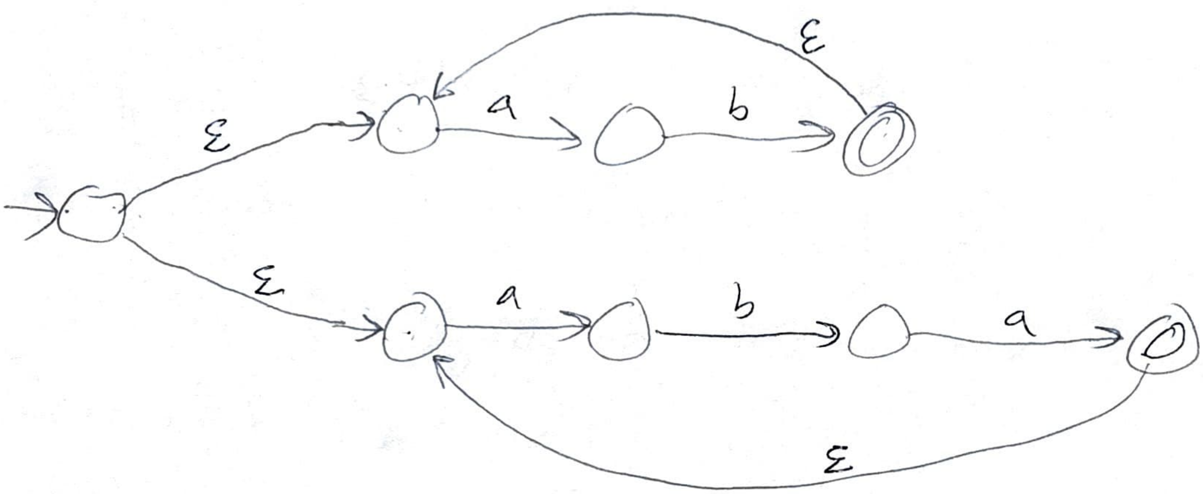


ε-transitions. In an ε-NFA, ~~some~~ transitions are allowed on ε (the empty string);



means can go from state q to state r without reading a symbol

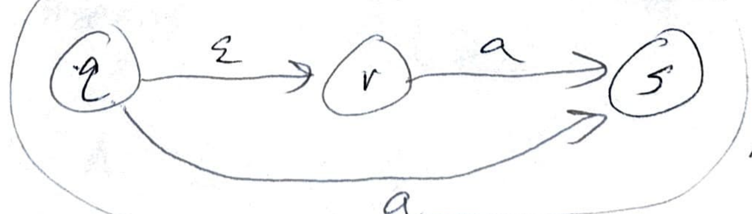
Ex: $L = \underbrace{\{(ab)^n : n \geq 1\}} \cup \{(aba)^n : n \geq 1\}$



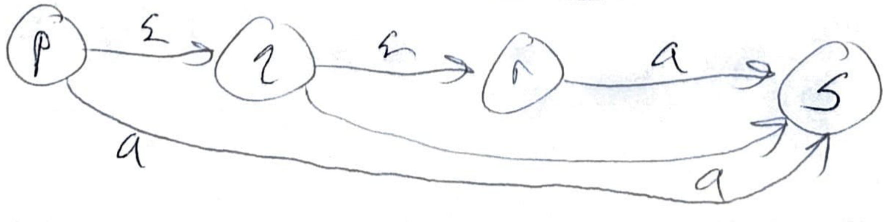
Removing ε-transitions: ε-NFA $\longmapsto$ equivalent NFA

Idea: Add non-ε-transitions that bypass ε-transitions

Ex:



"forward bypass"



or

"back-propagation of accepting state"

~~Algorithm~~

~~Given~~

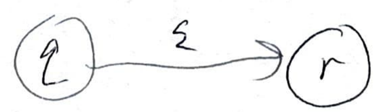
Formal Def of an ε-NFA:

Def: ~~An ε-NF~~ For any alphabet Σ, we let $\Sigma^{?} := \Sigma \cup \{\varepsilon\}$

($\Sigma^{?}$ is the set of strings of length 0 or 1)

An ε-NFA is a 5-tuple $\langle Q, \Sigma, \delta, q_0, F \rangle$ where Q, Σ, q_0, F are the same as with an NFA/DFA and

$$\delta : Q \times \Sigma^{?} \to 2^{Q}.$$



means $r \in \delta(q, \varepsilon)$

Given an ε-NFA $A = \langle Q, \Sigma, \delta, q_0, F \rangle$ and a string $w \in \Sigma^*$, say that <u>A accepts w</u>

if there exist $w_1, w_2, \ldots, w_n \in \Sigma^{?}$
and states $s_0, s_1, \ldots, s_n \in Q$ such that

1) $w = w_1 w_2 \cdots w_n$ (concatenation)
2) $s_0 = q_0$
3) $s_n \in F$
4 $\forall i,\ 1 \leq i \leq n,\ s_i \in \delta(s_{i-1}, w_i)$

ε-NFA $\longrightarrow$ NFA conversion

Given ε-NFA $A = \langle Q, \Sigma, \delta, q_0, F \rangle$
build NFA $B = \langle Q, \Sigma, \delta', q_0, F' \rangle$:

1. $B := A$ ($\delta' := \delta$ and $F' := F$)
2. while $\exists q, r \in Q$, // back-prop of acceptance

   $q \xrightarrow{\varepsilon} r$

   $r \in F'$ and $q \notin F'$
   and $r \in \delta(q, \varepsilon)$, do

   $F' := F' \cup \{q\}$

   end-while

   Nothing here changes the language recognized

3. while $\exists q, r, s \in Q$, // forward bypass
   $s \in \delta'(r, a)$, for some $a \in \Sigma$
   $r \in \delta'(q, \varepsilon)$, and
   $s \notin \delta'(q, a)$ do

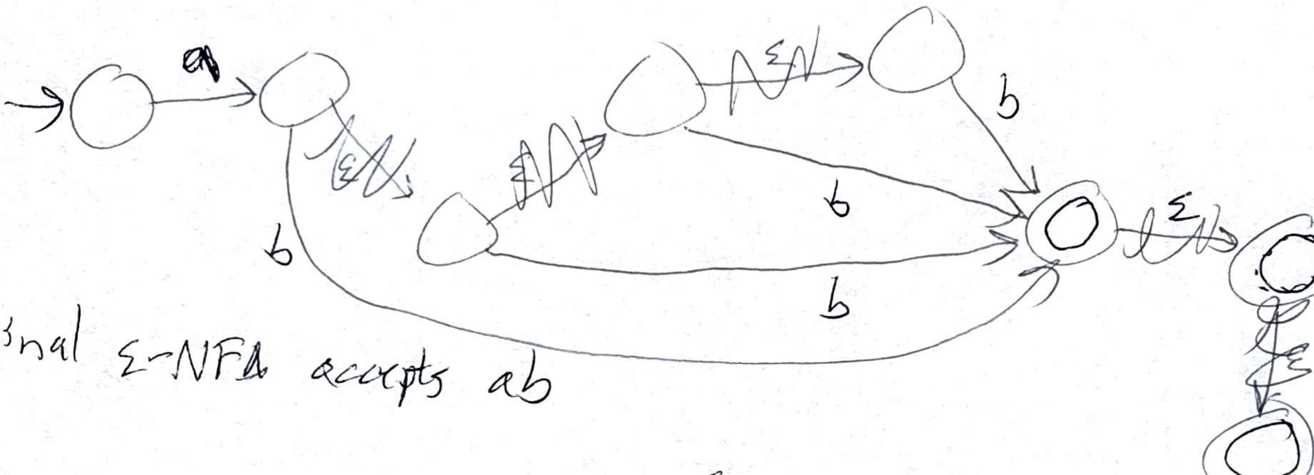
$\delta'(q,a) := \delta'(q,a) \cup \{s\}$

end-while

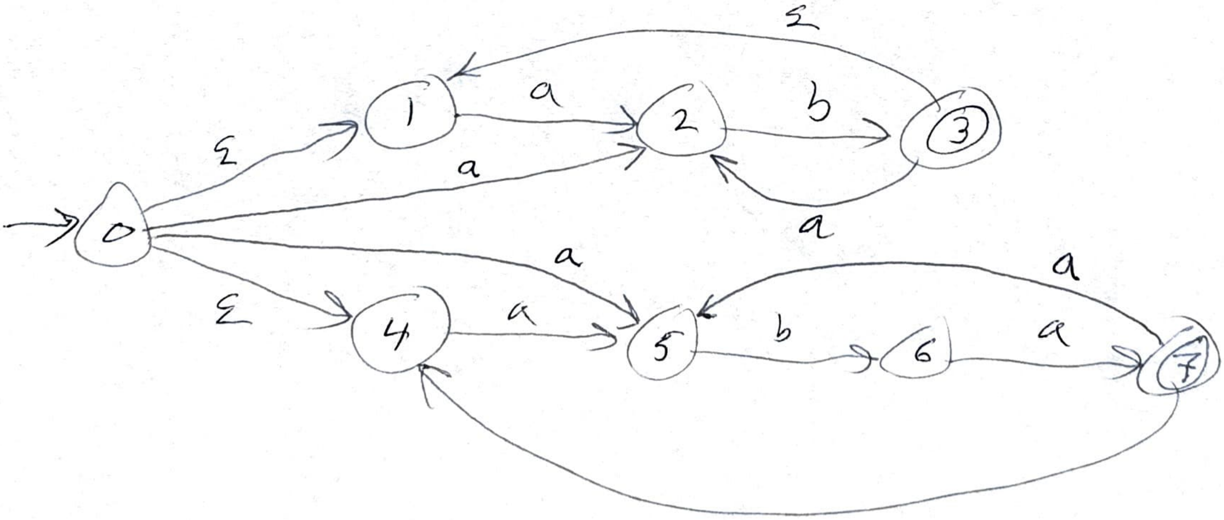
4. //Remove all ε-transitions:

$$\forall q \in Q, \quad \delta'(q,\varepsilon) := \emptyset$$

[Why does step 4 ~~not~~ give the same language?]

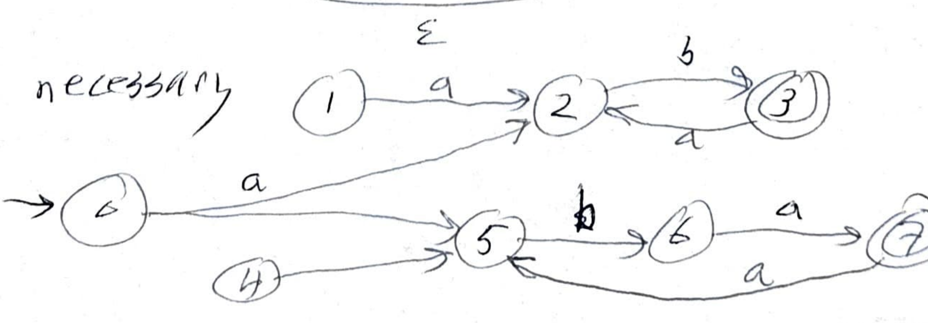


original ε-NFA accepts ab



No back prop necessary

Final:



can remove 1 & 4 (unreachable)