

- Question: How to match regular expressions

quickly? Quickly = one left-to-right pass
linear time

state := _____

for each character c in string

state := $f(c, \text{state})$ ← Free to define f
however we want

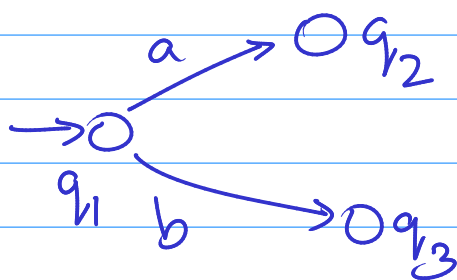
Return $f'(\text{state})$.

Free to define f' .

If the whole algorithm
needs $O(n)$ time,
then f has to
return in $O(1)$ time

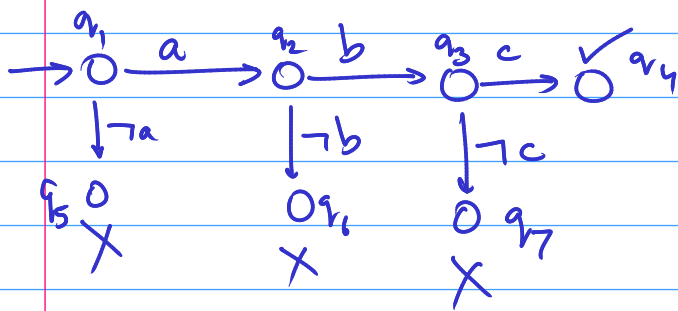
- Lookup tables provided size is bounded

- State transition systems

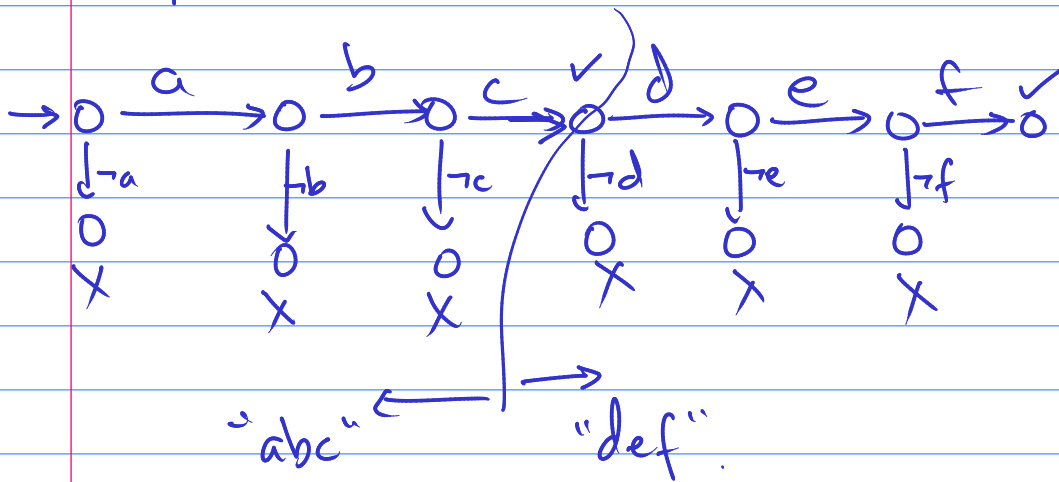


Current State	Char	Next State
q_1	a	q_2
q_1	b	q_3
\vdots	\vdots	\vdots
q_1	a	q_3

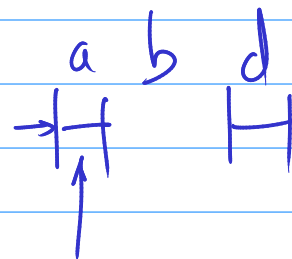
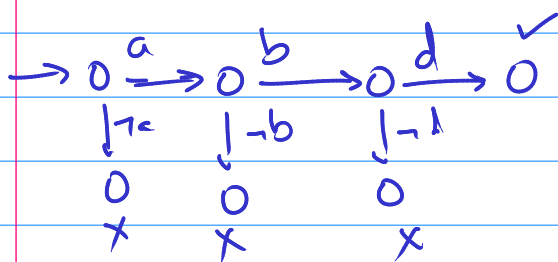
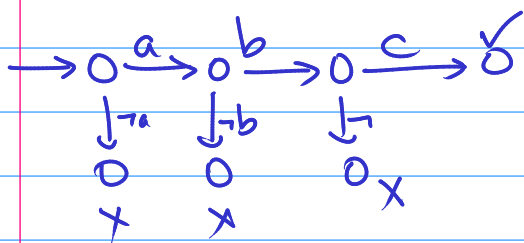
"abc"



"abc" | "def" Concatenation



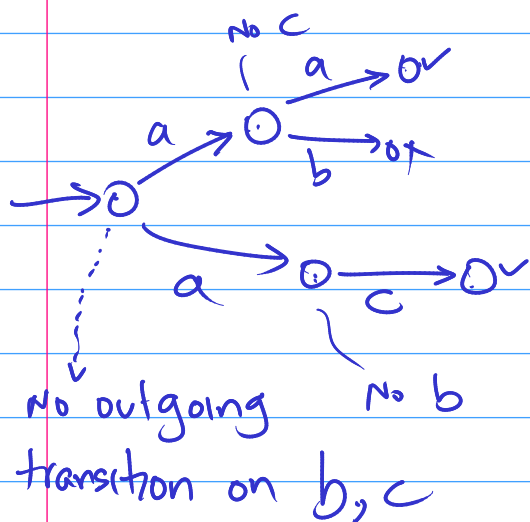
Union "abc" + "abd"



We don't know yet if it will eventually match abc or abd.

Idea: Run them both simultaneously!

(Deterministic) finite automata.
Non-deterministic finite automata.



State transition remains a $O(1)$ operation.

Return true if there exists some run of the machine which leads to an accepting state