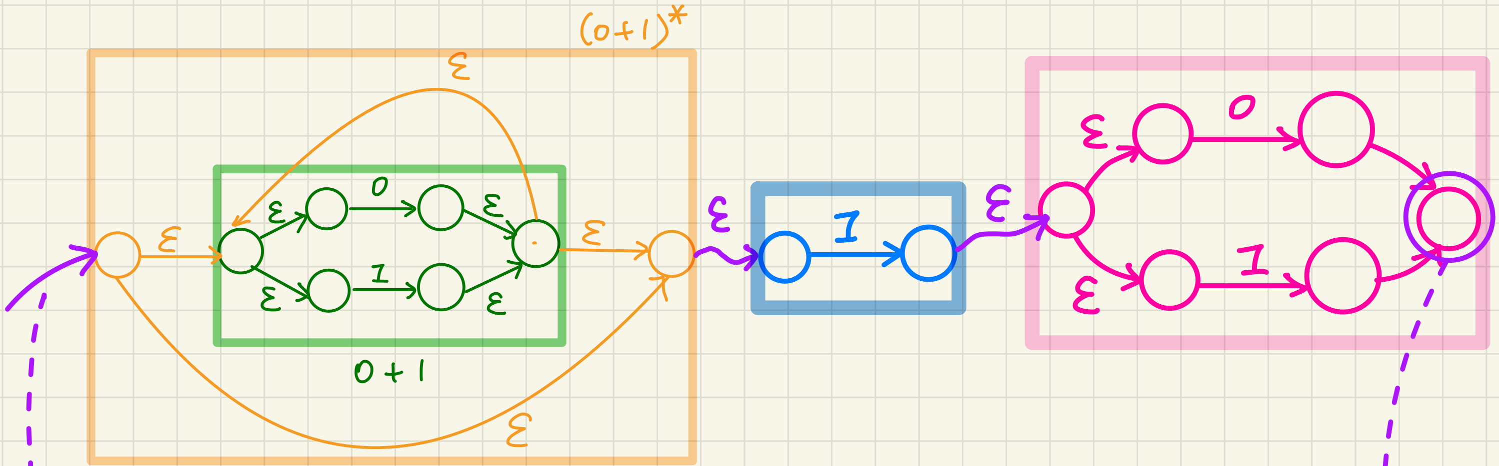
Lecture 10 - June 5

Lexical Analysis

ε-NFA: Formulation, Processing
ε-NFA to DFA: Extended Subset Const.
Minimizing DFA: Introduction

Regular Expression to epsilon-NFA: Example

$(0+1)^*1(0+1)$

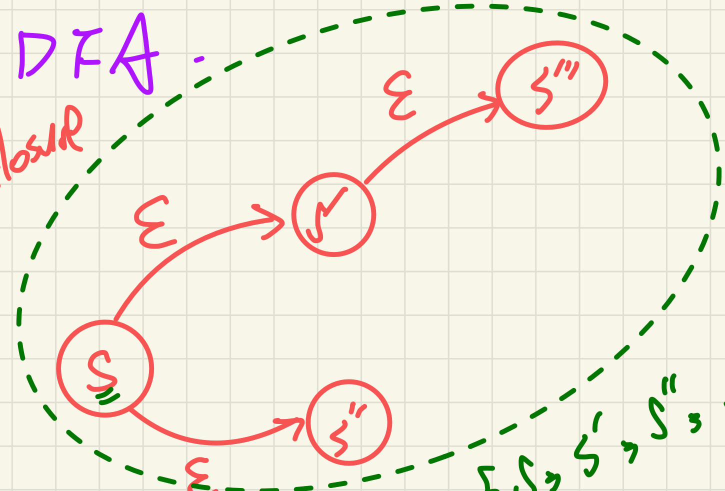


the start state of the left-most ε-NFA pattern is the overall start state

the accept state of the right-most ε-NFA pattern is the overall accept state

$$
\begin{tikzcd}
\boxed{\text{NFA}} \arrow[r] \arrow[d] & \text{DFA} \\
\varepsilon\text{-NFA} \arrow[r, "?"] & \text{DFA}
\end{tikzcd}
$$

epsilon-closure



$\text{Eclose}(s) = \{s, s', s'', s'''\}$

epsilon-NFA: Formulation (2)

An *ϵ-NFA* is a 5-tuple

$$M = (Q,\ \Sigma,\ \delta,\ q_0,\ F)$$

we define the *epsilon closure* (or *ϵ-closure*) as a function

$$\text{ECLOSE} : Q \to \mathbb{P}(Q)$$

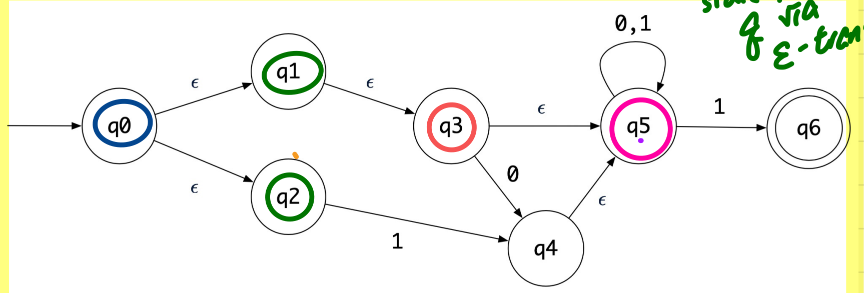
$\text{ECLOSE}(q) \subseteq Q$

For any state $q \in Q$

$q \in Q$

$$\text{ECLOSE}(q) = \{q\} \cup \bigcup_{p \in \delta(q,\epsilon)} \text{ECLOSE}(p)$$

⇒ each immediately reachable state from q via ε-transition.



Derive ECLOSE(q0).

$\text{ECLOSE}(q_0)$

$= \{q_0\} \cup \boxed{\text{ECLOSE}(q_1)} \cup \boxed{\text{ECLOSE}(q_2)}$

$\text{ECLOSE}(q_2)$: $\{q_2\}$

$\text{ECLOSE}(q_1)$: $\{q_1\} \cup \boxed{\text{ECLOSE}(q_3)}$

$\text{ECLOSE}(q_3)$: $\{q_3\} \cup \boxed{\text{ECLOSE}(q_5)}$

$\text{ECLOSE}(q_5)$: $\{q_5\}$

answer: $\{q_0, q_1, q_2, q_3, q_5\}$

epsilon-NFA: Formulation (3)

An ϵ-NFA is a 5-tuple

$$M = (Q,\ \Sigma,\ \delta,\ q_0,\ F)$$

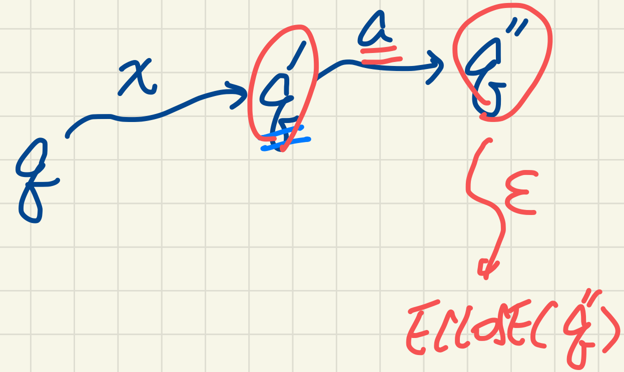
DFA: q
NFA: $\{q\}$

$$\hat{\delta} : (Q \times \Sigma^*) \rightarrow \mathbb{P}(Q)$$

We may define $\hat{\delta}$ recursively, using δ!

$$\boxed{\hat{\delta}(q, \epsilon)} = ECLOSE(q)$$

$$\hat{\delta}(q, xa) = \bigcup \{ ECLOSE(q'') \mid q' \in \hat{\delta}(q, x) \wedge q'' \in \delta(q', a) \}$$

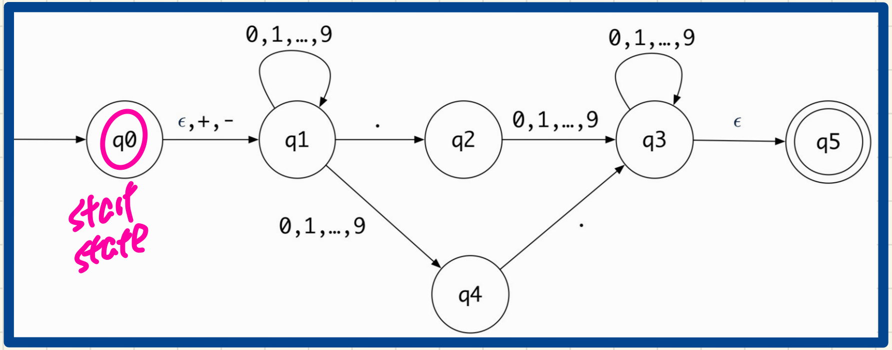


Language of a epsilon-NFA

$$L(M) = \{ w \mid w \in \Sigma^* \wedge \hat{\delta}(q_0, w) \cap F \neq \emptyset \}$$

cond. identical to NFA

epsilon-NFA: Processing Strings



Exercises

① .6

② .

③ +23

How an epsilon-NFA determines if input 5.6 should be processed

$\hat{\delta}(q_0, \epsilon) =$ ECLOSE(q_0) = $\{q_0, q_1\}$

- **Read 5:** $\delta(q_0, 5) \cup \delta(q_1, 5) = \emptyset \cup \{q_1, q_4\} = \{q_1, q_4\}$

$\hat{\delta}(q_0, 5) =$ ECLOSE(q_1) $\cup$ ECLOSE(q_4) $= \{q_1\} \cup \{q_4\} = \{q_1, q_4\}$

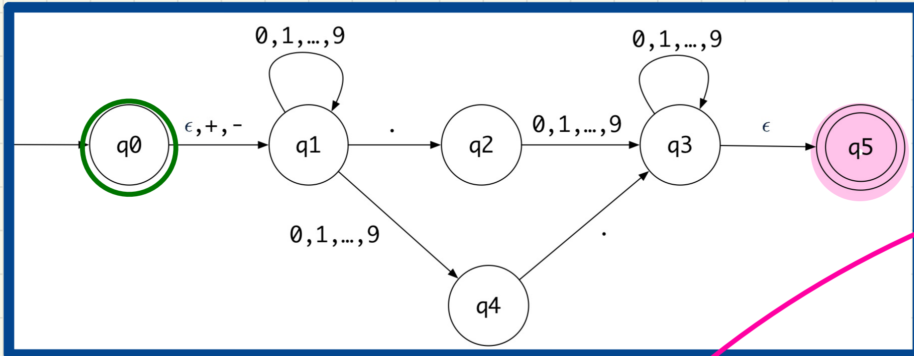
- **Read .:**

$\hat{\delta}(q_0, 5.) =$ Exercise

- **Read 6:**

$\hat{\delta}(q_0, 5.6) =$

epsilon-NFA to DFA: Extended Subset Construction



① $\delta(q_0, d) \cup \delta(q_1, d)$

$= \boxed{S} \rightarrow$ a set

② ECLOSEs of each mem. in the set S

union

subset state

EClose(q_0)

initial state of output DFA

	$d \in 0..9$	$s \in \{+,-\}$	.
$\{q_0, q_1\}$		$\{q_1\}$	$\{q_2\}$
$\{q_1, q_4\}$	$\{q_1, q_4\}$	$\emptyset$	$\{q_2, q_3, q_5\}$
$\{q_1\}$	$\{q_1, q_4\}$	$\emptyset$	$\{q_2\}$
$\{q_2\}$	$\{q_3, q_5\}$	$\emptyset$	$\emptyset$
$\{q_2, q_3, q_5\}$	$\{q_3, q_5\}$	$\emptyset$	$\emptyset$
$\{q_3, q_5\}$	$\{q_3, q_5\}$	$\emptyset$	$\emptyset$

accept states of DFA.

Minimizing DFA: Algorithm

① What if $M' = M$ ⇒ no optimization was necessary

② What if $\boxed{|Q_{M'}| > |Q_M|}$?

should not be possible!

```
ALGORITHM: MinimizeDFAStates
  INPUT: DFA M = (Q, Σ, δ, q0, F)
  OUTPUT: M' s.t. minimum |Q| and equivalent behaviour as M
PROCEDURE:
  P := ∅ /* refined partition so far */
  T := { F, Q − F } /* last refined partition */
  while (P ≠ T):
    P := T
    T := ∅
    for(p ∈ P):
      find the maximal S ⊂ p s.t. splittable(p, S)
      if S ≠ ∅ then
        T := T ∪ {S, p − S}
      else
        T := T ∪ {p}
    end
```

accept states

non-accept states

as soon as $P = T$ ⇒ no further optimization can be done.

given a set P of states as a partition, can we find a proper subset of P that can be split from the rest of P.

splittable(p, S) holds iff there is $c \in \Sigma$ s.t.

1. $S \subset p$ (or equivalently: $p - S \neq \varnothing$)
2. Transitions via c lead all $s \in S$ to states in ***same partition*** $p1$ $(p1 \neq p)$.