CSCE 355
9/6/22

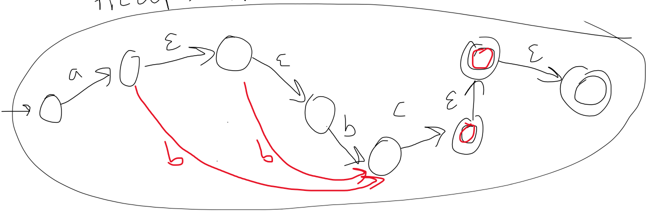
1. ε-NFA $\longrightarrow$ NFA
   A $\longmapsto$ equiv B

2. DFA minimization

ε-NFA $\rightarrow$ NFA idea:

<u>bypass</u> ε-moves
by adding
- more accepting states
- more non-ε-transitions

Accepts abc

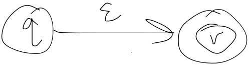


Given an ε-NFA $N = \langle Q, \Sigma, \delta, q_0, F \rangle$

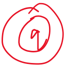
as input:

<u>Phase 1</u>: <u>while</u> there exist

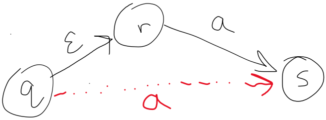
states $q, r \in Q$ s.t. $r \in F$, $q \notin F$, and $r \in \delta(q, \varepsilon)$, do



Put q into F



Phase 2: While there exists an $a \in \Sigma$ and states $q, r, s \in Q$ such that



$s \in \delta(r, a)$, $r \in \delta(q, \varepsilon)$, and $s \notin \delta(q, a)$, do

Put 's into $\delta(q, a)$.

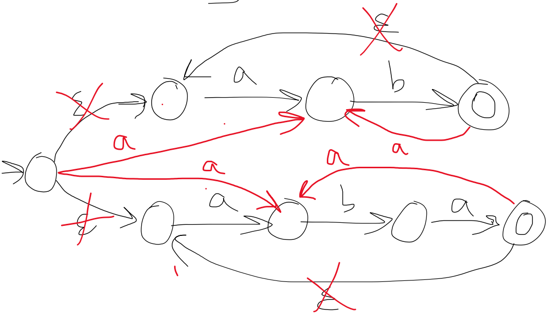
Phase 3: // remove all ε-transitions

For every $q \in Q$,

Set $\delta(q, \varepsilon) := \emptyset$

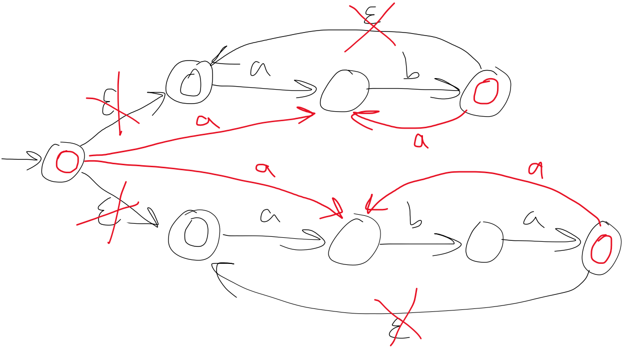
Example: $L = \{ w \in \{a,b\}^* :$

w is either ab repeated ~~one~~ zero or more times or aba repeated ~~one~~ zero or more times $\}$



Phase 1 — nothing

Phase 2 — added 4 transitions



DFA minimization

Find the min-state DFA

equivalent to a given DFA.

Given DFA $A = \langle Q, \Sigma, \delta, q_0, F \rangle$

1) Remove any state of A that is *not reachable* from the start state

Result: DFA where,

$$\forall q \in Q \ \exists w \in \Sigma^*,$$
$$q = \hat{\delta}(q_0, w)$$

such a DFA we will call "*sane*"

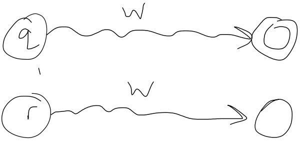
2) Merge states that are *indistinguishable* from each other.

Def: Let q, r be states of A (sane). A string w *distinguishes q from r* if one of

$$\hat{\delta}(q, w) \text{ and } \hat{\delta}(r, w)$$

is accepting and the other

rejecting.



or vice versa

q and r are distinguishable if there exists a string w that distinguishes them.

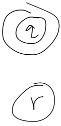
Step 2 in more detail:

- Find all pairs of states that are distinguishable, then merge any pairs left into one or more groups of mutually indistinguishable states

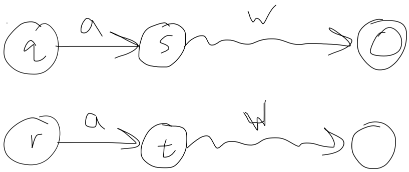
```
DistPairs(A)  // A is a sane DFA

  1) Initialize a table $T[Q,Q]$
     (2-dim where rows & cols indexed by states):
     For all states $q, r \in Q$
```

```
Set $T[q,r] := \text{blank}$
2) $\forall q, r \in Q$, if one of
   q and r is accepting
   and the other rejecting,
   then
       $\underline{T[q,r] := T[r,q] := \text{'X'}}$
   // mark (q,r) as a
   // distinguishable pair
```



```
   // distinguished by $\varepsilon$
3) $\underline{\text{While}}$ there exist states
   q and r such that
       $T[q,r] == \text{blank}$
   and $\exists a \in \Sigma$ such that
   $T[\underbrace{\delta(q,a)}_{s}, \underbrace{\delta(r,a)}_{t}] == \text{'X'}$
   $\underline{\text{do}}$
       $T[q,r] := T[r,q] := \text{'X'}$
end-while
```

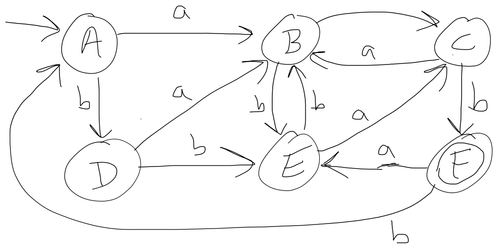


If w distinguishes s from t, then aw distinguishes q from r

Can use string-inductive proof that only set T[q,r] to X if q,r are distinguishable.
[Algorithm is sound]

Theorem: After the algorithm finishes, T[q,r] == X for all distinguishable pairs of states (q,r).
[Algorithm is complete]

Ex:



	A	B	C	D	E
B					
C	X	X			
D			X		
E			X		
F	X	X	X	X	X