

Computational Linguistics

Lecture 2 – Finite State Automata

Dietrich Klakow & Stefan Thater
FR 4.7 Allgemeine Linguistik (Computerlinguistik)
Universität des Saarlandes

Summer 2013

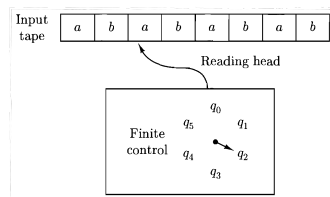
Some basic definitions

- **An alphabet Σ** is a finite set of symbols
- **A string over Σ** is a sequence of symbols from Σ
 - ϵ stands for the empty string
- **The length $|w|$** is the number of symbols in w
- Σ^* denotes this set of all strings over Σ
- **A (formal) language** is a subset of Σ^* for some alphabet Σ

2

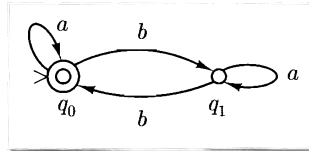
Deterministic Finite Automata

- **$M = (K, \Sigma, \delta, s, F)$**
 - K is a finite set of states
 - Σ is an input alphabet
 - δ is a transition function
 - $s \in K$ is the start state
 - $F \subseteq K$ is the set of final (accepting) states
- **Transition function**
 - $\delta(q, a) = q'$
 - when M is in state q and reads input a , it goes into state q'



3

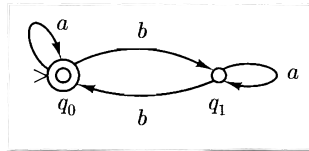
Automata as Graphs



- Nodes = states
- Edges = transition function
 - an edge from state q to state q' labeled by $a \Leftrightarrow \delta(q, a) = q'$
- \rightarrow marks the start state
- Double circles = final states

4

Automata as Graphs



- $M = \langle K, \Sigma, \delta, s, F \rangle$
 - $K = \{q_0, q_1\}$
 - $\Sigma = \{a, b\}$
 - $s = q_0$
 - $F = \{q_0\}$
 - $\delta(q_0, a) = q_0$
 - $\delta(q_0, b) = q_1$
 - $\delta(q_1, a) = q_0$
 - $\delta(q_1, b) = q_1$

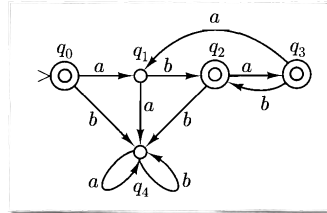
5

More definitions

- **A configuration** is a pair $\langle q, w \rangle \in K \times \Sigma^*$
 - q = the current state
 - w = the unread part of the string being processed
- **Yields in one step**
 - $\langle q, w \rangle \vdash_M \langle q', w' \rangle$
 - iff $w = aw'$ for some $a \in \Sigma$, $w' \in \Sigma^*$ and $\delta(q, a) = q'$
- **Yields**
 - \vdash_M^* is the reflexive, transitive closure of \vdash_M
- **The language accepted** by a DFA $M = \langle K, \Sigma, \delta, s, F \rangle$
 - $L(M) = \{ w \mid \langle s, w \rangle \vdash_M^* \langle q, \epsilon \rangle \text{ for some } q \in F \}$

6

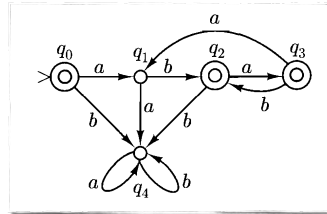
An Example



7

An Example

$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$

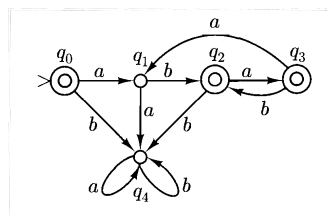


7

An Example

$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$

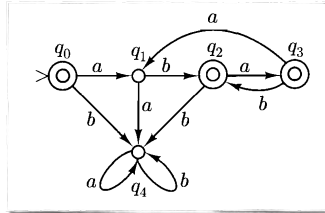
$\vdash_M \langle q_2, aba \rangle$



7

An Example

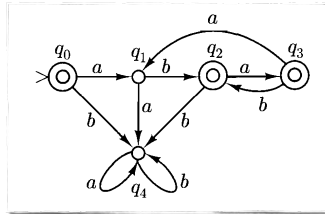
$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$
 $\vdash_M \langle q_2, aba \rangle$
 $\vdash_M \langle q_3, ba \rangle$



7

An Example

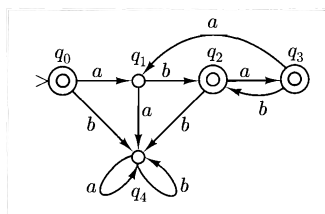
$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$
 $\vdash_M \langle q_2, aba \rangle$
 $\vdash_M \langle q_3, ba \rangle$
 $\vdash_M \langle q_2, a \rangle$



7

An Example

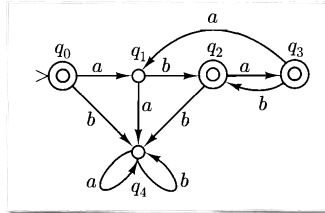
$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$
 $\vdash_M \langle q_2, aba \rangle$
 $\vdash_M \langle q_3, ba \rangle$
 $\vdash_M \langle q_2, a \rangle$
 $\vdash_M \langle q_3, \epsilon \rangle$



7

An Example

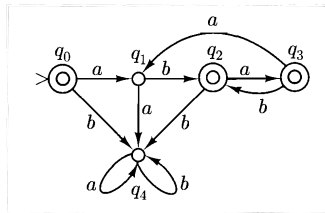
$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$
 $\vdash_M \langle q_2, aba \rangle$
 $\vdash_M \langle q_3, ba \rangle$
 $\vdash_M \langle q_2, a \rangle$
 $\vdash_M \langle q_3, \varepsilon \rangle$
 $\Rightarrow ababa \in L(M)$



7

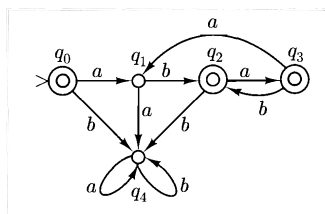
An Example

$\langle q_0, ababa \rangle \vdash_M \langle q_1, baba \rangle$
 $\vdash_M \langle q_2, aba \rangle$
 $\vdash_M \langle q_3, ba \rangle$
 $\vdash_M \langle q_2, a \rangle$
 $\vdash_M \langle q_3, \varepsilon \rangle$
 $\Rightarrow ababa \in L(M)$



7

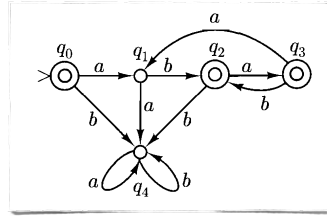
An Example



8

An Example

$\langle q_0, abaa \rangle \vdash_M \langle q_1, baa \rangle$

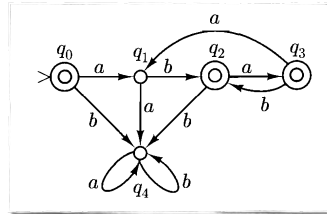


8

An Example

$\langle q_0, abaa \rangle \vdash_M \langle q_1, baa \rangle$

$\vdash_M \langle q_2, ba \rangle$



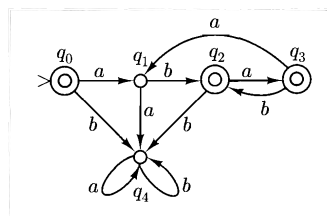
8

An Example

$\langle q_0, abaa \rangle \vdash_M \langle q_1, baa \rangle$

$\vdash_M \langle q_2, ba \rangle$

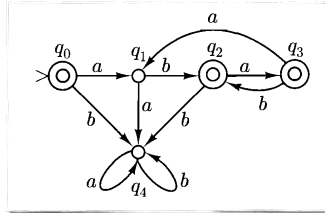
$\vdash_M \langle q_3, a \rangle$



8

An Example

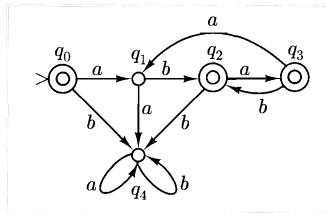
$\langle q_0, abaa \rangle \vdash_M \langle q_1, baa \rangle$
 $\vdash_M \langle q_2, ba \rangle$
 $\vdash_M \langle q_3, a \rangle$
 $\vdash_M \langle q_1, \varepsilon \rangle$



8

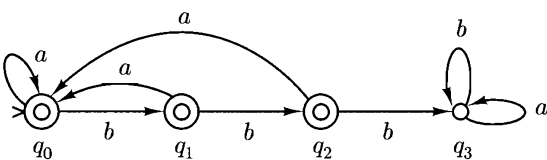
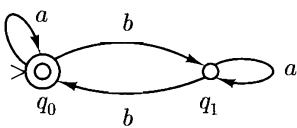
An Example

$\langle q_0, abaa \rangle \vdash_M \langle q_1, baa \rangle$
 $\vdash_M \langle q_2, ba \rangle$
 $\vdash_M \langle q_3, a \rangle$
 $\vdash_M \langle q_1, \varepsilon \rangle$
 $\Rightarrow abaa \notin L(M)$



8

Exercise: $L(M) = ?$



9

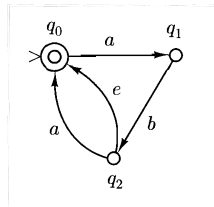
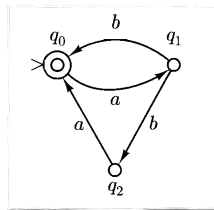
Recognition Algorithm

```
function RECOGNIZE(DFA M, STRING input)
  index ← 0
  state ← start state of M
  while index < length(input) do
    state ← trans[state, input[index]]
    index ← index + 1
  end
  if state is a final state of M
  then return accept
  else return reject
end
```

10

Nondeterministic Automata

- Nondeterministic finite automata:
 - several symbols can be read at once, or none at all
 - several possible next states



11

Nondeterministic Automata

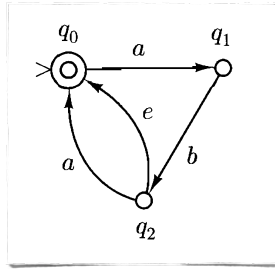
- $M = (K, \Sigma, \Delta, s, F)$
 - K is a finite set of states
 - Σ is an input alphabet
 - $\Delta \subseteq K \times \Sigma^* \times K$ is a finite transition relation
 - $s \in K$ is the start state
 - $F \subseteq K$ is the set of final (accepting) states
- **Transition relation** $\Delta \subseteq K \times \Sigma^* \times K$
 - $(q, w, q') \in \Delta$ = when the automaton is in state q and reads input w , it can go into state q'
 - Note: here we restrict ourselves to NFA where $|w| \leq 1$

12

An Example

- $M = \langle K, \Sigma, \Delta, s, F \rangle$

- $K = \{q_0, q_1, q_2\}$
- $\Sigma = \{a, b\}$
- $s = q_0$
- $F = \{q_0\}$
- $\Delta = \{(q_0, a, q_1), (q_1, b, q_2), (q_2, a, q_0), (q_2, \epsilon, q_0)\}$



13

Configurations

- **Configurations**

- are elements from $K \times \Sigma^*$ (as before)

- **Yields in one step**

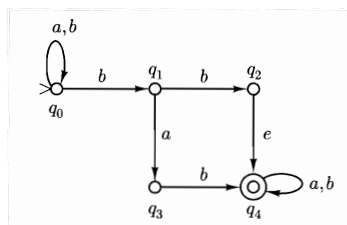
- $\langle q, w \rangle \vdash_M \langle q', w' \rangle$
- iff $w = uw'$ for some $u, w \in \Sigma^*$ and $\langle q, u, q' \rangle \in \Delta$

- **The language accepted** by an NFA

- $L(M) = \{ w \mid \langle s, w \rangle \vdash_M^* \langle q, \epsilon \rangle \text{ for some } q \in F \}$

14

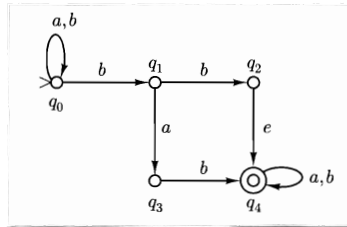
An Example



$\langle q_0, babba \rangle$

15

An Example

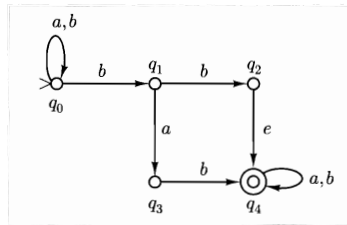


$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

15

An Example



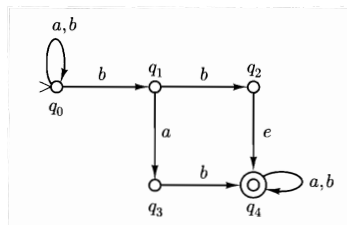
$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

15

An Example



$\langle q_0, babba \rangle$

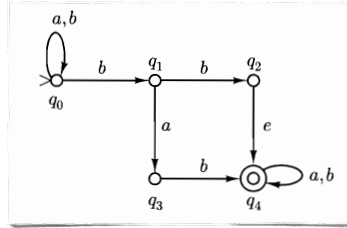
$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

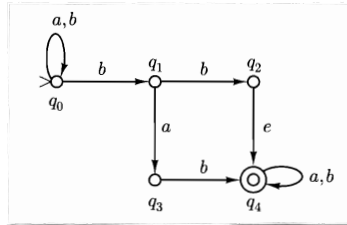
15

An Example



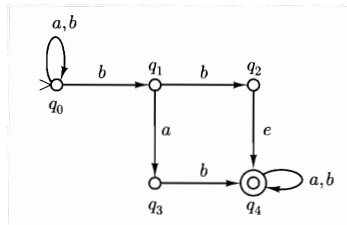
- $\langle q_0, babba \rangle$
- $\vdash_M \langle q_0, abba \rangle$
- $\vdash_M \langle q_0, bba \rangle$
- $\vdash_M \langle q_0, ba \rangle$
- $\vdash_M \langle q_1, a \rangle$

An Example



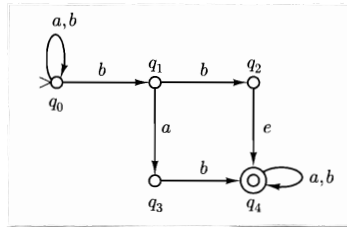
- $\langle q_0, babba \rangle$
- $\vdash_M \langle q_0, abba \rangle$
- $\vdash_M \langle q_0, bba \rangle$
- $\vdash_M \langle q_0, ba \rangle$
- $\vdash_M \langle q_1, a \rangle$
- $\vdash_M \langle q_3, \epsilon \rangle$

An Example



- $\langle q_0, babba \rangle$
 - $\vdash_M \langle q_0, abba \rangle$
 - $\vdash_M \langle q_0, bba \rangle$
 - $\vdash_M \langle q_0, ba \rangle$
 - $\vdash_M \langle q_1, a \rangle$
 - $\vdash_M \langle q_3, \epsilon \rangle$
- $\langle q_0, babba \rangle$

An Example



$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

$\vdash_M \langle q_1, a \rangle$

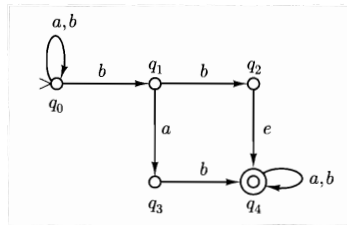
$\vdash_M \langle q_3, \varepsilon \rangle$

$\langle q_0, babba \rangle$

$\vdash_M \langle q_1, abba \rangle$

15

An Example



$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

$\vdash_M \langle q_1, a \rangle$

$\vdash_M \langle q_3, \varepsilon \rangle$

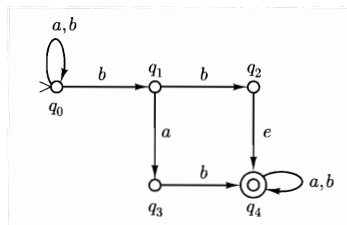
$\langle q_0, babba \rangle$

$\vdash_M \langle q_1, abba \rangle$

$\vdash_M \langle q_3, bba \rangle$

15

An Example



$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

$\vdash_M \langle q_1, a \rangle$

$\vdash_M \langle q_3, \varepsilon \rangle$

$\langle q_0, babba \rangle$

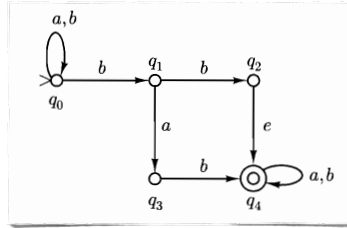
$\vdash_M \langle q_1, abba \rangle$

$\vdash_M \langle q_3, bba \rangle$

$\vdash_M \langle q_4, ba \rangle$

15

An Example



$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

$\vdash_M \langle q_1, a \rangle$

$\vdash_M \langle q_3, \epsilon \rangle$

$\langle q_0, babba \rangle$

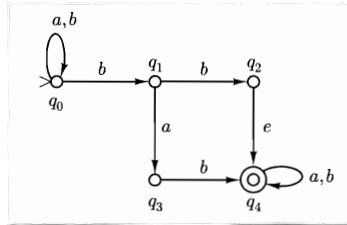
$\vdash_M \langle q_1, abba \rangle$

$\vdash_M \langle q_3, bba \rangle$

$\vdash_M \langle q_4, ba \rangle$

$\vdash_M \langle q_4, a \rangle$

An Example



$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

$\vdash_M \langle q_1, a \rangle$

$\vdash_M \langle q_3, \epsilon \rangle$

$\langle q_0, babba \rangle$

$\vdash_M \langle q_1, abba \rangle$

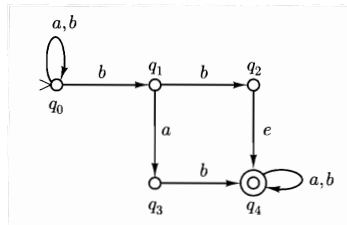
$\vdash_M \langle q_3, bba \rangle$

$\vdash_M \langle q_4, ba \rangle$

$\vdash_M \langle q_4, a \rangle$

$\vdash_M \langle q_4, \epsilon \rangle$

An Example



$\langle q_0, babba \rangle$

$\vdash_M \langle q_0, abba \rangle$

$\vdash_M \langle q_0, bba \rangle$

$\vdash_M \langle q_0, ba \rangle$

$\vdash_M \langle q_1, a \rangle$

$\vdash_M \langle q_3, \epsilon \rangle$

$\langle q_0, babba \rangle$

$\vdash_M \langle q_1, abba \rangle$

$\vdash_M \langle q_3, bba \rangle$

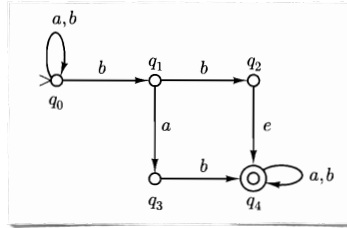
$\vdash_M \langle q_4, ba \rangle$

$\vdash_M \langle q_4, a \rangle$

$\vdash_M \langle q_4, \epsilon \rangle$

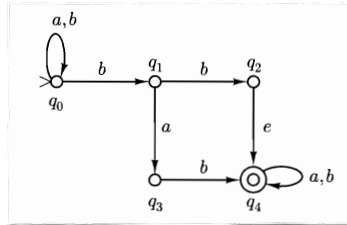
$\langle q_0, babba \rangle$

An Example



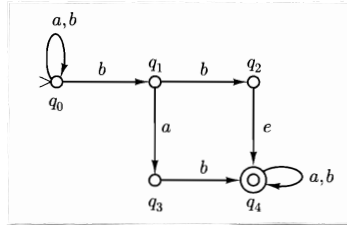
$\langle q_0, babba \rangle$	$\langle q_0, babba \rangle$	$\langle q_0, babba \rangle$
$\vdash_M \langle q_0, abba \rangle$	$\vdash_M \langle q_1, abba \rangle$	$\vdash_M \langle q_0, abba \rangle$
$\vdash_M \langle q_0, bba \rangle$	$\vdash_M \langle q_3, bba \rangle$	
$\vdash_M \langle q_0, ba \rangle$	$\vdash_M \langle q_4, ba \rangle$	
$\vdash_M \langle q_1, a \rangle$	$\vdash_M \langle q_4, a \rangle$	
$\vdash_M \langle q_3, \epsilon \rangle$	$\vdash_M \langle q_4, \epsilon \rangle$	

An Example



$\langle q_0, babba \rangle$	$\langle q_0, babba \rangle$	$\langle q_0, babba \rangle$
$\vdash_M \langle q_0, abba \rangle$	$\vdash_M \langle q_1, abba \rangle$	$\vdash_M \langle q_0, abba \rangle$
$\vdash_M \langle q_0, bba \rangle$	$\vdash_M \langle q_3, bba \rangle$	$\vdash_M \langle q_0, bba \rangle$
$\vdash_M \langle q_0, ba \rangle$	$\vdash_M \langle q_4, ba \rangle$	
$\vdash_M \langle q_1, a \rangle$	$\vdash_M \langle q_4, a \rangle$	
$\vdash_M \langle q_3, \epsilon \rangle$	$\vdash_M \langle q_4, \epsilon \rangle$	

An Example



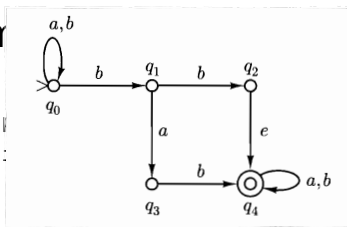
$\langle q_0, babba \rangle$	$\langle q_0, babba \rangle$	$\langle q_0, babba \rangle$
$\vdash_M \langle q_0, abba \rangle$	$\vdash_M \langle q_1, abba \rangle$	$\vdash_M \langle q_0, abba \rangle$
$\vdash_M \langle q_0, bba \rangle$	$\vdash_M \langle q_3, bba \rangle$	$\vdash_M \langle q_0, bba \rangle$
$\vdash_M \langle q_0, ba \rangle$	$\vdash_M \langle q_4, ba \rangle$	$\vdash_M \langle q_1, ba \rangle$
$\vdash_M \langle q_1, a \rangle$	$\vdash_M \langle q_4, a \rangle$	
$\vdash_M \langle q_3, \epsilon \rangle$	$\vdash_M \langle q_4, \epsilon \rangle$	

Recognition Algorithm

```

function RECOGNIZE(NFA M, STRING input)
  agenda = list of configurations, initially containing only
           the configuration (start state of M, input)
  while agenda is not empty do
    conf ← POP(agenda)
    if conf is an accepting configuration then
      return accept
    else
      for all conf' such that conf ⊢ conf' do
        PUSH(agenda, conf')
      end
    end
  return reject
end
  
```

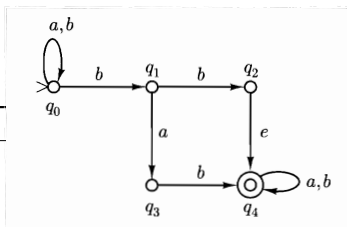
Recognition Algorithm



```

function RECOGNIZE(NFA M, STRING input)
  agenda = list of configurations, initially containing only
           the configuration (start state of M, input)
  while agenda is not empty do
    conf ← POP(agenda)
    if conf is an accepting configuration then
      return accept
    else
      for all conf' such that conf ⊢ conf' do
        PUSH(agenda, conf')
      end
    end
  return reject
end
  
```

An Example



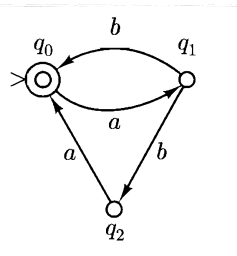
conf	agenda
-	(q ₀ , babba)
(q ₀ , babba)	(q₀, abba), (q₁, abba)
(q ₀ , abba)	(q₀, bba), (q₁, abba)
(q ₀ , bba)	(q₀, ba), (q₁, ba), (q₁, abba)
(q ₀ , ba)	(q₀, a), (q₁, a), (q₁, ba), (q₁, abba)
(q ₀ , a)	(q₀, ε), (q₁, a), (q₁, ba), (q₁, abba)
(q ₀ , ε)	(q ₁ , a), (q ₁ , ba), (q ₁ , abba)
(q ₁ , a)	(q₃, ε), (q₁, ba), (q₁, abba)
(q ₃ , ε)	(q ₁ , ba), (q ₁ , abba)
(q ₁ , ba)	(q₂, a), (q₁, abba)
(q ₂ , a)	(q₄, a), (q₁, abba)
(q ₄ , a)	(q₄, ε), (q₁, abba)
(q ₄ , ε)	(q ₁ , abba)

NFA = DFA (preliminary)

- **Theorem:** for every NFA $M = \langle K, \Sigma, \Delta, s, F \rangle$ there is an equivalent DFA $M' = \langle K', \Sigma, \delta, s', F' \rangle$ such that $L(M) = L(M')$
- Let us first consider the special case where for all elements $(q, w, q') \in \Delta$, w is a string of length 1
- We construct the DFA $M' = \langle K', \Sigma, \delta, s', F' \rangle$ as follows:
 - $K' = 2^K$
 - $s' = \{s\}$
 - $\delta(Q, a) = \{ k \in K \mid (q, a, k) \in \Delta \text{ for some } q \in Q \}$
 - for all $Q \subseteq K$
 - $F' = \{ Q \subseteq K \mid Q \cap F \neq \emptyset \}$

18

NFA = DFA (preliminary)

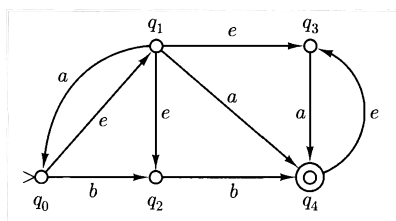


- **Theorem:** for every NFA $M = \langle K, \Sigma, \Delta, s, F \rangle$ there is an equivalent DFA $M' = \langle K', \Sigma, \delta, s', F' \rangle$ such that $L(M) = L(M')$
- Let us first consider the special case elements $(q, w, q') \in \Delta$, w is a string of length 1
- We construct the DFA $M' = \langle K', \Sigma, \delta, s', F' \rangle$ as follows:
 - $K' = 2^K$
 - $s' = \{s\}$
 - $\delta(Q, a) = \{ k \in K \mid (q, a, k) \in \Delta \text{ for some } q \in Q \}$
 - for all $Q \subseteq K$
 - $F' = \{ Q \subseteq K \mid Q \cap F \neq \emptyset \}$

18

ϵ -Closure

- **ϵ -Closure**
 - $E(q) = \{ k \mid (q, \epsilon) \vdash^* (k, \epsilon) \}$
- Examples:
 - $E(q_0) = \{ q_0, q_1, q_2, q_3 \}$
 - $E(q_1) = \{ q_1, q_2, q_3 \}$
 - $E(q_2) = \{ q_2 \}$
- Note:
 - For all $q, q' \in E(q)$



19

NFA = DFA

- **Theorem:** for every NFA $M = \langle K, \Sigma, \Delta, s, F \rangle$ there is an equivalent DFA M' such that $L(M) = L(M')$
- We construct the DFA $M' = \langle K', \Sigma, \delta', s', F' \rangle$ as follows:
 - $K' = 2^K$
 - $s' = E(s)$
 - $\delta'(Q, a) = \cup \{ E(k) \in K' \mid \langle q, a, k \rangle \in \Delta \text{ for some } q \in Q \}$,
 - for all $Q \subseteq K$
 - $F' = \{ Q \subseteq K \mid Q \cap F \neq \emptyset \}$

20

NFA = DFA

- **Lemma:** For any string $w \in \Sigma^*$ and any states $p, q \in K'$:
 - $\langle q, w \rangle \vdash_M^* \langle p, \epsilon \rangle$ iff $\langle E(q), w \rangle \vdash_{M'}^* \langle P, \epsilon \rangle$
for some P containing p
- Using this lemma, it is easy to show that $L(M) = L(M')$
 - $w \in L(M)$
 - iff $\langle s, w \rangle \vdash_M^* \langle f, \epsilon \rangle$, for some $f \in F$
 - iff $\langle E(s), w \rangle \vdash_{M'}^* \langle Q, \epsilon \rangle$, for some Q containing f
 - iff $\langle E(s), w \rangle \vdash_{M'}^* \langle Q, \epsilon \rangle$, for some $Q \in F'$
 - iff $w \in L(M')$

21

Subset construction algorithm

- **ϵ -closure(s)**
returns the set of NFA states reachable from state s using ϵ -transitions
- **ϵ -closure(T)**
returns the set of NFA states reachable from some s in T using ϵ -transition
- **move(T, a)**
returns the set of NFA states to which there is transition for input $a \in \Sigma$ from some state $s \in T$

22

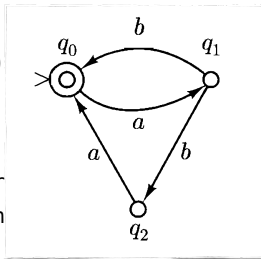
Subset construction algorithm

```
function DFA(K,  $\Sigma$ ,  $\Delta$ , s, F)
  K'  $\leftarrow$  list that contains only  $\epsilon$ -closure(s), unmarked
  while there is an unmarked state T in K' do
    mark T
    for each symbol a  $\in$   $\Sigma$  do
      U  $\leftarrow$   $\epsilon$ -closure(move(T, a))
      if U  $\notin$  K' then
        add U as an unmarked state to K'
         $\delta$ [T, a]  $\leftarrow$  U
      end
    end
  end
  return <the corresponding DFA>
end
```

23

Subset construction algo

```
function DFA(K,  $\Sigma$ ,  $\Delta$ , s, F)
  K'  $\leftarrow$  list that contains only  $\epsilon$ -closur
  while there is an unmarked state T in
    mark T
    for each symbol a  $\in$   $\Sigma$  do
      U  $\leftarrow$   $\epsilon$ -closure(move(T, a))
      if U  $\notin$  K' then
        add U as an unmarked state to K'
         $\delta$ [T, a]  $\leftarrow$  U
      end
    end
  end
  return <the corresponding DFA>
end
```



23

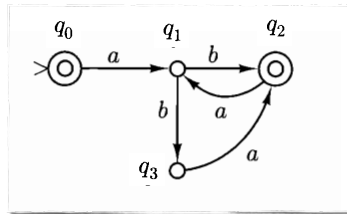
Literature

- Jurafsky and Martin (2009). Chapter 2.
- Lewis and Papadimitriou (1981). Elements of the theory of computation. Chapter 2.

24

Exercise 1

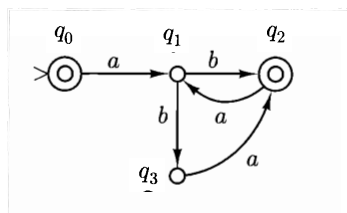
- Apply (using pen and paper) the recognition algorithm on slide 16 for the nondeterministic automaton shown below to the input string “abaaba”
- There is a problem with this algorithm. Which one? How can the algorithm be improved?



25

Exercise 2

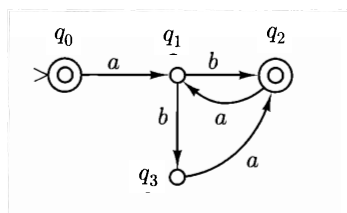
- Construct a deterministic automaton for the nondeterministic automaton shown below, using the subset construction algorithm on slide 23.



26

Exercise 3

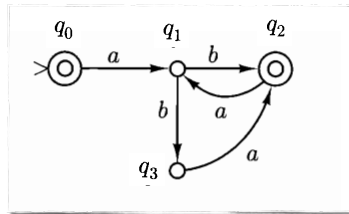
- Implement the recognition algorithm for NFA on slide 16.
- Your submission should use the automaton shown below and the following inputs as test case.
 - $ab \in L(M)$
 - $aba \in L(M)$
 - $abaab \in L(M)$
 - $abba \notin L(M)$
 - $aabab \notin L(M)$
 - More test cases are welcome!



27

Exercise 4

- Implement the subset construction algorithm on slide 23.
- Your submission should use the automaton show below as a test case.



28

Exercises - Remarks

- Submit the source code by email to me (stth@...)
- The source code should contain a comment that tells how the code can be used.

29