

# Endliche Automaten

---

## Übersicht

- ◆ Einführung
- ◆ Beispiel: Der Lachautomat
- ◆ Deterministische vs. Nicht-Deterministische Endliche Automaten
- ◆ Bestandteile, Begriffe
- ◆ Deterministische Endliche Automaten (DEA) in Prolog
- ◆ Literatur

## Zweck

- ◆ Einfache Automaten entwerfen
- ◆ DEA in Prolog programmieren

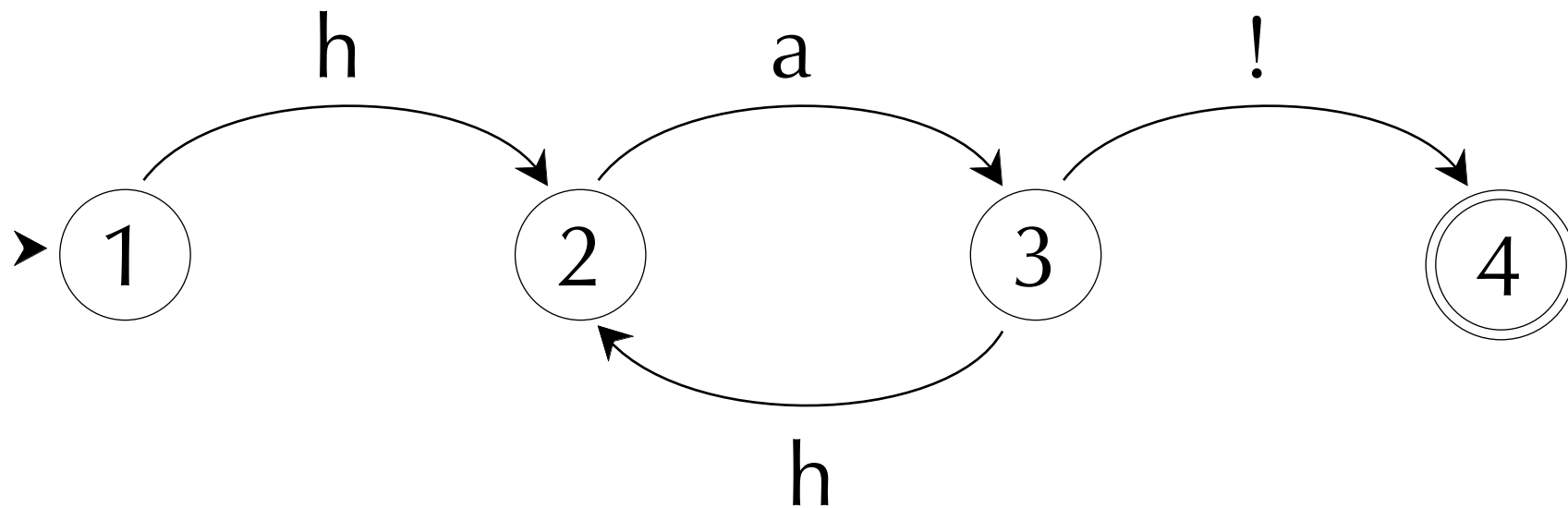
# Endliche Automaten



## Endliche Automaten (*Finite-State Automata*)

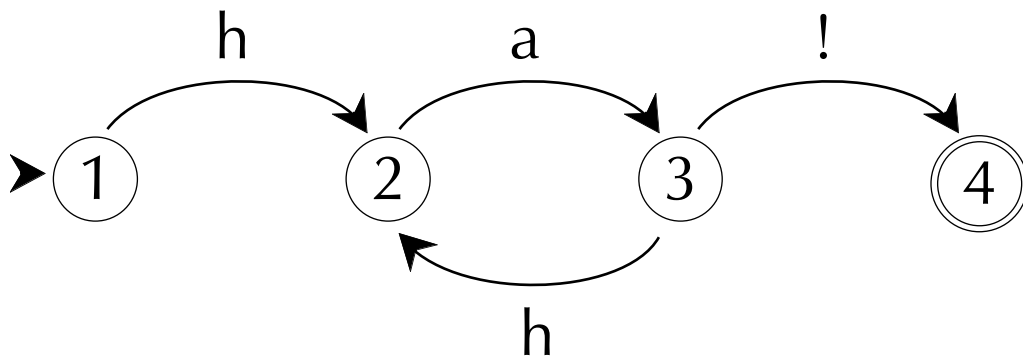
- ◆ mathematisch wohl-definiert
- ◆ wichtiges theoretisches Konzept
  - ◆ Informatik: Grundlage der Berechenbarkeitstheorie
  - ◆ Sprachwissenschaft: Sind Teile der menschlichen Sprache mit Endlichen Automaten beschreibbar? Struktur von Wörtern, Sätzen, ganzen Dialogen
- ◆ leicht zu implementieren
- ◆ extrem effizient bei der Verarbeitung mit einem Computer
- ◆ viele Anwendungen
  - ◆ Doppelklick in einer Textverarbeitung  $\Rightarrow$  Auswahl eines Wortes
  - ◆ viele Kommunikationsprotokolle als Endliche Automaten definiert
  - ◆ Sprachverarbeitung: Lexikonzugriff, Message-Extraction-Verfahren, ...

# Ein Lachautomat



# Eingabe des Automaten

Der Lachautomat erhält eine Zeichenkette als Eingabe:



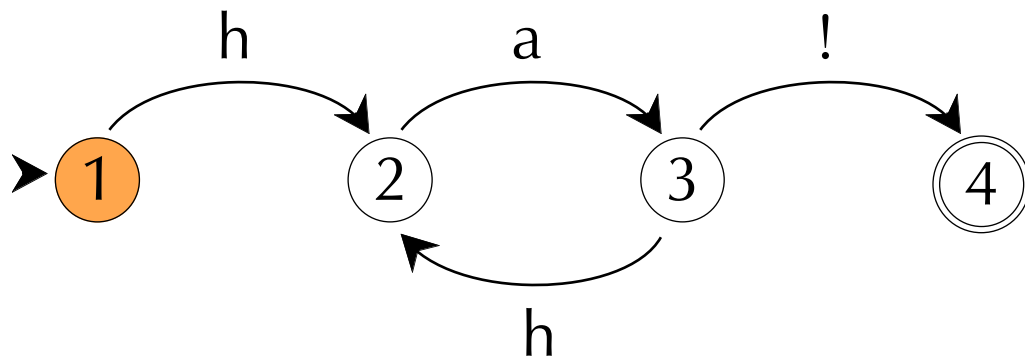
**h a h a !**

Was macht der Automat damit?

# Beginn der Verarbeitung

## Zu Beginn:

- ◆ der Automat ist im Startzustand ➤ ○
- ◆ er schaut auf das erste Zeichen der Eingabe

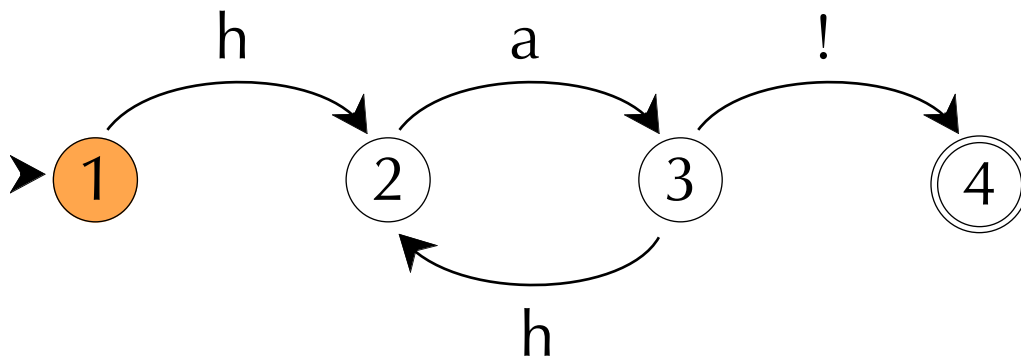
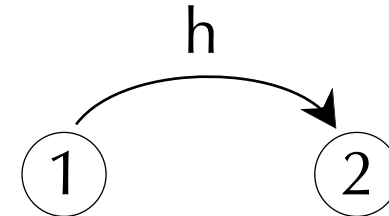


**h a h a !**

# Ein einzelner Verarbeitungsschritt

Der Automat nimmt jenen Übergang,

- ◆ der vom aktuellen Zustand ausgeht,
- ◆ und mit jenem Zeichen in der Eingabekette beschriftet ist, auf das der Automat gerade schaut.

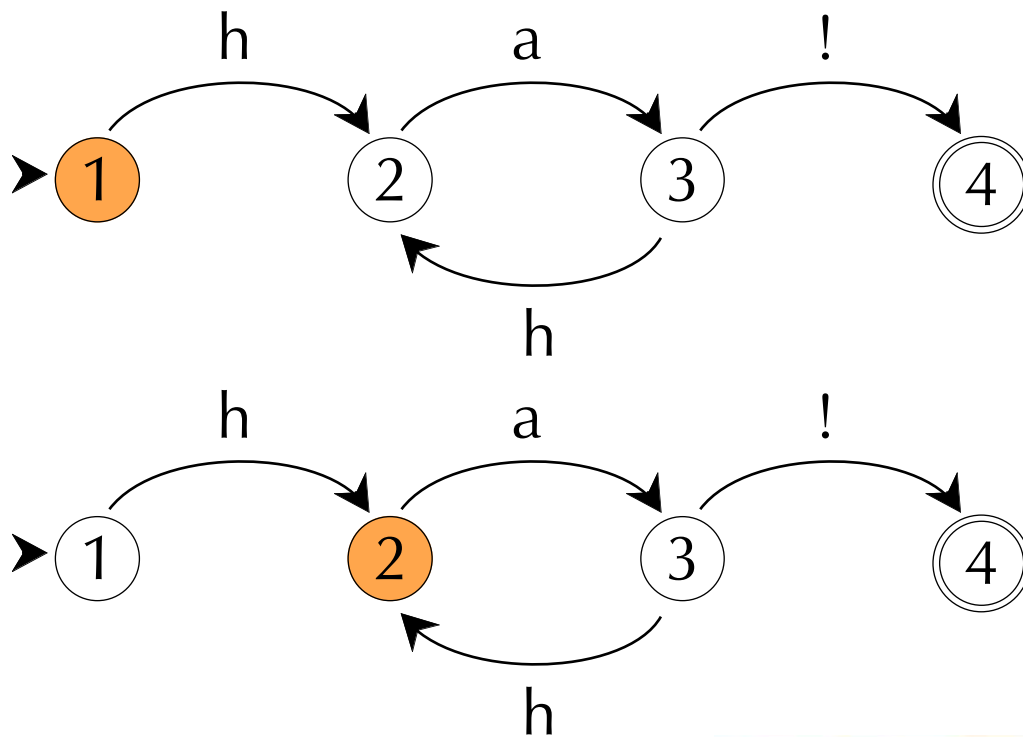


**h a h a !**

# Ein einzelner Verarbeitungsschritt

## Beim Nehmen eines Übergangs

- ◆ springt der Automat in einen neuen Zustand
- ◆ schaut auf das nächste Zeichen in der Eingabekette

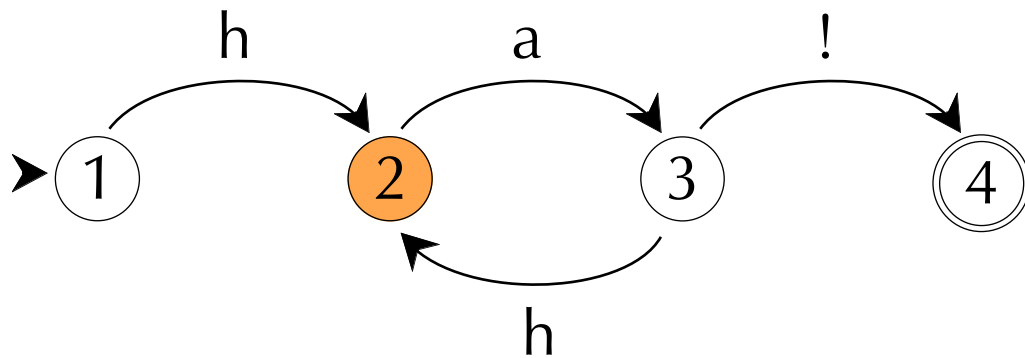


**h a h a !**

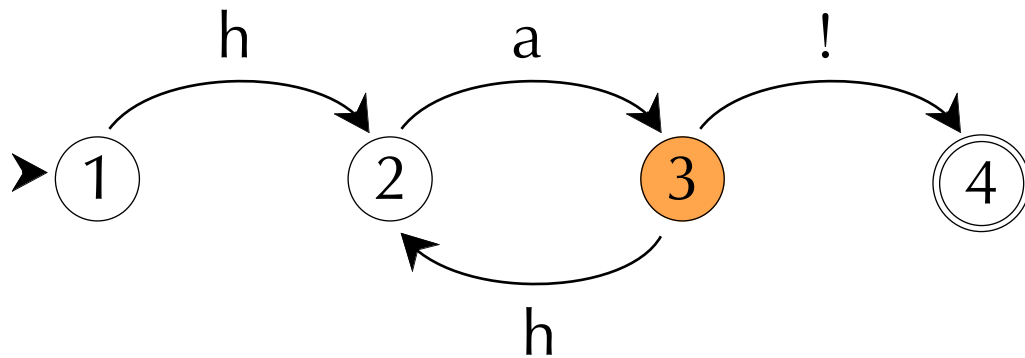
**h a h a !**

# Abarbeiten der Eingabe

Der Automat konsumiert so Zeichen um Zeichen ...



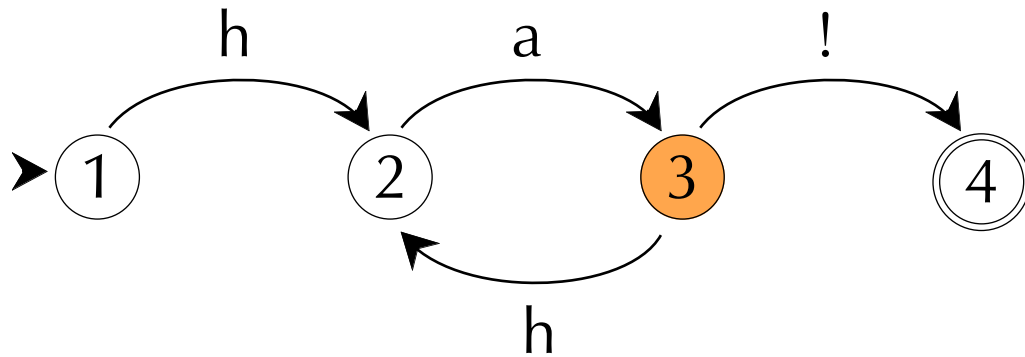
h a h a !



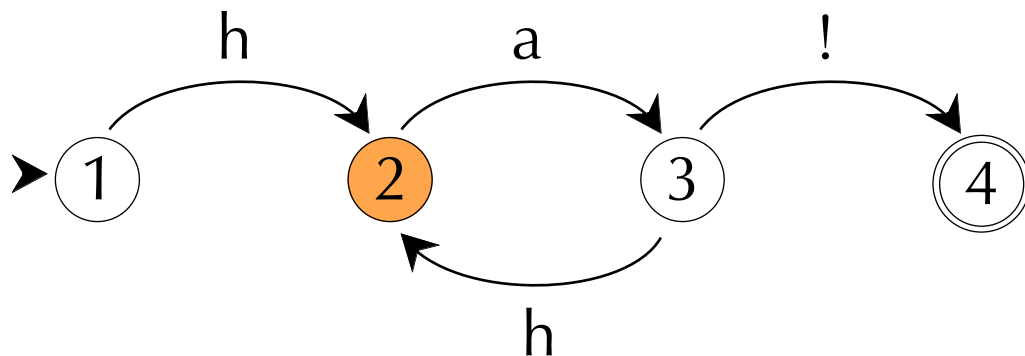
h a h a !



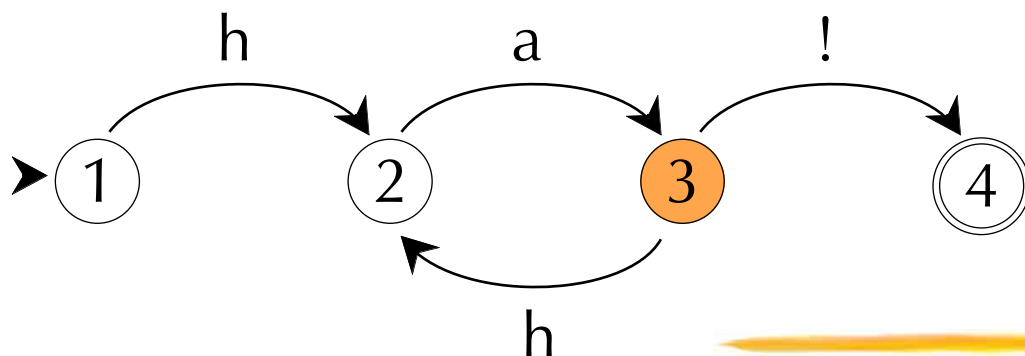
# Abarbeiten der Eingabe



h a h a !



h a h a !

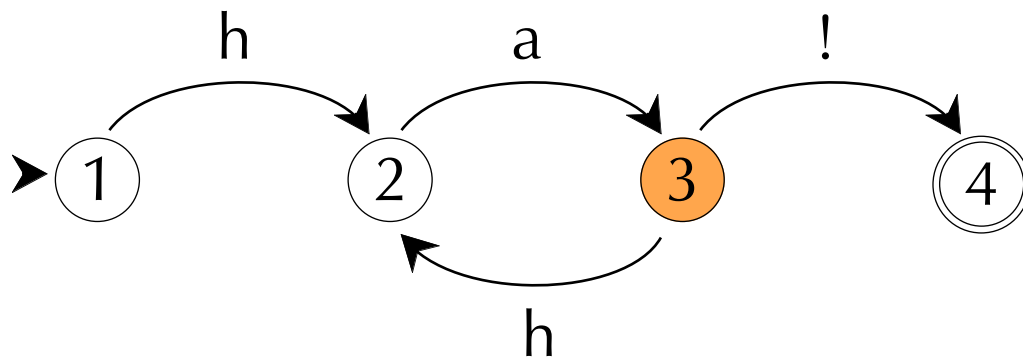


h a h a !

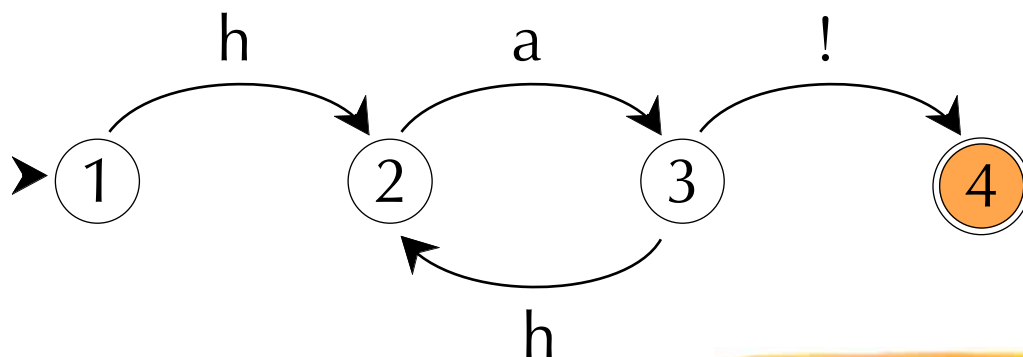
# Ende der Verarbeitung

Der Automat konsumiert so Zeichen um Zeichen ...

- ◆ bis auch das letzte Zeichen der Eingabe konsumiert wurde



h a h a !



h a h a !

# Ende der Verarbeitung

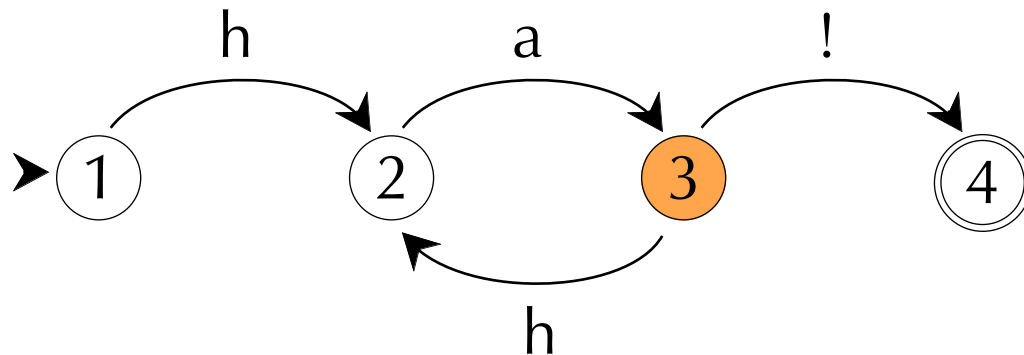
Wenn die Eingabe vollständig konsumiert ist, gibt es zwei Möglichkeiten:

- ◆ der aktuelle Zustand ist **ein Endzustand** (4)  
⇒ »der Automat hat die Eingabe akzeptiert«
- ◆ der aktuelle Zustand ist **kein Endzustand** (1) (2) (3)  
⇒ »der Automat hat die Eingabe nicht akzeptiert«

**Ein Automat kann auch mehrere Endzustände besitzen.**

# Ende der Verarbeitung

Kommt der Automat nicht weiter, weil kein Übergang zum aktuellen Eingabezeichen passt, wird die Eingabe ebenfalls *nicht* akzeptiert.



h a i !

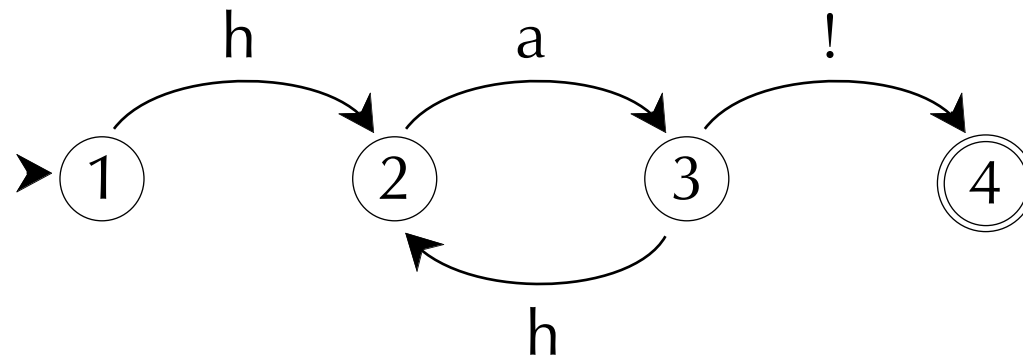
# Akzeptoren

## Der Lachautomat ist ein Akzeptor:

- ◆ Eingabe: Zeichenkette
- ◆ Ausgabe: »akzeptiert« oder »nicht akzeptiert«

ha! hahaha!  
haha!  
hahahaha!  
*Ausgabe: Ja*

hi! aha!  
ah! hahah!  
ha  
*Ausgabe: Nein*

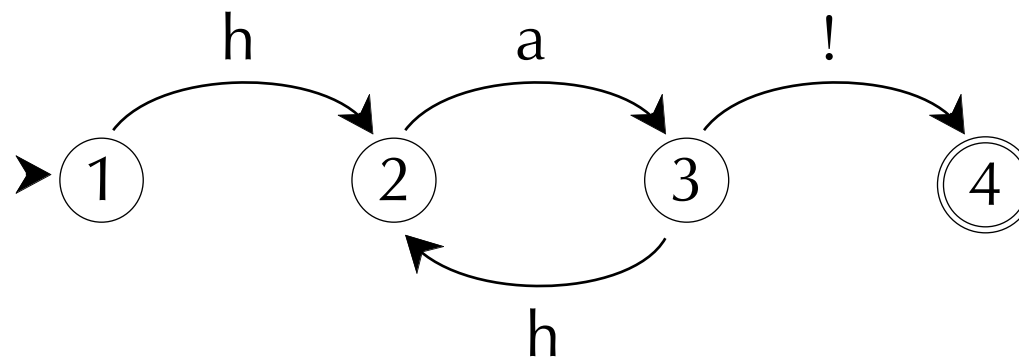


# Deterministische Endliche Automaten

## Deterministische Endliche Automaten (DEA)

### *Deterministic Finite-State Automata (DFA)*

- ◆ Von einem Zustand gehen nur Übergänge mit verschiedenen Beschriftungen aus.
- ▶ **Es kommt immer höchstens ein Übergang in Frage.**

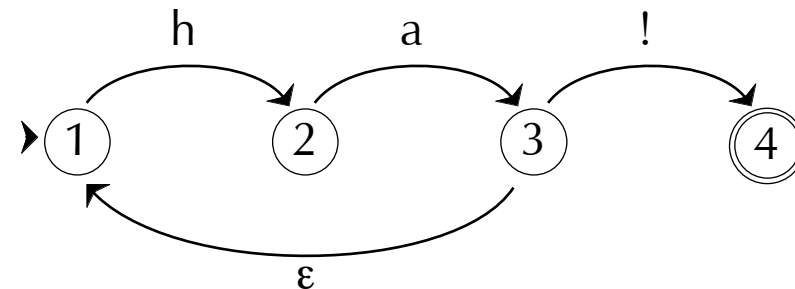
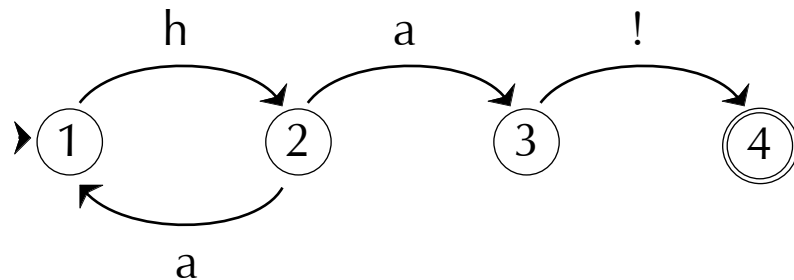


# Nicht-deterministische Endl. Autom.

## Nicht-deterministische Endliche Automaten (NEA)

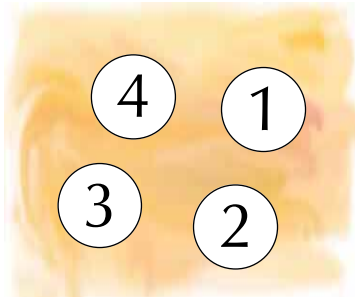
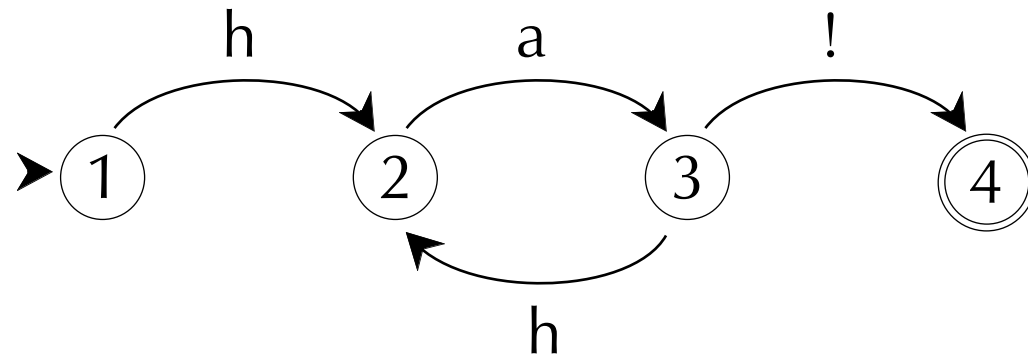
### *Non-deterministic Finite-State Automata (NFA)*

- ◆ Mehrere gleich beschriftete Übergänge von einem Zustand
- ◆  $\varepsilon$ -Übergänge, bei denen kein Eingabesymbol konsumiert wird
- ▶ Es kann geschehen, dass mehrere Übergänge gleichzeitig genommen werden können.



# Endliche Automaten: Bestandteile

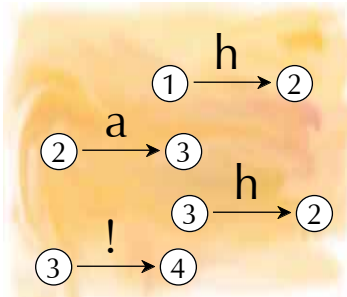
## Bestandteile eines Endlichen Automaten:



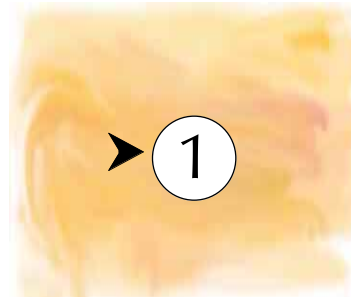
Zustände  
States



Alphabet  
Alphabet



Übergänge  
Transitions



Startzustand  
Start State



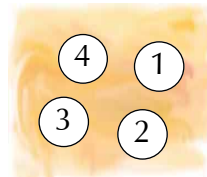
Endzustände  
Final States



# Deterministische Endliche Automaten

Ein Deterministischer Endlicher Automat ist ein Fünftupel

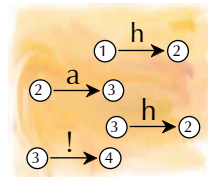
$$\langle K, \Sigma, \delta, s, F \rangle$$



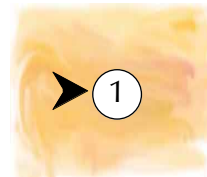
Zustände



Alphabet



Übergänge



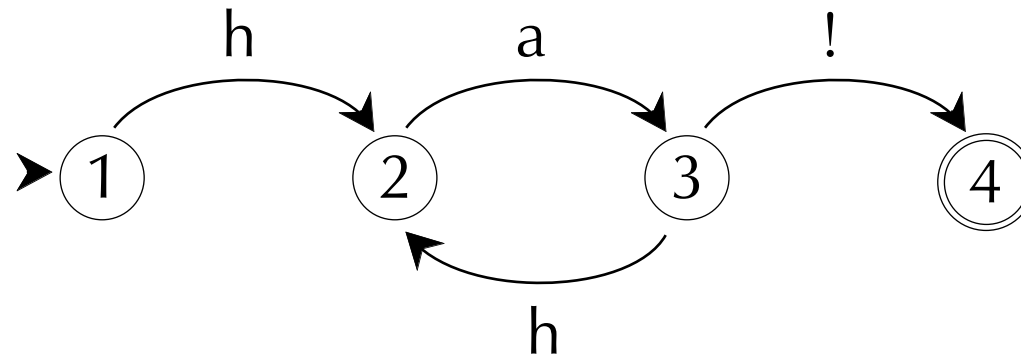
Startzustand



Endzustände

- ◆ endliche, nicht leere Menge von Zuständen  $K$
- ◆ Eingabealphabet  $\Sigma$
- ◆ Übergangsfunktion  $\delta \in (K \times \Sigma) \rightarrow K$
- ◆ Startzustand  $s$
- ◆ Menge von Endzuständen  $F$

# Deterministische Endliche Automaten



**Dieser Automat ist ein 5-Tupel  $\langle K, \Sigma, \delta, s, F \rangle$  mit**

- ◆  $K = \{1, 2, 3, 4\}$
- ◆  $\Sigma = \{a, i, h, !\}$
- ◆  $\delta = \{\langle 1, h, 2 \rangle, \langle 2, a, 3 \rangle, \langle 3, h, 2 \rangle, \langle 3, !, 4 \rangle\}$
- ◆  $s = 1$
- ◆  $F = \{4\}$

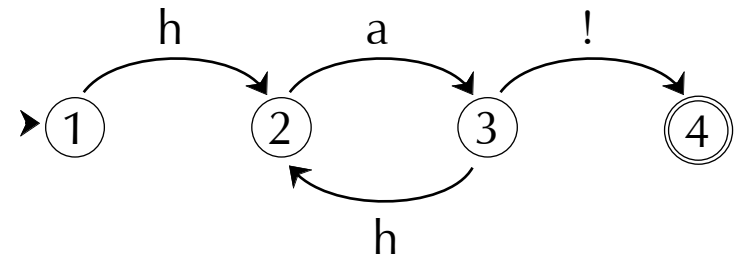
# Endliche Automaten: Begriffe

## Sprache eines Automaten

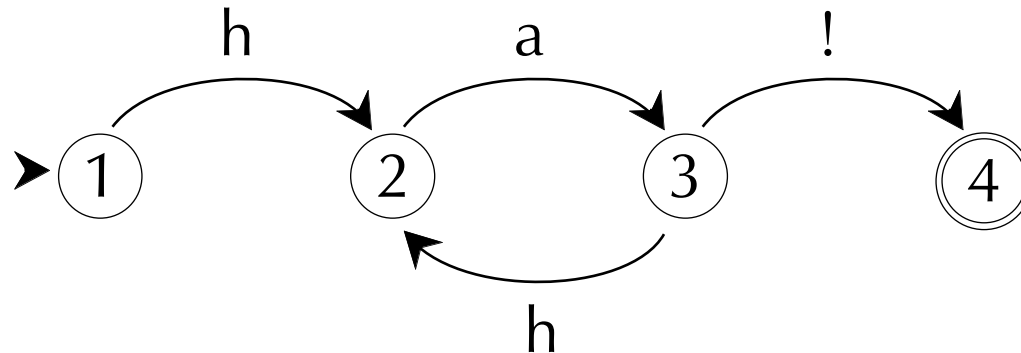
- ◆ Die Menge aller Eingabeketten, die von einem Endlichen Automaten  $A$  akzeptiert werden, heisst Sprache des Automaten, meist geschrieben als  $L(A)$ .

## Konfiguration, $\vdash$

- ◆ Das Paar  $\langle \text{Zustand}, \text{restliches Eingabewort} \rangle$  gibt vollständig an, wo in der Verarbeitung eines Eingabewortes ein Automat steht. Ein solches Paar heisst Konfiguration.
- ◆  $\vdash$  führt von einer Konfiguration zur nächsten.
- ◆ Beispiel:  $\langle 1, \text{haha!} \rangle \vdash \langle 2, \text{aha!} \rangle \vdash \langle 3, \text{ha!} \rangle \vdash \langle 2, \text{a!} \rangle \vdash \langle 3, \text{!} \rangle \vdash \langle 4, \epsilon \rangle$



# DEA in Prolog



```
start(1).
```

```
final(4).
```

```
delta(1, h, 2).
```

```
delta(2, a, 3).
```

```
delta(3, h, 2).
```

```
delta(3, '!', 4).
```

```
accept(String) :-
```

```
    start(StartState),
```

```
    accept(StartState, String).
```

```
accept(State, []) :-
```

```
    final(State).
```

```
accept(State, [Char|Chars]) :-
```

```
    delta(State, Char, NewState),
```

```
    accept(NewState, Chars).
```

# Literaturhinweise

---

## Mathematische Grundlagen der Linguistik:

- ◆ Barbara H. Partee/Alice ter Meulen/Robert E. Wall: *Mathematical Methods in Linguistics. Studies in Linguistics and Philosophy*, vol. 30. Dordrecht: Kluwer Academic Publishers, 1990. ISBN 90-277-2245-5 (Paperback); 90-277-2244-7 (gebunden).

*Ausführliche, sehr gut verständliche Einführung in Mengenlehre, Logik, Algebra, Lambda-Kalkül, Automatentheorie. Empfehlenswert.*

## Automatentheorie:

- ◆ Harry L. Lewis/Christos H. Papadimitriou: *Elements of the Theory of Computation*. London: Prentice-Hall, 1981. ISBN 0-13-273426-5. ca. CHF 77.—

*Schwierig, aber gut.*

# Aufgaben: Endliche Automaten

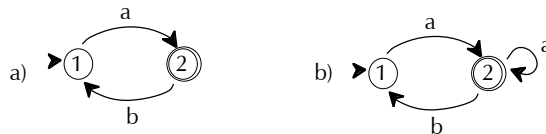
Programmiertechniken der Computerlinguistik 2 · Sommersemester 1999

## 1. Akzeptieren einer leeren Eingabe

Unter welchen Umständen akzeptiert ein Deterministischer Endlicher Automat eine leere Eingabe?

## 2. Akzeptierte Sprache

Welche Sprachen akzeptieren die folgenden Deterministischen Endlichen Automaten?



## 3. Lachautomat

Nicht alle Menschen lachen auf genau dieselbe Art und Weise. Erweitere den in der Vorlesung vorgestellten Lachautomaten so, dass er zusätzlich zu ha/haha/hahaha/... auch hi/hihi/hihihi/... und ho/hoho/hohoho/... erkennt. Hingegen sollen Kombinationen wie »hahi«, »hohahihahohihoha« etc. ausgeschlossen sein. Das Ausrufezeichen ist dieses Mal nicht Teil des Alphabets.

- Zeichne den Automaten graphisch.
- Gib den Automaten als Fünftupel  $\langle K, \Sigma, \delta, s, F \rangle$  an.
- Gib an, welche Konfigurationen bei der Eingabe *hohoo* aufeinander folgen.
- Implementiere den Automaten in Prolog.

## 4. Automaten entwerfen

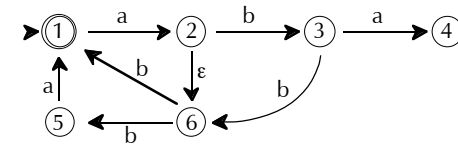
Entwirf Deterministische Endliche Automaten für die folgenden Sprachen über dem Alphabet  $\{a, b\}$ :

- Unmittelbar vor und nach jedem *a* steht ein *b*.
- Im Wort ist *abab* enthalten.
- Im Wort ist weder *aa* noch *bb* enthalten.

Zeichne die Automaten jeweils graphisch und gib sie als Fünftupel an.

## 5. Nichtdeterministische Endliche Automaten

Gegeben sei folgender nichtdeterministischer Automat. Welche der untenstehenden Eingabeketten werden von ihm akzeptiert?



- aa
- aba
- abb
- ab
- abab

## 6. Nominalphrasen erkennen

Teile eines Satzes wie »das grüne Haus«, »die schöne blaue Donau«, »die wahren, schönen und guten Werte« werden Nominalphrasen genannt.

In unserem (nicht allzu) hypothetischen Sprachverarbeitungssystem existiere bereits ein sogenannter *Part-of-Speech Tagger*, der jedem Wort seine Wortart zuweist.

Gib graphisch einen Endlichen Automaten (deterministisch oder nicht-deterministisch) an, der einfache deutsche Nominalphrasen erkennt. Der Automat arbeite auf dem Alphabet  $\Sigma = \{\text{Adjektiv, Verb, Nomen, Artikel, Präposition, Konjunktion, Komma}\}$ .

Wozu könnte ein solcher Automat zu gebrauchen sein?