University of Applied Sciences
HOCHSCHULE EMDEN·LEER

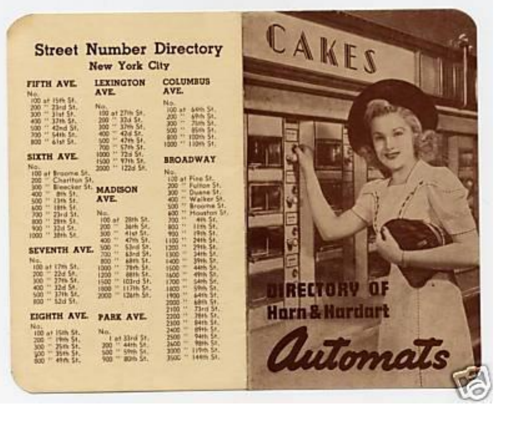
Theoretische Informatik
Automaten

Prof. Dr. Juho Mäkiö

24.10. 2018

Automat?

- Maschine, die den Arbeitsablauf selbst steuert.
- In der Informatik
  - Ein allgemeines Modell für diskrete dynamische Systeme
  - Eine abstrakte Maschine, die sich nach vorgegebenen Regeln verhält.



"Automat" Sells Toys To Youngsters

When youngsters in Stockholm go shopping they can buy their toys in a new "automat for play-things." The display machine is divided into bins with a window in each bin. The child inspects the toys through the windows and de-posits his money to receive the toy he selects.

Automat in der Informatik

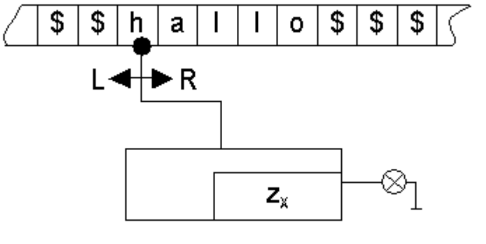
- Anwendung:
  - Verarbeitung von Sprachen
  - Studium von Grenzen der Informatik
  - Strukturierung dynamischer Programmabläufe, z.B.
  - Spiele-Engines
  - Systemanalyse
  - Spezifikation, z.B. Protokolle
  - …

Automaten in der theoretischen Informatik

- Maschinenmodelle
  - Beschreiben das Verhalten zustandsabhängiger Systeme, die auf Eingaben unterschiedlich reagieren können, je nach dem, welchen Zustand sie von den vorheerigen Eingaben versetzt worden sind.
  - Turing-Maschine
  - Bestandteile:
    - Zustand,
    - Ein- / Ausgabe,
    - Zustandsübergang,
    - Start und Ende

Turing-Maschine

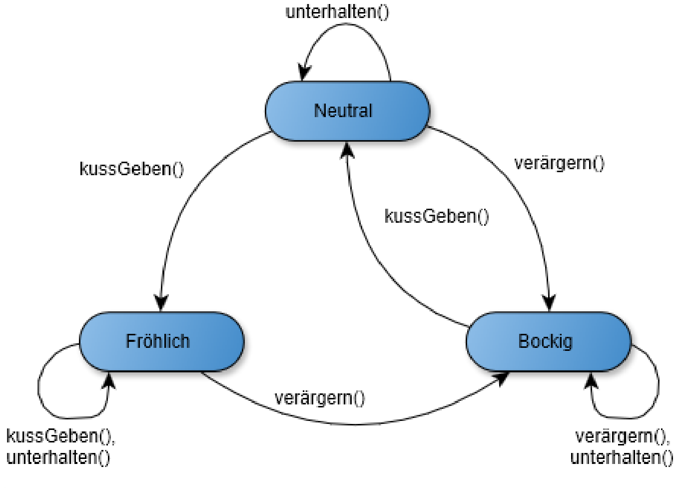
- ein unendlich langes Eingabeband mit Zellen für jedes Zeichen,
- eine endliche Menge von Zuständen
- Lese-Schreib-Kopf auf dem Band



Arbeitsweise einer Turing-Maschine

- Eingabezeichen lesen
- Schreiboperation auf das Band durchführen
  -> Bewegung des Lese-Schreibkopfes (links/rechts/keine)
  -> Zustandsübergang: abhängig vom aktuellen Zustand und dem Eingabezeichen
- Eventuell Wiederholung der vorherigen Schritte
- Turing-Maschine können deterministisch oder nichtdeterministisch arbeiten

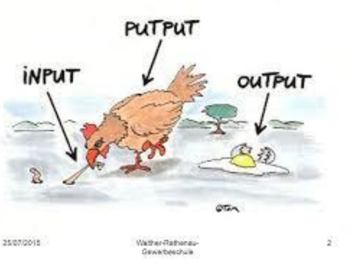
Zustand



https://www.philipphauer.de/study/se/design-pattern/state.php

Ein- / Ausgabe

- Ein- und Ausgabe = Modellierung der Wechselwirkung mit der Umgebung
- Ein- und Ausgaben werden zu Folgen zusammengefasst = Wörter
  →Zu einem Automat gehören Alphabete X und Y



Turing-Maschinen sind Modell eines geschlossenen Systems, also ohne Ein- und Ausgabe

Zustand

- Zustand = Momentaufnahme aller für das weitere Verhalten *wesentlichen Größen* (für Automat: diskrete Größen)
- Bei Turing-Maschine hieß das: Konfiguration
- Bei Automaten:
  Zustand = Element der Zustandsmenge Z

Zustandsübergang

- Abhängig vom Zustand und von der Eingabe
- Überführungsfunktion δ: erzeugt neuen Zustand
- Deterministisch: $\delta: Z \times X \rightarrow Z$
  oder
- Nichtdeterministisch : $\delta: Z \times X \rightarrow \rho(Z)$
- Erzeugt Ausgabe: $\lambda : Z \times X \rightarrow Y$

Start und Ende

- Anfangszustand: $z_0 \in Z$
- Menge von Endzuständen: $F \subseteq Z$
- Arbeit eines Automaten beginnt in z_0, und kann in einem Zustand aus F enden.

Automaten

- Es gibt verschiedene Automaten, die unterschiedliche Sprachklassen erkennen können, z.B.
  - endliche Automaten
    - deterministisch
    - nicht deterministisch
  - Kellerautomaten
  - Turing-Maschinen

abstrakter Automat in der theoretischen Informatik

- „Für die theoretische Informatik ist ein Computer eine Blackbox, in die Informationen hineingehen, verarbeitet werden und Ergebnisinformationen herauskommen. Das dabei verwendete Modell eines abstrakten Automaten ist ein mathematisches Modell eines Systems mit diskreter Ein- und Ausgabe.“

Endliche Automat

- Maschine, die bei korrekter Bedienung (Eingabe) etwas leistet.
- Z.B. Getränkeautomat, der nach Eingabe eines Euros ein Getränk liefert.

Eingabe 1€ → Automat →  Getränk

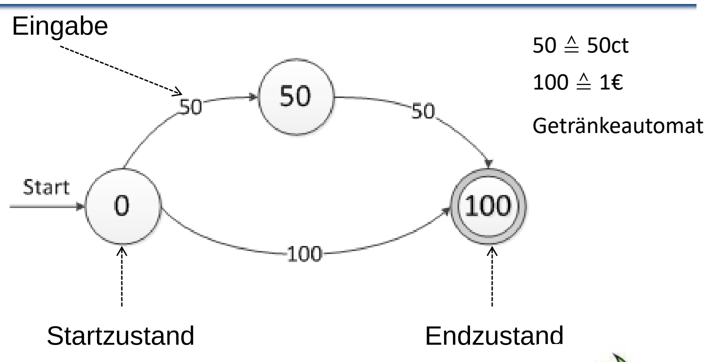
Der Automat akzeptiert 50ct und 1€-Münzen

Automat

- Wie merkt der Automat, dass insgesamt 1€ eingeworfen wurde?

→Zustände
- → Startzustand: 0€ in Automat
- → Zwischenzustand: 50ct im Automat
- → Endzustand:1€ im Automat

Zustandsdiagramm: Getränkeautomat



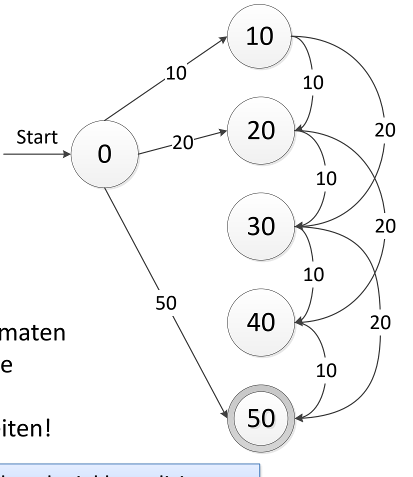
Der Automat ist sehr einfach:
- Er akzeptiert nur 50ct, 1€.
- 1€ muss exakt eingeworfen werden
- Nach Einwurf von 50ct akzeptiert er nur 50ct.

Getrankeautomat II

- Etwas komplizierterer Getränkeautomat :
  - Das Getränk kostet 50ct.
  - Es können 10ct, 20ct, 50ct eingeworfen werden.
  - 50ct müssen exakt eingeworfen werden

10 10 10 10 10
10 20 10 10
…
50

vom Automaten akzeptierte Eingabe-möglichkeiten!



Der erste Automat (1€) wird noch viel komplizierter, wenn er 10ct, 20ct, 50ct, 1€ akzeptieren sollte.

Definition NEA

- Ein nichtdeterministischer abstrakter Automat ist ein Tupel N = [∑,Q, V] mit
  - ∑ einer nicht leeren, endlichen Menge von Alphabetsbuchstaben
  - Q einer nicht leeren Menge von Zuständen
  - V einer auf Q x ∑ definierten Relation (Verhaltensgesetz)

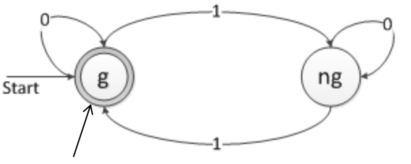
Ist V eine Funktion, so ist N ein deterministischer abstrakter Automat.

Beispiel….

- Gesucht ist ein Automat, der folgende (formale) Sprache über ∑ = {0, 1} erkennt / akzeptiert :
  - L = {ω ∈ ∑* | ω enthält eine gerade Anzahl von 1} → Paritätsprüfung
  - L = {ε, 11, 011, …, 01011010, …}
  - L = {ω ∈ ∑* | $|\omega|_1$ = 2*n für n ∈ ℕ}

Test:
Eingabe: 00101
Zustände: g→g →ng→g

Eingabe: 011010
Zustände: g→ng→g→g→ng→ng

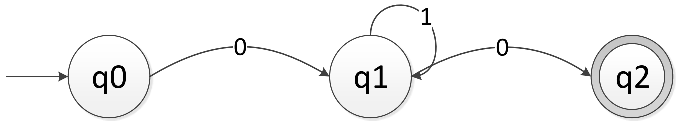


Endzustand

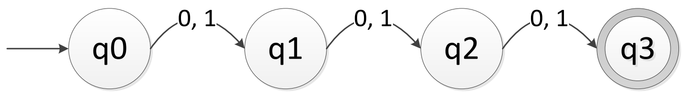
g ≙ gerade Anzahl von 1
ng ≙ keine gerade Anzahl von 1

Beispiel II

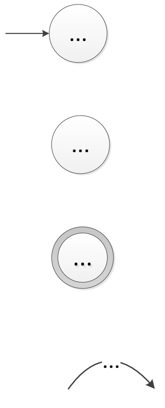
- $L = \{0\ 1^n\ 0 \mid n \geq 0\}$



- $L = \{\omega \mid \omega \in \Sigma^* \wedge |\omega| = 3\}$



Notation



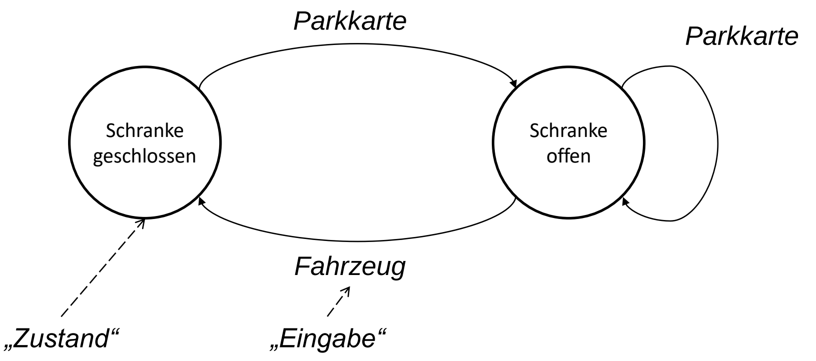
≙ Startzustand / Beginn

≙ Zustand

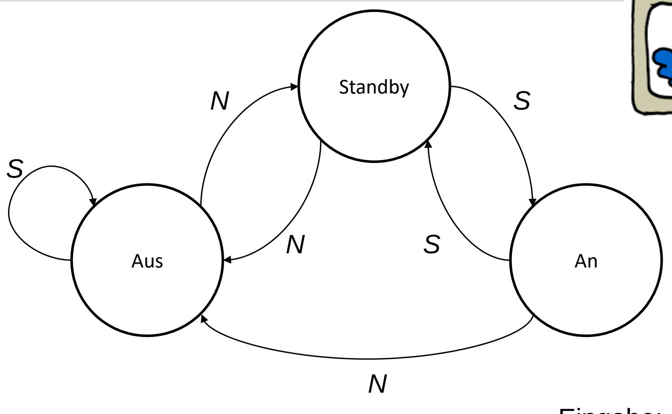
≙ Endzustand (akzeptierter Zustand)

≙ Zustandsüberführung bei Eingabe von …

Parkscheinautomat



Fernseher mit Standby-Funktion

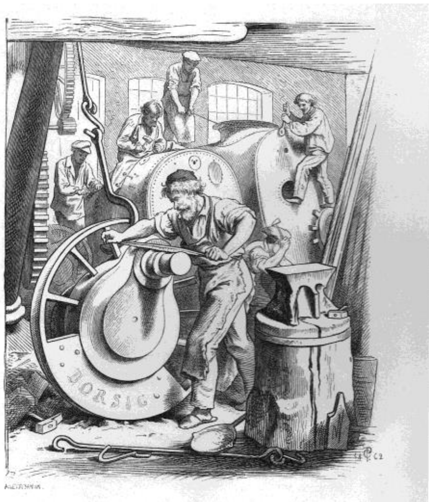


Zustände: An, Aus, Standby

Eingabe:
N = Netz-Taste
S = Standby-Taste

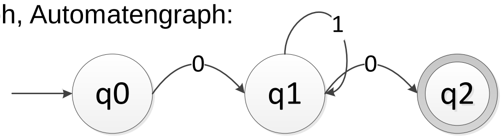
Deterministischer endliche Automat (DEA)

- Zustandsgraph/ Automatengraph (s.o)
- Automatentabelle (s.u.)
- 5-Tupel (s.u.) (Quintupel)



Automatentabelle

Zustandgraph, Automatengraph:



Automatentabelle:

Zustand	Eingabe	
	0	1
q_0	q_1	-
q_1	q_2	q_1
q_2	-	-

Automaten

Automaten mit und ohne Ausgabe

Automaten können grob anhand Ihrer Verwendung unterschieden werden:

Akzeptor

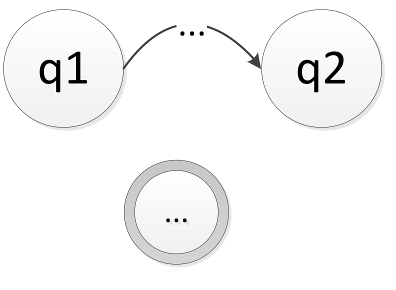
Wenn eine Eingabe den Automaten von einem bestimmten Zustand, dem Startzustand, in einen der Endzustände führt, dann sagt man, der Automat akzeptiert die Eingabe. Einen solchen Automaten nennt man daher einen **Akzeptor**: Eingabe → Akzeptiert / nicht akzeptiert

Transduktor ⤳ später

Automaten mit Ausgabe werden **Transduktoren** genannt. Sie ordnen entweder jedem Zustand (Moore-Automaten) oder jedem Paar aus Zustand und Eingabezeichen (Mealy-Automaten) ein Ausgabezeichen zu. Auf diese Weise bildet ein Automat eine Verarbeitungseinheit: Eingabe → Ausgabe

Automaten

- Die bisher behandelten Automaten sind Akzeptorautomaten:
  - Eingabealphabet Σ
  - Menge von Zuständen Z
  - Übergangsfunktion ≙
  - Endzustände

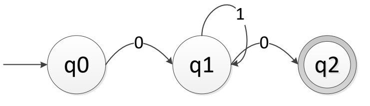


Definitipon DEA

- Ein deterministischer, endlicher (Akzeptor-) Automat (DEA) ist ein 5-Tupel $(Z, \Sigma, \delta, z_0, E)$ mit
  - $Z \triangleq$ endliche Menge von Zuständen
  - $\Sigma \triangleq$ endliches Eingabealphabet
  - $\delta : Z \times \Sigma \to Z \triangleq$ Übergangsfunktion
  - $z_0 \in Z \triangleq$ Startzustand
  - $E \subseteq Z \triangleq$ Menge der Endzustände (akzeptierten Zustände)

Erläuterungen

- endlich ✓
- deterministisch:
  - $\delta : Z \times \Sigma \to Z$
  - „Der Übergang ist eindeutig bestimmt (deterministisch).“
  - z.B. $\delta_{((q0,\ 0))} = q_1$



$L = \{0\ 1^n\ 0 \mid n \geq 0\}$

$Z = \{q_0, q_1, q_2\}$

$\Sigma = \{0, 1\}$

$\delta : Z \times \Sigma \to Z$

$\delta_{((q0,\ 0))} = q_1$

$\delta_{((q1,\ 0))} = q_2$

$\delta_{((q1,\ 1))} = q_1$

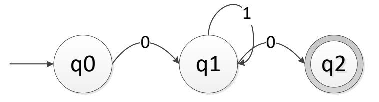
q0 : Startzustand

{q2} : Menge der Endzustände (1 Endzustand)

Akzeptierte Wörter

- Lediglich die Wörter $\omega \in \Sigma^*$, die zum Endzustand führen, werden als akzeptierte bzw. erkannte Wörter bezeichnet.
- Bsp.:

ω = 0 111 0



01110	q_0 -> q_1
01110	q_1 -> q_1
01110	q_1 -> q_1
01110	q_1 -> q_1
01110	q_1 -> q_2 (Endzustand)

Alles abgearbeitet + Endzustand → akzeptiert
Dieser Automat erkennt das Wort 01110.

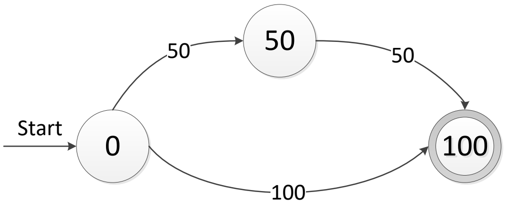
Akzeptierte Wörter?

- Welche Sprache L akzeptiert nun ein Automat A?

- Ist eine Sprache L die von A akzeptierte Sprache?

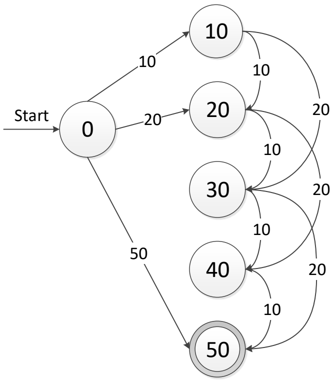
Aufgaben

- Geben Sie für den unteren Automaten die Automatentabelle und das 5-Tupel an.



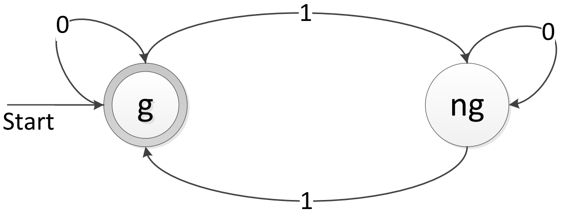
Aufgaben

- Geben Sie für den unten liegenden Automaten das 5-Tupel an.
  - Zeigen Sie, dass das Wort 10 20 20 akzeptiert wird.
  - Zeigen Sie, dass das Wort 10 10 10 bzw. 10 50 nicht akzeptiert wird.



Aufgaben

- Geben Sie für den unten ligenden Automaten das 5-Tupel an.
  - Zeigen Sie, dass 01101010 akzeptiert wird.



Akzeptierte Sprache

- Gegeben ist A = (Z, Σ, δ, z_0, E).
  - L(A) bezeichnet die von A erkannte, akzeptierte Sprache mit L(A) $\subset \Sigma^*$.
  - L(A) = {$\omega \in \Sigma^*$ | ω wird von A akzeptiert}
    $\omega = \omega 0 \ldots \omega n$ mit $\omega i \in \Sigma$ für i = 0, …, n
  - L(A) = $\{\omega_0 \ldots \omega_n \mid \delta(\ldots(\delta(\delta(z_0, \omega_0)),\omega_1),\ldots,\omega_n) \in E\}$

- ε gehört zu L(A), falls $z_0 \in E$ gilt.

Beispiel

$\omega = 01110 \in L(A_L)$

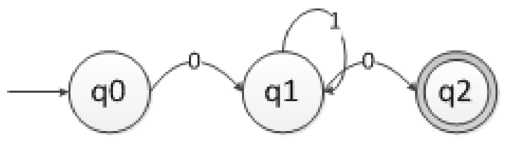
$\delta(\ \delta(\ \delta(\ \delta(\ \delta(q_0,\ 0)\ ,1)\ ,1)\ ,1)\ ,0) = q_2 \in E$

$= q_1$

$= q_1$

$= q_1$

$= q_1$



Konfigration und Übergangsrelation

$A = (Z, \Sigma, \delta, z_0, E)$

- Konfiguration: $(z, \omega) \in Z \times \Sigma^*$
  - $z \triangleq$ aktueller Zustand des Automaten
  - $\omega \triangleq$ das noch verbleibende Eingabewort (noch nicht verarbeitet)
- Übergangsrelation:
  - $(z, a\omega')$ -> (z', ω')
  - mit $\omega = a\omega'$, $a \in \Sigma$ und $\delta(z, a) = z'$

Mehrfache Anwendung der Übergangsrelation

$(z, a\underline{\omega'}) \quad \text{->} \quad (z', \omega')$

$= b\omega''$

$\rightarrow (z'', \omega'')$

mit $\delta(z', b) = z''$

$\rightarrow \ldots$

$\rightarrow^*$

- A akzeptiert ω genau dann, wenn es ein $z_e \in E$ gibt mit
  $(z_0, \omega) \rightarrow^* (z_e, \varepsilon)$
  - komplette Verarbeitung von ω
  - Erreichung eines Endzustandes
- Damit
- $L(A) = \{\omega \in \Sigma^* \mid (z_0, \omega) \rightarrow^* (z_e, \varepsilon), z_e \in E\}$

Yes we can…