

Theoretische Informatik

Rechnen an sich

Was ist „Rechnen“?

Welche Rechenmechanismen gibt es?

Kann man „alles“ „berechnen“?

Wie „teuer“ ist Rechnen?

Rechenmodelle / Rechensituationen

1. PC

Eingaben:

Tastatur
Maus
SD-Kartenleser
WLAN
Webcam
...



Ausgaben:

Bildschirm
Lautsprecher
Drucker
...

Abbildung 1: Rechnermodell PC

Auf bestimmte Eingaben sollen bestimmte Ausgaben erfolgen.

2. AIRBUS

Viele vernetzte Computereinheiten

Eingaben: Steuerungen durch Besatzung, Sensoren (Temperatur, GPS, Luftgeschwindigkeit)

Ausgabe: Info für Besatzung, Aktuatoren

3. „Einfachste Situation“

soll Funktion F : Zeichenketten $\rightarrow$ Zeichenketten ($\{0; 1\}$ JA/NEIN) berechnen

Eingabe: s ; Ausgabe $F(s)$

Einschub: Zeichenketten

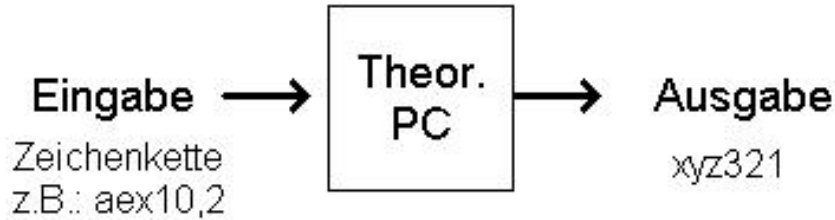


Abbildung 2: Rechnermodell „Einfachste Situation“

- Alphabet Σ : endliche Menge
- String (Zeichenkette) über Σ : endliche Folge von Elementen von Σ z.B.:
 $v = v_1v_2\dots v_k; \quad v_i \in \Sigma \quad \forall i, |v| = k$ (Länge)
- Σ^k : Menge der Strings der Länge k
- $\Sigma^0 = \{\varepsilon\}$; ε ist der leere String
- $\Sigma^* = \bigcup_{k \geq 0} \Sigma^k$: Menge aller Strings über Σ (mit endl. Länge)
- Konkatenation: $u, v \in \Sigma^*$; $u \in \Sigma^l; v \in \Sigma^k$

$$uv = u_1u_2\dots u_lv_1\dots v_k$$

$$u\varepsilon = u$$

$$\varepsilon u = u$$

- $v \in \Sigma^*; i \in \mathbb{N} = \{0, 1, \dots\}$

$$v^0 = \varepsilon$$

$$v^i = vv^{i-1}$$

- $L \subset \Sigma^*$: „Sprache über Σ “

Einschub Ende.

„Maschine definiert Sprache“

$$\{s \in \Sigma^* \mid F(s) = \text{JA}\}$$

Welche Sprachen können von welchen Arten von „Automaten“ berechnet werden?
 3 Arten von Automaten: Endlicher Automat, Keller Automat, Turing Maschine

1. Endlicher Automat



Abbildung 3: Endlicher Automat

endlich viele Steuerungsregeln folgender Form:

Wenn Zustand = $q \wedge$ Eingabekopf liest Zeichen a dann

1. Rücke Eingabekopf um eine Zelle nach rechts
2. wechsele in Zustand q'

Q : Zustandsmenge

$s \in Q$: Startzustand

$F \subset Q$: Endzustände

Anfangskonfiguration: Maschine ist im Startzustand s , Eingabekopf ganz links

Endkonfiguration: Maschine in einem der Endzustände, Eingabekopf ganz rechts

Maschine akzeptiert einen Eingabestring u , wenn aus der Startkonfiguration durch wiederholtes Anwenden von Regeln eine Endkonfiguration erreicht (wird) werden kann.

Bsp.: $\Sigma = \{a, b\}$, $Q = \{\underbrace{0}_{=s}, 1, 2\}$, $F = \{2\}$

derzeitiger Zustand	gelesenes Zeichen	neuer Zustand
0	a	1
0	b	0
1	a	2
1	b	0
2	a	2
2	b	0

Beispiel akzeptiert Sprache aller Strings über $\{a, b\}$, die in aa enden.

derzeitiger Zustand	gelesenes Zeichen	neuer Zustand
$\alpha: 0$	a	0
$\beta: 0$	b	0
$\gamma: 0$	a	1
$\delta: 1$	a	2

nicht-deterministische Maschine: mehr als eine Regel in gleicher Situation anwendbar

2. Kellerautomat

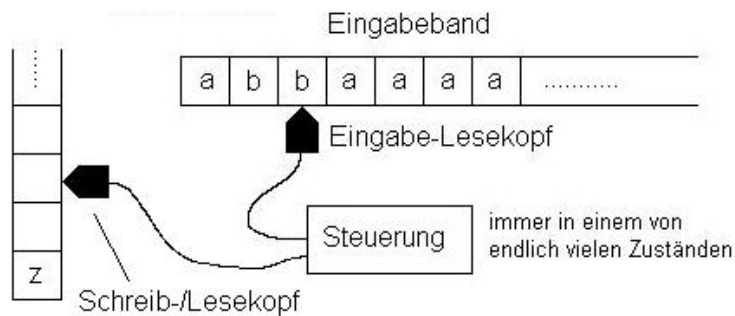


Abbildung 4: Keller-Automat

Q : Zustandsmenge

$s \in Q$: Startzustand

$F \subset Q$: Endzustände

Regeln:

Wenn Zustand = $q \wedge$ Eingabekopf liest $a \wedge$ Arbeitskopf liest A , dann

1. gehe in Zustand q'
2. Bewege Eingabekopf
 - (i) rechts

(ii) nicht

3. Arbeitskopf:

(i) ersetze A durch A'

(ii) Kopf nach unten, A gelöscht

(iii) Kopf nach oben, schreibe A'

Anfangskonfiguration: Zustand = s , Eingabekopf ganz links, Arbeitsband:

Lesekopf ganz unten, z in unterster Zelle

Endkonfiguration: Maschine in einem der Endzustände. Eingabekopf ganz rechts

$$\{a^n b^n | n \geq 1\}$$