Grundlagen der Theoretischen Informatik

Sommersemester 2015

29.04.2015

Viorica Sofronie-Stokkermans

e-mail: sofronie@uni-koblenz.de

Bis jetzt

1. Motivation
2. Terminologie
3. Endliche Automaten und reguläre Sprachen
4. Kellerautomaten und kontextfreie Sprachen
5. Turingmaschinen und rekursiv aufzählbare Sprachen
6. Berechenbarkeit, (Un-)Entscheidbarkeit
7. Komplexitätsklassen P und NP

Bis jetzt

- **Alphabete, Wörter**
  - Operationen auf Wörtern

    Konkatenation, i-te Potenz, Reverse
- **Sprache**
  - Operationen auf Sprachen

    Konkatenation, i-te Potenz, Reverse, Kleene-Hülle
- **Reguläre Ausdrücke**
- **Grammatiken**
  - Ableitung
  - Die von einer Grammatik erzeugte Sprache.
- **Die Chomsky-Hierarchie**

Grammatik

Definition. Eine Grammatik G über einem Alphabet Σ ist ein Tupel $G = (V, T, R, S)$. Dabei ist

- V eine endliche Menge von **Variablen**
- $T \subseteq \Sigma$ eine endliche Menge von **Terminalen** mit $V \cap T = \emptyset$
- R eine endliche Menge von **Regeln**
- $S \in V$ das **Startsymbol**

Konvention (meistens): **Variablen** als Großbuchstaben; **Terminale** als Kleinbuchstaben.

Eine **Regel** ist ein Element $(P, Q) \in \left((V \cup T)^* \; V \; (V \cup T)^*\right) \times (V \cup T)^*$.

- P und Q sind Wörter über $(V \cup T)$
- P muss mindestens eine Variable enthalten
- Q ist beliebig

Schreibweise: $P \to_G Q$, $P \to Q$ (P: Prämisse, Q: Conclusio)

Erzeugte Sprache, Äquivalenz

Ableitung, Rechnung.

- $w \Longrightarrow_G w'$ („w geht über in w'“) falls

$$\exists u, v \in (V \cup T)^* \; \exists P \to Q \in R \;\; \big(w = uPv \text{ und } w' = uQv\big)$$

- $w \Longrightarrow_G^* w'$ falls es Wörter $w_0, \ldots, w_n \in (V \cup T)^*$ $(n \geq 0)$ gibt mit
  - $w = w_0$
  - $w_m = w'$
  - $w_i \Longrightarrow_G w_{i+1}$ für $0 \leq i < n$

Definition (Erzeugte Sprache) Gegeben: Eine Grammatik G
Die von G erzeugte Sprache $L(G)$ ist die Menge aller **terminalen** Wörter, die durch G vom Startsymbol S aus erzeugt werden können:

$$L(G) \quad := \quad \{w \in T^* \mid S \Longrightarrow_G^* w\}$$

Definition Zwei Grammatiken G_1, G_2 heißen **äquivalent** gdw $L(G_1) = L(G_2)$.

Die Chomsky-Hierarchie

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **rechtslinear** gdw

$$\forall (P \to Q) \in R \quad \left(P \in V \text{ und } Q \in T^* \cup T^+ V\right)$$

Das heißt, bei jeder Regelanwendung:

- Links eine **einzelne Variable**; Rechts **höchstens eine Variable**
- Wenn rechts eine Variable steht, steht sie **ganz rechts im Wort**.

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **kontextfrei** gdw

$$\forall (P \to Q) \in R \quad \left(P \in V \text{ und } Q \in (V \cup T)^*\right)$$

Das heißt, bei jeder Regelanwendung:

- Links eine **einzelne Variable**; Rechts steht etwas beliebiges
- Die Prämisse macht keine Aussage, was der Kontext dieser Variablen ist („kontextfrei“)

Die Chomsky-Hierarchie

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **kontextsensitiv** gdw $\forall (P \to Q) \in R$:

1. $\exists u, v, \alpha \in (V \cup T)^* \; \exists A \in V \; \big(P = uAv$ und $Q = u\alpha v$ mit $|\alpha| \geq 1\big)$, **oder** die Regel hat die Form $S \to \varepsilon$
2. S nicht in Q

Das heißt, bei jeder Regelanwendung:

- Eine Variable A wird in einen String α mit $|\alpha| \geq 1$ überführt; die Ersetzung von A durch α findet nur statt, wenn der in der Regel geforderte **Kontext (u und v), vorhanden ist**
- **Das Wort wird nicht kürzer, außer bei $\varepsilon \in L$**

Definition Eine Grammatik $G = (V, T, R, S)$ heißt **beschränkt** gdw $\forall (P \to Q) \in R$:

1. $|P| \leq |Q|$, **oder** die Regel hat die Form $S \to \varepsilon$
2. S nicht in Q

Das heißt, bei jeder Regelanwendung:

- Die Conclusio ist mindestens so lang wie die Prämisse, außer bei $\varepsilon \in L$.
- Das Wort wird nicht kürzer, außer bei $\varepsilon \in L$

Die Chomsky-Hierarchie

Aufbauend auf den Grammatikarten kann man Sprachklassen definieren.

Definition (Sprachklassen)

Klasse	**definiert als**	**Sprache heißt**
$\mathbf{L_3}$, REG	$\{L(G) \mid G$ ist rechtslinear$\}$	Typ 3, **regulär**
$\mathbf{L_2}$, CFL	$\{L(G) \mid G$ ist kontextfrei$\}$	Typ 2, **kontextfrei**
$\mathbf{L_1}$, CSL	$\{L(G) \mid G$ ist kontextsensitiv$\}$	Typ 1, **kontextsensitiv**
$\mathbf{L_1}$, CSL	$\{L(G) \mid G$ ist beschränkt$\}$	Typ 1, **beschränkt**
$\mathbf{L_0}$, r.e.	$\{L(G) \mid G$ beliebig$\}$	Typ 0, **aufzählbar**
$\mathbf{L}$	$\{L \mid \ L \subseteq \Sigma^* \}$	**beliebige** Sprache

Probleme über Sprachen

Interessante Probleme (informell)

- Ist ein gegebenes Wort in einer Sprache (definiert durch eine Grammatik) enthalten?
- Erzeugen zwei gegebene Grammatiken dieselbe Sprache?

Mit welchen Algorithmen können diese Probleme gelöst werden?

Definition (Problem, Algorithmus)

Ein **Problem** P ist die Frage, ob eine bestimmte Eigenschaft auf gegebene Objekte zutrifft.

Dabei ist eine bekannte, abzählbaren Grundmenge solcher Objekte gegeben. Für jedes Objekt o gilt: die Eigenschaft trifft auf o zu oder nicht.

Ein **Algorithmus** für ein Problem P ist eine Vorschrift (ein Programm), die zu beliebigem Objekt o berechnet, ob die Eigenschaft für o zutrifft oder nicht.

Endlich, unendlich und dann?

Abzählbarkeit

Definition (Abzählbarkeit)
Eine Menge M heißt **abzählbar**, wenn

- es eine surjektive Funktion $f : \mathbb{N} \to M$ gibt,
- oder M leer ist.

Intuition

Eine Menge ist abzählbar, wenn sie höchstens so mächtig wie $\mathbb{N}$ ist.

Abzählbarkeit

Lemma. Eine Menge M ist abzählbar, wenn es eine injektive Funktion

$$f : M \to \mathbb{N}$$

gibt.

Abzählbarkeit

Beispiel:

Abzählbar sind:

- $\mathbb{N}$
- $\mathbb{Q}$
- alle endlichen Mengen
- die Vereinigung zweier abzählbarer Mengen
- die Vereiningung abzählbar vieler abzählbarer Mengen

David Hilbert

David Hilbert (1862-1943)

- Einer der bedeutensten und einflußreichsten Mathematiker aller Zeiten
- Professor in Königsberg und Göttingen
- Wichtige Beiträge zu
  - Logik
  - Funktionalanalysis
  - Zahlentheorie
  - Mathematische Grundlagen der Physik
  - uvm.

Hilbert's Hotel

In einem Hotel mit endlich vielen Zimmern können keine Gäste mehr aufgenommen werden, sobald alle Zimmer belegt sind (Schubfachprinzip).

Hilbert's Hotel

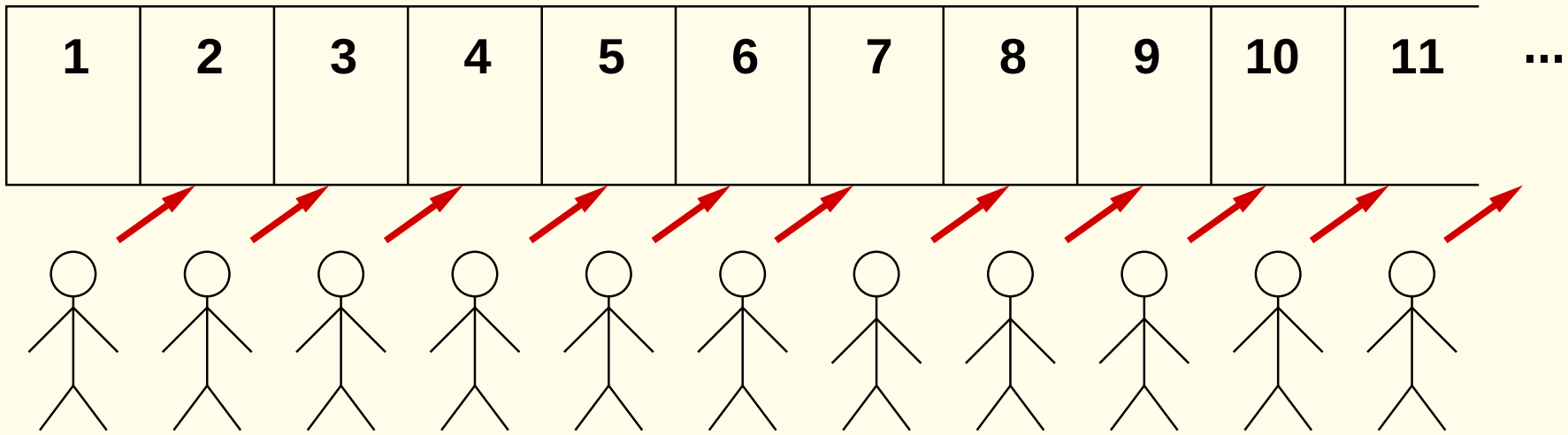
In einem Hotel mit endlich vielen Zimmern können keine Gäste mehr aufgenommen werden, sobald alle Zimmer belegt sind (Schubfachprinzip).

Hilberts Hotel hat nun unendlich viele Zimmer (durchnummeriert mit natürlichen Zahlen bei 1 beginnend).

1	2	3	4	5	6	7	8	9	10	11	...

Man könnte annehmen, dass dasselbe Problem auch hier auftreten würde, wenn alle Zimmer durch (unendlich viele) Gäste belegt sind.

Hilbert's Hotel



Es gibt einen Weg, Platz für einen weiteren Gast zu machen, obwohl alle Zimmer belegt sind:

- Der Gast von Zimmer 1 geht in Zimmer 2,
- der Gast von Zimmer 2 geht in Zimmer 3,
- der von Zimmer 3 nach Zimmer 4 usw.

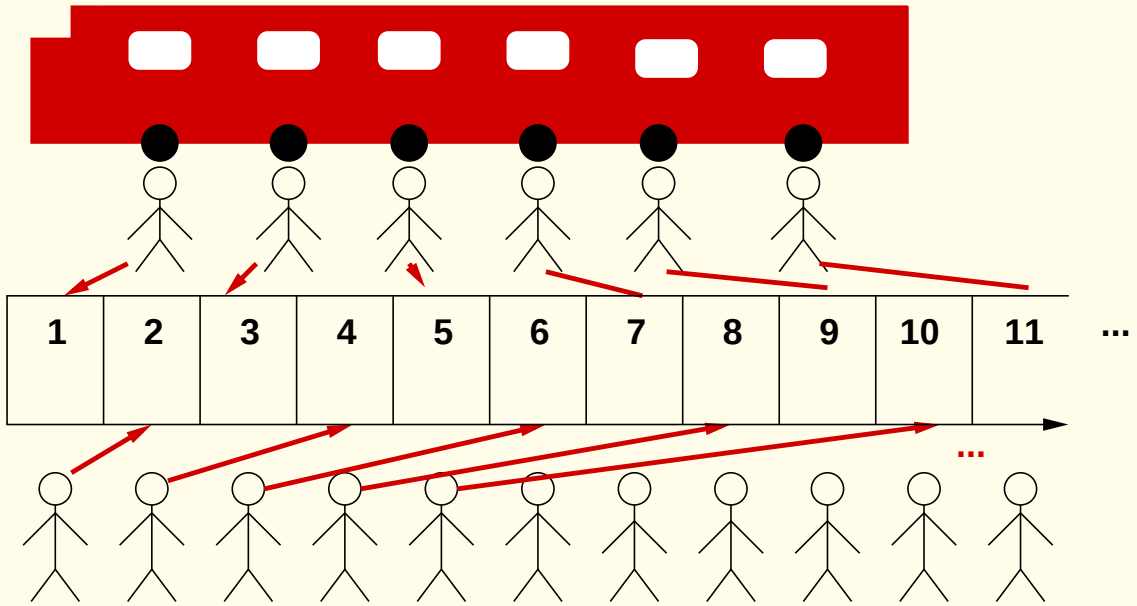
Damit wird Zimmer 1 frei für den neuen Gast.

Hilbert's Hotel

Da die Anzahl der Zimmer unendlich ist, gibt es keinen "letzten" Gast, der nicht in ein weiteres Zimmer umziehen könnte. Wiederholt man das, erhält man Platz für eine beliebige, aber endliche Zahl neuer Gäste.

Hilbert's Hotel

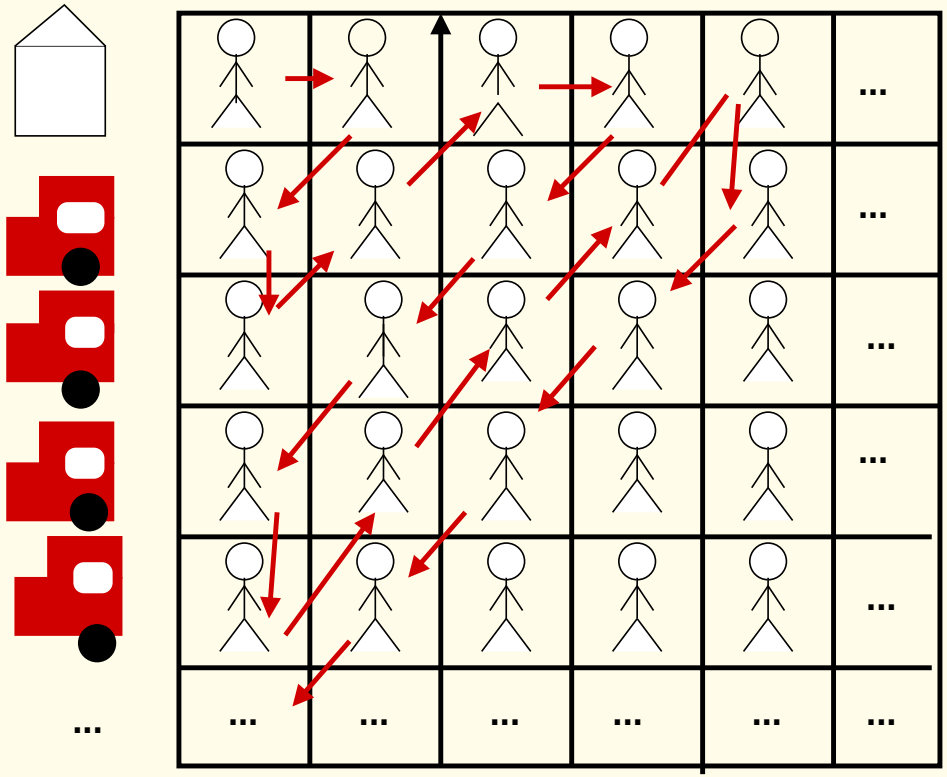
Es ist sogar möglich, Platz für abzählbar unendlich viele neue Gäste zu machen.



Der Gast von Zimmer 1 geht in Zimmer 2, der Gast von Zimmer 2 aber in Zimmer 4, der von Zimmer 3 in Zimmer 6 usw. Damit werden alle Zimmer mit ungerader Nummer frei für die abzählbar unendlich vielen Neuankömmlinge.

Hilbert's Hotel

Wenn abzählbar unendlich viele Busse mit je abzählbar unendlich vielen Gästen vorfahren, können auch diese Gäste alle im bereits vollen Hotel untergebracht werden.



Hilbert's Hotel

Andere Möglichkeit:

- die Zimmer mit ungeraden Nummern wie eben beschrieben wird frei gemacht
- die Gäste aus Bus 1 in die Zimmer 3, 9, 27, ... schickt (also in jene Zimmer, die mit Potenzen von 3 nummeriert sind; $3 = 3^1, 9 = 3^2, 27 = 3^3, \ldots$),
- die Gäste aus Bus 2 in die Zimmer 5, 25, 125, 625, etc., usw.,

also die Gäste aus Bus i in die Zimmer p_i, p_i^2, p_i^3 etc.,
wobei p_i die $i+1$-te Primzahl ist.

Dadurch sind alle angekommenen Gäste im Hotel untergebracht und sogar noch unendlich viele Zimmer (wie zum Beispiel das Zimmer 15, dessen Nummer keine Potenz einer Primzahl ist) frei.

Hilbert's Hotel

Weitere Möglichkeit:

Eine andere, effizientere Möglichkeit wäre die, die Hotelgäste jeweils aus den Zimmern n in die Zimmer $2n - 1$ umziehen zu lassen, sodass alle geraden Zimmer frei werden.

Dann können die neuen Gäste aus dem Bus mit der Nummer n die Zimmer belegen, deren Zimmernummern durch 2^n, nicht aber durch 2^{n+1} teilbar sind, sodass kein Zimmer frei bliebe.

Korollar

$\mathbb{N}$ ist abzählbar

Alle endlichen Mengen sind abzählbar

Korollar

- die Vereinigung zweier abzählbarer Mengen ist abzählbar

  Falls Σ_1, Σ_2 abzählbar, so ist $\Sigma_1 \cup \Sigma_2$ abzählbar

- die Vereiningung abzählbar vieler abzählbarer Mengen ist abzählbar

  Falls $\Sigma_1, \Sigma_2, \ldots$ abzählbar, so ist $\bigcup_{i \in \mathbb{N}} \Sigma_i$ abzählbar

  Falls Σ abzählbar, so ist Σ^i abzählbar für alle $i \in \mathbb{N}$.

Diagonalisierungsargument für Überabzählbarkeit

Theorem. Die Menge $\mathbb{R}$ der rellen Zahlen ist überabzählbar.

Beweis. Wir zeigen, dass schon das Intervall $[0, 1]$ überabzählbar ist.

Annahme: Es gibt eine Aufzählung, also eine surjektive Funktion

$$f : \mathbb{N} \to [0, 1]$$

Dann sei

$$f(i) = 0, d_0^i d_1^i d_2^i \ldots$$

die Dezimaldarstellung der i-ten reellen Zahl.

Diagonalisierungsargument für Überabzählbarkeit

Beweis. Fortsetzung:

Wir definieren eine neue Zahl $d = 0, \overline{d}_0 \overline{d}_1 \overline{d}_2 \ldots$ durch

$$\overline{d}_n = \begin{cases} d_n^n + 1 & \text{falls } d_n^n < 9 \\ 0 & \text{sonst} \end{cases}$$

d unterscheidet sich in der n-ten Stelle von d_n.

Also $d \neq d_n$ für alle $n \in \mathbb{N}$

Also kommt d in der Aufzählung nicht vor. Widerspruch!

Wieviele gibt es?

Wieviele

– Grammatiken
– Sprachen
– Algorithmen

gibt es Überhaupt?

Wieviele gibt es?

Wieviele

– Grammatiken
– Sprachen
– Algorithmen

gibt es Überhaupt?

Mögliche Antworten:

- Endlich viele
- Unendlich viele
- Abzählbar viele
- Überabzählbar viele
- Nicht klar für Algorithmen, da dieser Begriff nicht genau definiert wurde

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Signatur Σ, endlich oder abzählbar unendlich

Dann ist Σ^* abzählbar unendlich.

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Signatur Σ, endlich oder abzählbar unendlich

Dann ist Σ^* abzählbar unendlich.

Beweis.

Σ ist abzählbar, also ist Σ^i abzählbar, für alle $i \in \mathbb{N}$.

Σ^* ist die Vereinigung der abzählbar vielen abzählbaren Mengen Σ^i.

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Signatur Σ, endlich oder abzählbar unendlich

Dann ist die Menge aller Grammatiken über Σ abzählbar unendlich

Wieviele Wörter, Grammatiken gibt es?

Lemma. Gegeben: Signatur Σ, endlich oder abzählbar unendlich

Dann ist die Menge aller Grammatiken über Σ abzählbar unendlich

Beweis.

Grammatiken sind endlich und also als Wörter über einer geeigneten erweiterten Signatur

$$\Sigma \cup V \cup \{\rightarrow, \ldots\}$$

darstellbar.

Die Menge der Wörter über dieser erweiterten Signatur ist abzählbar.

Wieviele Algorithmen gibt es?

Lemma. Es gibt (nur) abzählbar viele Algorithmen.

Wieviele Algorithmen gibt es?

Lemma. Es gibt (nur) abzählbar viele Algorithmen.

Beweis.

Algorithmen müssen per Definition eine endliche Beschreibung haben.

Sie sind also als Wörter über einer Signatur Σ darstellbar
(für jedes abzählbare Σ).

Also sind sie abzählbar.

Wieviele Funktionen $f : \mathbb{N} \to \mathbb{N}$ gibt es?

Lemma. Es gibt überabzählbar viele Funktionen $f : \mathbb{N} \to \mathbb{N}$.

Wieviele Funktionen $f : \mathbb{N} \to \mathbb{N}$ gibt es?

Lemma. Es gibt überabzählbar viele Funktionen $f : \mathbb{N} \to \mathbb{N}$.

Beweis.

Angenommen, es existiere eine Abzählung

$$f_1, f_2, \ldots f_n, \ldots$$

Dann sei

$$C : \mathbb{N} \to \mathbb{N} \quad \text{mit} \quad C(n) = \begin{cases} 1 & \text{falls } f_n(n) = 0; \\ 0 & \text{sonst} \end{cases}$$

$C(i) \neq f_i(i)$

Also: C ist von allen f_i verschieden. Widerspruch!

Wieviele Funktionen $f : \mathbb{N} \to \{0, 1\}$ gibt es?

Lemma. Es gibt überabzählbar viele Funktionen $f : \mathbb{N} \to \{0, 1\}$.

Beweis. Analog.

Wieviele Sprachen gibt es?

Lemma. Gegeben eine Signatur Σ (endlich oder unendlich).

Die Menge der Sprachen über Σ ist überabzählbar.

Wieviele Sprachen gibt es?

Lemma. Gegeben eine Signatur Σ (endlich oder unendlich).

Die Menge der Sprachen über Σ ist überabzählbar.

Beweis.

Sei eine beliebige Abzählung aller Wörter über Σ gegeben:

$$w_1, w_2, \ldots$$

Dann kann man die Sprachen L über Σ mit den Funktionen $f : \mathbb{N} \to \{0, 1\}$ identifizieren, vermittels

$$f(i) = 1 \quad \text{gdw} \quad w_i \in L$$

Von diesen gibt es überabzählbar viele.

Korollar

Korollar.

Nicht jede Sprache kann durch eine Grammatik dargestellt werden.

Zusammenfassung

Gegeben eine Signatur Σ

Abzählbar

- $\mathbb{N}$
- Menge aller Wörter
- Menge aller Grammatiken
- Menge aller Algorithmen

Zusammenfassung

Gegeben eine Signatur Σ

Abzählbar

- $\mathbb{N}$
- Menge aller Wörter
- Menge aller Grammatiken
- Menge aller Algorithmen

Überabzählbar

- Die Menge aller Teilmengen von $\mathbb{N}$
- Die Menge aller reellen Zahlen
- Die Menge aller Funktionen $f : \mathbb{N} \to \mathbb{N}$ bzw. $f : \mathbb{N} \to \{0, 1\}$
- Die Menge aller Sprachen