

Einführung in Berechenbarkeit, Komplexität und formale Sprachen

WS 2019/20

Prof. Dr. Christian Scheideler

Wäre es nicht schön ...

Könnte ich
doch nur das
Halteproblem
lösen!



Großer fermatscher Satz: Die Gleichung

$$a^n + b^n = c^n$$

**hat für positive ganze Zahlen a, b, c, n mit $n > 2$
keine Lösung.**

Wäre es nicht schön ...

Alg(n)

```
for t from 1 to  $\infty$  do
  for x from 1 to t
    for y from 1 to t
      for z from 1 to t
        if  $x^n + y^n = z^n$ 
          return (x, y, z)
```

Könnte ich
doch nur das
Halteproblem
lösen!



Alg hält bei Eingabe n genau dann, wenn $x, y, z \in \mathbb{N}$ existieren mit $x^n + y^n = z^n$.

Halteproblem

Gibt es einen Algorithmus Halte, der

- als Eingabe einen beliebigen Algorithmus Alg und eine Eingabe w für Alg erhält,
- entscheidet, ob Alg gestartet mit Eingabe w hält?

Satz (Turing) Einen solchen Algorithmus kann es nicht geben.

Die zentrale Frage

**Was sind die wesentlichen Möglichkeiten
und Grenzen von Computern?**

Berechenbarkeit und Komplexität

- **Was ist ein Problem?**

Entscheidungsproblem,

Berechnung einer Funktion,

Optimierungsproblem, ...

Entscheidungsproblem: Befindet sich auf dem Bild eine Katze?



Foto: vetproduction

Berechnung einer Funktion:



Optimierungsproblem: Das Problem des Handlungsreisenden



Berechenbarkeit und Komplexität

- **Was ist ein Problem?**

 - Entscheidungsproblem,

 - Berechnung einer Funktion,

 - Optimierungsproblem, ...

- **Was ist ein Algorithmus, Computer?**

 - Registermaschinen (RAMs), μ -Rekursion,

 - λ -Kalkulus, Turingmaschinen,...

Berechenbarkeit und Komplexität

- Alle bekannten Rechenmodelle können sich gegenseitig simulieren (inkl. Quantencomputern!)
→ Churchsche These
- Es gibt Probleme, die **nicht** algorithmisch gelöst werden können → **Berechenbarkeit**.
- Manche Probleme können zwar algorithmisch gelöst werden, aber sie können **nicht effizient** algorithmisch gelöst werden → **Komplexität**.

Notationen:

- $A = \{a_1, \dots, a_l\}$: **Alphabet**, a_i 's: **Symbole / Buchstaben**
- $A^0 := \{\varepsilon\}$, wobei ε das **leere Wort** ist.
- $A^* = \bigcup_{n=0}^{\infty} A^n$: Menge aller Worte
- $A^+ = \bigcup_{n=1}^{\infty} A^n$: Menge aller nichtleeren Worte
- $A^{\leq m} = \bigcup_{n=0}^m A^n$: Menge aller Worte der Länge $\leq m$

- $\alpha \in A^*$: **Wort / Satz** über Alphabet A
- $L \subseteq A^*$: **Sprache** über A , $\bar{L} = A^* \setminus L$

Notationen:

- Für $\alpha, \beta \in A^*$ ist $\alpha \circ \beta$ die Konkatenation von α und β .
Vereinfacht schreiben wir auch $\alpha\beta$.
- Für ein $\alpha \in A^*$ ist $|\alpha|$ die Länge von α .
- Für ein $a \in A$ ist $a^0 = \varepsilon$, $a^n = aa \dots a$ (n-mal).
- Für ein $n \in \mathbb{N}$ ist $\text{bin}(n)$ die Binärdarstellung von n .
- Für $b_0, \dots, b_{r-1} \in \{0, 1\}$ bezeichnet $(b_{r-1}, \dots, b_0)_2$ die durch b_0, \dots, b_{r-1} binär dargestellte Zahl z , d.h.

$$z = \sum_{i=0}^{r-1} b_i 2^i$$