Grundlagen der Programmierung

Grenzen von Algorithmen:

Berechenbarkeit ♦ Abzählbarkeit

Algorithmisches Denken: Vom Problem zur Lösung

1. **Identifizieren des Problems**
2. **Formulieren des Problems**
3. **Entwurf des Algorithmus**
4. **Implementierung des Algorithmus**
5. **Anwendung des Algorithmus**

→ Problemlösung

Sind alle Probleme algorithmisch lösbar?

- **Behauptung:**
  Nein: sogar die meisten Probleme sind **nicht** algorithmisch lösbar.

➢ Die meisten Funktionen sind **nicht** (algorithmisch) **berechenbar**.

➢ Es gibt (deutlich) mehr Funktionen als Algorithmen.

- *Mehr?*
  Es gibt unendlich viele Funktionen und Algorithmen.
  ???

Unendlichkeitsbegriffe

- Es gibt in der Mathematik verschiedene „Grade“ der Unendlichkeit.
- Es gibt (*nur*) **abzählbar unendlich** viele Algorithmen.
- Es gibt (*aber*) **überabzählbar viele** Funktionen.

Abzählbar unendliche Mengen

- Eine Menge *M* ist **abzählbar unendlich**, wenn sie gleichmächtig zur Menge $\mathbb{N}$ der natürlichen Zahlen ist.

- Zwei Mengen *A* und *B* heißen **gleichmächtig**, $A \sim B$, wenn es eine eineindeutige Abbildung von *A* auf *B* *(also eine Bijektion)* gibt.
  - jedes Element von *A* wird abgebildet (DB: *A*)
  - jedes Element von *B* kommt als Bild vor (WB: *B*)
  - 1-1-Abbildung

Abzählbarkeit: Beispiel Zahlenbereiche

- Die Gleichmächtigkeit ist eine Äquivalenzrelation.
  - **reflexiv**: $A \sim A$ mit $f = \mathrm{id}$
  - **transitiv**: Wenn $A \sim B$ mit f_1 und $B \sim C$ mit f_2 dann $A \sim C$ mit $f_2 \circ f_1$
  - **symmetrisch:** Wenn $A \sim B$ mit f dann $A \sim B$ mit f^{-1}

Abzählbarkeit: Beispiel Zahlenbereiche

- Die Menge $\mathbb{N}$ der natürlichen Zahlen ist abzählbar unendlich.

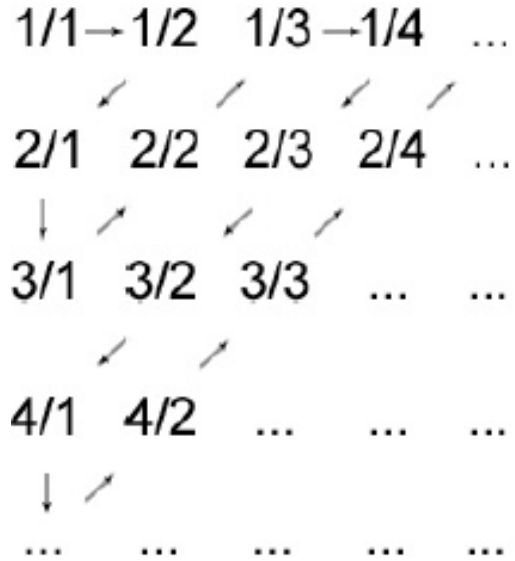
- Die Menge $\mathbb{Z}$ der ganzen Zahlen ist abzählbar unendlich.

$$f(n) = \begin{cases} 2n & \text{falls } n \geq 0 \\ -2n-1 & \text{falls } n < 0 \end{cases}$$

➢ Die Vereinigung zweier abzählbar unendlicher Mengen ist abzählbar unendlich.

Abzählbarkeit: Beispiel Zahlenbereiche

- Die Menge $\mathbb{Q}^+$ der positiven gebrochenen Zahlen ist abzählbar unendlich.



➢ Das kartesische Produkt zweier abzählbar unendlicher Mengen ist abzählbar unendlich.

Abzählbarkeit: Beispiel Zahlenbereiche

- Die Menge $\mathbb{Q}$ der rationalen Zahlen ist abzählbar unendlich.
  - $\mathbb{Q}^+$ ist abzählbar unendlich
  - $\mathbb{Q}^-$ ist abzählbar unendlich
  - Vereinigung zweier abzählbar unendlichen Mengen ist abzählbar unendlich.

  → Es existiert eine Bijektion $g: \mathbb{Q}^+ \cup \mathbb{Q}^- \longrightarrow \mathbb{N}$
  - Die rationale Zahl 0 muss noch abgebildet werden.
    → $f(0) = 0$; $f(n) = g(n)+1$ für alle $n > 0$

Überabzählbarkeit

- Die Menge $\mathbb{R}$ der reellen Zahlen ist **überabzählbar** unendlich (nicht abzählbar unendlich).
  - Reelle Zahlen sind unendliche Dezimalbrüche (ohne Neuner-Periode).
  - Beweis *indirekt*:
    Annahme: $\mathbb{R}$ ist abzählbar unendlich.
  - Dann gibt es eine Bijektion von $\mathbb{R}$ auf $\mathbb{N}$.

    Beschränken uns auf das Einheitsintervall (0; 1):

    Es gibt $F : (0; 1) \rightarrow \mathbb{N}$ *und somit eine Anordnung der Elemente aus* (0; 1): $x_0, x_1, x_2, x_3, x_4, x_5, \ldots$

Cantors Diagonalisierungsverfahren

- Seien die Zahlen aus dem reellen Intervall (0; 1) wie folgt angeordnet:

$x_0 = 0, a_{00}\ a_{01}\ a_{02}\ a_{03}\ a_{04}\ a_{05}\ \ldots$

$x_1 = 0, a_{10}\ a_{11}\ a_{12}\ a_{13}\ a_{14}\ a_{15}\ \ldots$

$x_2 = 0, a_{20}\ a_{21}\ a_{22}\ a_{23}\ a_{24}\ a_{25}\ \ldots$

$x_3 = 0, a_{30}\ a_{31}\ a_{32}\ a_{33}\ a_{34}\ a_{35}\ \ldots$

$x_4 = 0, a_{40}\ a_{41}\ a_{42}\ a_{43}\ a_{44}\ a_{45}\ \ldots$

$x_5 = 0, a_{50}\ a_{51}\ a_{52}\ a_{53}\ a_{54}\ a_{55}\ \ldots$

$\ldots$

Betrachten nun die Zahl

$b = 0, b_0\ b_1\ b_2\ b_3\ b_4\ b_5\ \ldots$

mit $b_n = \begin{cases} 1 & \text{falls } a_{nn} = 0 \\ 0 & \text{falls } a_{nn} > 0 \end{cases}$

$b \neq x_n$ für alle $n \geq 0$

→ **Widerspruch**, da $b \in (0; 1)$

Wir wollten zeigen:

1. Es gibt (*nur*) **abzählbar unendlich** viele Algorithmen.

2. Es gibt (*aber*) **überabzählbar viele** Funktionen.

Abzählen aller Algorithmen

- Algorithmen sind repräsentiert durch einen **Text**.
  - Pseudocode
  - Python-Code
  - ...
- Jeder Text ist eine endliche Zeichenkette (ein **Wort**).
  - Besteht aus ASCII-Zeichen oder Unicode-Zeichen, ... (inkl. Leerzeichen, Tabulator, Zeilenumbruch, ...)
  - Buchstaben des Wortes aus einem endlichen **Alphabet.**
  - Länge des Wortes ist endlich.
  - Es gibt m^n Wörter der Länge n über einem Alphabet mit m Buchstaben.

Abzählen aller Algorithmen

- Algorithmen sind repräsentiert durch einen **Text**.
- Es gibt m^n Wörter der Länge n über einem Alphabet mit m Buchstaben.
- Anordnung aller Wörter:
  - zuerst das leere Wort, dann alle Wörter der Länge 1, dann der Länge 2, 3, ... (Ordnen der Länge nach)
  - Wörter gleicher Länge werden alphabetisch geordnet.

➢ Somit existiert auch eine Anordnung aller Texte, die Algorithmen repräsentieren.

➢ Die Menge dieser Texte (damit aller Algorithmen) ist abzählbar unendlich.

Überabzählbar viele Funktionen

- Betrachten Menge ***P*** aller Funktionen von $\mathbb{N}$ in {0,1}.
- **Annahme:** ***P*** ist abzählbar unendlich.
  Dann existiert eine Bijektion $F : \boldsymbol{P} \rightarrow \mathbb{N}$ mit $F^{-1}(n) = f_n$

  Betrachten die Matrix aller $a_{nk} = f_n(k)$:

f_0 : $a_{00}\ a_{01}\ a_{02}\ a_{03}\ a_{04}\ a_{05}\ \ldots$
f_1 : $a_{10}\ a_{11}\ a_{12}\ a_{13}\ a_{14}\ a_{15}\ \ldots$
f_2 : $a_{20}\ a_{21}\ a_{22}\ a_{23}\ a_{24}\ a_{25}\ \ldots$
f_3 : $a_{30}\ a_{31}\ a_{32}\ a_{33}\ a_{34}\ a_{35}\ \ldots$
f_4 : $a_{40}\ a_{41}\ a_{42}\ a_{43}\ a_{44}\ a_{45}\ \ldots$
f_5 : $a_{50}\ a_{51}\ a_{52}\ a_{53}\ a_{54}\ a_{55}\ \ldots$
...

Betrachten nun die Funktion

$f : \mathbb{N} \rightarrow \{0,1\}$ mit

mit $f(n) = 1 - f_n(n)$

$\rightarrow\ f \neq f_n$ für alle $n \geq 0$

→ Widerspruch

Fazit

- Die meisten Funktionen sind nicht algorithmisch berechenbar.

- *Frage:*
  Gibt es eine (in der Informatik interessante) Funktion, die nicht algorithmisch berechenbar ist?