

# Berechenbarkeit und Komplexitätstheorie

Wintersemester 2024/2025

Aufgabenblatt 3

Abgabe: 19. November 2024 um 12.15 Uhr (in der Übung)

## Definition(en)

- Die Cantor'sche Bijektion  $\langle \cdot, \cdot \rangle : \mathbb{N}^2 \rightarrow \mathbb{N}$  ist definiert als:

$$\langle x, y \rangle = y + \left( \sum_{i=1}^{x+y} i \right) = y + \frac{(x+y)(x+y+1)}{2}$$

- Die  $k$ -stellige Cantor'sche Bijektion  $\langle \cdot \rangle : \mathbb{N}^k \rightarrow \mathbb{N}$  ist definiert als:

$$\langle x_1, \dots, x_{k-1}, x_k \rangle = \langle x_1, \langle \dots \langle x_{k-1}, x_k \rangle \dots \rangle \rangle$$

Weiter bezeichnen  $p_1, \dots, p_k$  die Umkehrfunktion(en) der Cantor'schen Bijektion, sodass  $\langle p_1(z), \dots, p_k(z) \rangle = z$  gilt.

- Die Ackermannfunktion  $A : \mathbb{N}_0 \times \mathbb{N}_0 \rightarrow \mathbb{N}$  ist wie folgt definiert:

$$\begin{aligned} A(0, y) &= y + 1 \\ A(x + 1, 0) &= A(x, 1) \\ A(x + 1, y + 1) &= A(x, A(x + 1, y)) \end{aligned}$$

## Aufgabe 3.1 (2 + 2 Punkte)

Sei  $f : \mathbb{N} \rightarrow \mathbb{N}$  eine WHILE-berechenbare, injektive, totale Funktion.

- Zeigen Sie, dass die Umkehrfunktion  $f^{-1}$  von  $f$  ebenfalls WHILE-berechenbar ist.
- Gilt das gleiche auch für LOOP-berechenbare Funktionen? (Begründen Sie Ihre Antwort.)

## Aufgabe 3.2 (2 + 2 Punkte)

- Gilt  $\langle 1, 2, 3 \rangle \leq \langle 2, 2, 2 \rangle$ ? (Begründen Sie kurz.)
- Bestimmen Sie  $a, b, c \in \mathbb{N}$ , sodass  $\langle a, b, c \rangle = 102$ .

## Aufgabe 3.3 (2 + 3 + 2 Punkte)

- Implementieren Sie die Ackermann-Funktion in einer Programmiersprache Ihrer Wahl! (Die gewählte Sprache sollte "lesbar" sein – BRAINFUCK, MALBOLGE oder sonstige esoterische Programmiersprachen werden wahrscheinlich nicht korrigiert.)
- Modifizieren Sie Ihre Implementierung aus dem vorigen Teil so, dass diese nicht rekursiv arbeitet, sondern die Rekursion durch einen Stack auflöst.
- Schreiben Sie ein WHILE-Programm, welches die Ackermann-Funktion berechnet.

(Hinweis: Implementieren Sie den Stack im WHILE-Programm mithilfe der Cantor'schen Bijektion. Sie dürfen  $\langle x, y \rangle$ ,  $p_1$  und  $p_2$  im WHILE-Programm nutzen, ohne diese selbst zu implementieren.)