Algorithmen und Berechnungskomplexität II — SS 16

Universität Bonn, Institut für Informatik, Abteilung I

2. Aufgabenblatt zur Vorlesung — **Abgabe: Mi. 27.03.** (09^{00})

- *Unter **www.i1.informatik.uni-bonn.de** werden Unterlagen und Informationen zu Vorlesung und Übung bereitgestellt.*
- *Bearbeitung und Abgabe der Übungsblätter ist in festen Gruppen von bis zu 3 Personen erlaubt.*
- *Die Abgabe muss auf dem ersten Blatt in der ersten Zeile die Namen der Studierenden und deutlich erkennbar die Nummer der Übungsgruppe enthalten. Eine Abgabe aus mehreren Blättern ist zu heften!*
- *Die Lösungen können bis **Mittwoch 09 Uhr** in den Postkasten im AVZ III eingeworfen werden.*

Aufgabe 8: Beschränktes Minimum (4 Punkte)

Für eine Funktion $f : \mathbb{N}_0^{k+1} \to \mathbb{N}_0$ definieren wir das beschränkte Minimum $\min_f : \mathbb{N}_0^{k+1} \to \mathbb{N}_0$:

$$\min_f(\mathfrak{x}, y) := \begin{cases} \min\{n \in \mathbb{N}_0 \mid n \leq y \wedge f(\mathfrak{x}, n) = 0\}, & \text{falls } n \text{ existiert} \\ 0, & \text{sonst.} \end{cases}$$

Zeigen Sie, dass für jede primitiv rekursive Funktion f gilt: $\min_f \in \mathcal{P}$.

Aufgabe 9: Diagonalisierung (4 Punkte)

Beweisen Sie mittels Diagonalisierung über $\mathcal{P}$, dass es eine intuitiv berechenbare Funktion gibt, die nicht primitiv rekursiv ist.

Hinweis: Sie dürfen annehmen, dass $\mathcal{P}$ abzählbar ist und können sich auf einstellige Funktionen beschränken.

Bitte wenden!

Aufgabe 10: **μ-rekursive Funktionen** **(4 Punkte)**

1. Sei $f : \mathbb{N}_0^2 \to \mathbb{N}_0$ mit $f(x, y) = x^2 \dot{-} y$. Was berechnet μf?
2. Sei $g : \mathbb{N}_0^2 \to \mathbb{N}_0$ mit $g(x, y) = x \dot{-} y^2$. Was berechnet μg?
3. Zeigen Sie, dass die Funktion log : $\mathbb{N}_0^2 \to \mathbb{N}_0$ mit $\log(x, y) = \lceil \log_x y \rceil$ μ-rekursiv ist.

Aufgabe 11: **LOOP-Programme** **(Präsenzaufgabe)**

In den Übungen haben wir bereits LOOP-Programme kennengelernt. Eine (intuitiv) berechenbare Funktion f heißt *LOOP-berechenbar*, falls ein LOOP-Programm formuliert werden kann, welches f berechnet. Mit $\mathcal{F}_{\text{LOOP}}$ bezeichnen wir die Klasse/Menge der LOOP-berechenbaren Funktionen.

Zeigen Sie, dass jede primitiv rekursive Funktion auch *LOOP-berechenbar* ist, d.h. es gilt $\mathcal{P} \subseteq \mathcal{F}_{\text{LOOP}}$.