Theoretische Informatik 2

Gewertete Aufgaben, Blatt 3

Abgabe: wird auf der Website bekanntgegeben
Besprechung: KW 21

---

1. (30%=10%+10%+10%) Geben Sie für folgende Funktionen LOOP-Programme an (für Aufgabe b) dürfen Sie Aufgabe a) und für Aufgabe c) dürfen Sie Aufgabe b) als Unterprogramm verwenden):

a) $\text{teilt} : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ mit

$$\text{teilt}(x, y) = \begin{cases} 0 & \text{falls } x = 0 \text{ oder } (x > 0 \text{ und } x \text{ teilt } y \text{ nicht}), \\ 1 & \text{falls } x > 0 \text{ und } x \text{ teilt } y. \end{cases}$$

b) $\text{tzahl} : \mathbb{N} \to \mathbb{N}$ mit

$$\text{tzahl}(x) = \sum_{i=0}^{x} \text{teilt}(i, x).$$

c) $\text{prim} : \mathbb{N} \to \mathbb{N}$ mit

$$\text{prim}(x) = \begin{cases} 1 & \text{falls } x \text{ eine Primzahl ist}, \\ 0 & \text{sonst}. \end{cases}$$

2. (30%=10%+10%+10%)

Für $i \geq 0$ sei LOOP_i die kleinste Menge von LOOP-Programmen, so dass folgendes gilt:

- Jedes LOOP-Programm P, in dem kein `LOOP`-Operator vorkommt, ist in LOOP_0 (Beispiel: $x := y + 5; z := x; y := z \dot{-} 1$ ist in LOOP_0).
- $\mathsf{LOOP}_i \subseteq \mathsf{LOOP}_{i+1}$.
- Falls $P \in \mathsf{LOOP}_i$ und x eine Variable ist, so ist folgendes Programm in LOOP_{i+1}:

$$\texttt{LOOP } x \texttt{ DO } P \texttt{ END}$$

Wir nennen eine Funktion $\mathbb{N}^k \to \mathbb{N}$ LOOP_i-*berechenbar*, falls f von einem LOOP_i-Programm berechenbar ist.

a) Zeigen Sie, dass es eine LOOP_1-berechenbare Funktion $f_1 : \mathbb{N} \to \mathbb{N}$ gibt, die nicht LOOP_0-berechenbar ist.

b) Zeigen Sie, dass es eine LOOP_2-berechenbare Funktion $f_2 : \mathbb{N} \to \mathbb{N}$ gibt, die nicht LOOP_1-berechenbar ist.

c) Zeigen Sie, dass es eine LOOP_3-berechenbare Funktion $f_3 : \mathbb{N} \to \mathbb{N}$ gibt, die nicht LOOP_2-berechenbar ist.

Hinweise: Zeigen Sie

a) dass für jede LOOP_0-berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$ eine Konstante $c \in \mathbb{N}$ existiert mit $f(n) \leq n+c$, es jedoch eine LOOP_1-berechenbare Funktion $f_1 : \mathbb{N} \to \mathbb{N}$ gibt mit $f_1(n) \geq 2n$ für alle $n \in \mathbb{N}$.

b) dass für jede LOOP_1-berechenbare Funktion $f : \mathbb{N} \to \mathbb{N}$ Konstanten $c_1, c_2 \in \mathbb{N}$ existieren mit $f(n) \leq c_1 \cdot n + c_2$, es jedoch eine LOOP_2-berechenbare Funktion $f_2 : \mathbb{N} \to \mathbb{N}$ gibt mit $f_2(n) \geq n^2$ für alle $n \in \mathbb{N}$.

c) Verfahren Sie wie bei a) und b) für geeignete Funktionen f und f_3.

3. (20%=10%+10%) Für die folgenden Aufgaben dürfen Sie alle Funktionen verwenden, die bereits in der Vorlesung oder in der Übung als primitiv-rekursiv nachgewiesen wurden. Zeigen Sie von folgenden Funktionen, dass sie primitiv-rekursiv sind. Wenn Sie eine Funktion durch primitive Rekursion definieren, dann geben Sie bitte explizit die Funktionen g und h wie in Beispiel 14.1 im Skript an.

a) $\mathsf{sub} : \mathbb{N}^2 \to \mathbb{N}$ mit

$$\mathsf{sub}(x, y) = \begin{cases} x - y & \text{falls } x \geq y \\ 0 & \text{sonst.} \end{cases}$$

b) $\mathsf{sdiff} : \mathbb{N}^2 \to \mathbb{N}$ mit

$$\mathsf{sdiff}(x, y) = \begin{cases} x - y & \text{falls } x \geq y \\ y - x & \text{sonst.} \end{cases}$$

4. (20%=10%+10%) Gegeben sei die partielle Funktion $f : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ mit

$$f(x, y) = \begin{cases} 1 & \text{falls } x \geq 1 \text{ und } x \text{ teilt } y \\ 0 & \text{falls } x \geq 1 \text{ und } x \text{ teilt } y \text{ nicht} \\ \text{undefiniert} & \text{sonst} \end{cases}$$

Hinweis: Die Funktion f ist bei $\{0\} \times \mathbb{N}$ undefiniert.

a) Geben Sie ein WHILE-Programm an, das f berechnet.

b) Notieren Sie f als μ-rekursive Funktion.