Ausgabedatum: 23.11.2012
Version: 2012-11-21 16:50

6. Übungsblatt zur Übung Beweistechniken, WS 12/13

Abgabe: *Bis Freitag, 30.11.2012, 12:00 Uhr, Abgabenkasten vor 48-694.*

Hinweis: Geben Sie jeweils an, welche Beweismethode Sie verwenden.

22. Aufgabe

3 + 2 Punkte

In dieser Aufgabe werden Sie schrittweise zeigen, dass es Funktionen $f : \mathbb{N} \to \mathbb{N}$ geben muss, die nicht `While`-berechenbar sind.

a) *[Gödelisierung für While-Programme]*

Zeigen Sie:
Die Menge der `While`-Programme $\mathcal{W}$ ist *abzählbar*, d. h. es gibt eine injektive Funktion $g : \mathcal{W} \to \mathbb{N}$.

Tipp: Das Problem lässt sich in drei Teilaufgaben zerlegen:

(i) Definieren Sie eine eindeutige Kodierung für `While`-Programme als Zeichenkette, also eine Funktion *string* : $\mathcal{W} \to \Sigma^\star$, wobei

$$\Sigma = \{\texttt{a}, \ldots, \texttt{z}, \texttt{0}, \texttt{1}, \ldots, \texttt{9}, \texttt{:}, \texttt{!}, \texttt{=}, \texttt{+}, \texttt{-}, \texttt{;}, \texttt{␣}\}$$

das *endliche* Alphabet der erlaubten Zeichen darstellt und $\Sigma^\star$ die Menge aller *endlichen* Zeichenketten über Σ.

Beispiel: `"x17 := x1; while x17 != 0 do x17 := x7 - 42 end"`

Ihre Definition von *string* sollte entlang des induktiven Aufbaus der `While`-Programme gemäß deren Definition verlaufen.

(ii) Zeigen Sie, dass $\Sigma^\star$ abzählbar ist für jedes endliche Σ.

(iii) A abzählbar $\wedge$ $B \subseteq A \implies B$ abzählbar.

b) *[Überabzählbarkeit von* $\mathsf{Abb}(\mathbb{N}, \mathbb{N})$*]*

Zeigen Sie:
Die Menge aller Funktionen $f : \mathbb{N} \to \mathbb{N}$ ist *überabzählbar* (d. h. nicht abzählbar).

Tipp: Verwenden Sie Diagonalisierung.

Tipp: Jede Funktion $f : \mathbb{N} \to \mathbb{N}$ korrespondiert zu einer *Folge* $(a_i)_{i \in \mathbb{N}}$ vermöge $a_i := f(i)$ und umgekehrt.

Damit gibt es weit mehr Funktionen $f : \mathbb{N} \to \mathbb{N}$ als es `While`-Programme gibt, sodass für manche Funktionen schlicht kein `While`-Programm mehr übrig sein kann. Diese Funktionen sind folglich nicht `While`-berechenbar.

23. Aufgabe

1 + 2 Punkte

a) Sei

$$\begin{aligned} b_0 &= 2, \\ b_1 &= 6, \\ b_2 &= 7 \qquad \text{und} \\ b_i &= 3 \cdot b_{i-2} + 2 \cdot b_{i-3} \qquad \text{für } i \geq 3 \,. \end{aligned}$$

Zeigen Sie **mit Induktion**: $b_n = \frac{7}{3}2^n - (-1)^n(n + \frac{1}{3})$.

b) Sei

$$\begin{aligned} a_0 &= 1, \\ a_1 &= 1 \quad \text{und} \\ a_n &= \sum_{i=1}^{\lfloor n/2 \rfloor} a_{n-2i} \qquad \text{für } n \geq 2 \,. \end{aligned}$$

Zeigen Sie **mit Induktion** für $n \geq 2$: $a_n = 2^{\lfloor n/2 \rfloor - 1}$

24. Aufgabe

2 Punkte

Wir betrachten folgenden Algorithmus zum zufälligen Durchmischen eines Arrays A.

Shuffle(A, n)

[1] Wenn $n \geq 2$, dann:

[1.1] Wähle r uniform zufällig aus $\{1, \ldots, n\}$

[1.2] Vertausche $A[n]$ und $A[r]$.

[1.3] Shuffle$(A, n-1)$

Zeigen Sie für alle $n \in \mathbb{N}$:
Gilt zu Beginn $A[i] = a_i$ für $i = 1, \ldots, n$ für *paarweise verschiedene* a_i und seien $b_i = A[i]$ für $i = 1, \ldots, n$ die Elemente von $A[1..n]$ *nach* einem Aufruf von Shuffle(A, n).
Dann ist $(b_1, \ldots, b_n)$ eine zufällige Permutation π von $\{a_1, \ldots, a_n\}$, wobei jede der $n!$ möglichen Permutationen mit *gleicher* Wahrscheinlichkeit realisiert wird, d. h.

$$\Pr\Big[\forall i \in \{1, \ldots, n\} : b_i = \pi(a_i)\Big] = \frac{1}{n!} \,.$$

Tipp: Gehen Sie induktiv vor.