

Grundlagen der Theoretischen Informatik, SS12

Übungsblatt 3 / 5 zum Nachholen der Klausurzulassung, Abgabe bis zum **Mo. 14. Mai**¹

—

Allgemeine Hinweise:

- Es wird erwartet und empfohlen, dass Sie diese Uebungen *alleine* bearbeiten.
- Alle relevanten Informationen werden auf der GTI Homepage bekanntgegeben. Hier sind auch das Skript und diese Übungsblätter zu finden.
- In jeder Aufgabe können wie immer 5 Punkte erreicht werden.
- Bitte geben Sie Ihre Lösungen in gut leserlicher und sauberer Form ab. Begründen Sie Ihre Antworten und argumentieren Sie *nachvollziehbar*.

Aufgabe 1. Sei $f : \mathbb{N} \rightarrow \mathbb{N}$ eine totale rekursive Funktion. Zeigen Sie, dass auch die folgenden beiden Funktionen total und rekursiv sind:

- Die Funktion $g : \mathbb{N} \rightarrow \mathbb{N}$, die jedem $n \in \mathbb{N}$ das Maximum der Zahlen $f(0), \dots, f(n)$ zuordnet.
- Die Funktion $h : \mathbb{N} \rightarrow \mathbb{N}$, die jedem $n \in \mathbb{N}$ eine Zahl m zuordnet, so dass $f(m)$ das Maximum der Zahlen $f(0), \dots, f(n)$ ist.

Benutzen Sie hierbei keine Sätze oder Lemmata aus der Vorlesung bis auf Satz 3.2.49 (der mit den vielen Funktionen, die alle primitiv rekursiv sind).

Aufgabe 2. Geben Sie eine Turingmaschine an, die, bei Eingabe eines Wortes w über einem beliebigen Eingabealphabet Σ , das Wort spiegelt (umdreht) und dann hält.
(Also z.B. wird aus 'abcdef' das Wort 'fedcba').

¹ Abgabe am Besten *persönlich* bei Christian Uhrhan (EN-B 0125), oder im Sekretariat der theoretischen Informatik (EN-B 0121).

Aufgabe 3. Die Sprache WHILE ist vom Umfang her sehr klein gehalten, damit man Aussagen über diese Sprache leichter beweisen kann. Normale Programmiersprachen enthalten eine Vielzahl von weiteren Konstruktionen. Diese machen zwar das Leben des Programmierers einfacher, erweitern aber normalerweise nicht die Mächtigkeit.

Beschreiben Sie, wie sich die folgende Typ Schleife in WHILEübersetzen lässt: Eine normale “for”-Schleife, bei der ein Programmblock genau n -mal durchlaufen wird, wobei der Wert n in einer Variable stehen kann.

Aufgabe 4. Um zum Begriff der Turing-berechenbaren Funktion und der WHILE-berechenbaren Funktion zu kommen, war es notwendig die natürlichen Zahlen mittels geeigneter Kodierungen in die Datentypen Wörter und Binärbäume einzubetten. Ebenso war es in den Beweisen, dass die Klasse der Turing-berechenbaren Funktionen und WHILE-berechenbaren Funktionen zulässige Programmiersysteme besitzen, notwendig Binärbäume und Wörter durch natürliche Zahlen zu kodieren.

Zeigen Sie, dass es auch geeignete Kodierungen von Binärbäumen in Wörter und von Wörtern in Binärbäumen gibt.