Technische Universität München
Institut für Informatik
PD Dr. Markus Holzer
Dipl.-Inf. Michael Tautschnig

WS 2007/2008
Übungsblatt 1
Mittwoch, 24. Oktober 2007

Übung zur Vorlesung Automaten, Formale Sprachen und Berechenbarkeit

Lösungshinweise

Aufgabe 1 Wiederholung

Fragen?

Aufgabe 2 *Kleensche Hülle*

(i) Falsch; Gegenbeispiel: $A = \{\lambda\}$.

(ii) Wahr; folgt unmittelbar aus der Definition.

(iii) Falsch; man betrachte etwa $A = \{a\}$, $B = \{aa\}$.

(iv) Falsch; man betrachte etwa $A = \{a\}$, $B = \{b\}$.

(v) Falsch; man betrachte etwa $A = B = \{a\}$.

Aufgabe 3 *uvw-Theorem*

Wir nehmen an, daß L akzeptierbar ist. Sei n also die Konstante aus dem uvw-Theorem. Wähle das Wort $z = a^n b^{n+1} c \in L$. Es gilt $|z| \geq n$. Nun überprüfen wir, ob alle Zerlegungen des Wortes z, die dem uvw-Theorem genügen, zum Widerspruch geführt werden können. Auf Grund der Bedinungen des uvw-Theorems sei $u = a^i$, $v = a^j$ mit $j \geq 1$ und $i + j \leq n$, und $w = a^{n-i-j} b^{n+1} c$. Dann gilt $z = uvw$ und $uv^2w = a^i a^{2j} a^{n-i-j} b^{n+1} c = a^{n+j} b^{n+1} c \notin L$, da $n + j \geq n + 1$ gilt.

Aufgabe 4 *WHILE-Programme*

(i) Das folgende WHILE-Programm berechnet $f(x) = 2x$:

$Y := 0$
$Z := 0$
/* Iteratives Zählen in Z */
while $X \neq Y$ **do**
  $Z := succ(Z)$
  $Z := succ(Z)$
  $X := pred(X)$
od
/* Umkopieren der Variable Z nach X */
$X := succ(Z)$
$X := pred(X)$

(ii) Da nur eine Variable im WHILE-Programm erlaubt ist, können wir ohne Beschränkung der Allgemeinheit nur Programme betrachten, in denen *keine* WHILE-Anweisungen enthalten sind (da die einzig mögliche Schleifenbedingung $X \neq X$ *nie* erfüllt sein kann, können wir eventuell vorhandene Schleifen gänzlich aus dem Programmcode streichen). In einem derartigen WHILE-Programm können die Werte der Variablen daher lediglich einfache *succ*-Anweisungen erhöht werden. Insbesondere ist daher der maximale Wert der Programmvariable während des Programmablaufs durch $x + n$ beschränkt, wobei x die Eingabe und n die Länge des Programms repräsentiert. Die Funktion $f(x) = 2x$ wächst jedoch für steigendes x asymptotisch über die Schranke $x + n$; kein WHILE-Programm mit nur einer Variable kann daher $f(x)$ berechnen. Ein formaler Beweis kann mittels Induktion über die Programmlänge geführt werden.