

BuK Abgabe 5 | Gruppe 17

Malte Meng (354529) , Charel Ernster (318949), Sebastian Witt (354738)

November 30, 2016

1 Aufgabe 5.1

(a). Gegeben ist:

$L_1 \leq L_2$ und $L_2 \leq L_3 \rightarrow \exists f_1, f_2$ mit:

$f_{1|2}$ bildet alle Ja/Nein-Instanzen von $L_{1|2}$ auf Ja/Nein-Instanzen von $L_{2|3}$.

Somit gibt es die Bildmenge M_1 der Ja/Nein-Instanzen in L_2 von der Abbildung f_1 ($L_1 \xrightarrow{f_1} L_2$).

$M_1 \subseteq L_2 \Rightarrow$

$\exists M_2$ mit $M_2 = f_2(M_1)$ und $M_2 \subseteq L_3 \Rightarrow$

$\exists f_3$ mit $f_3 = L_1 \xrightarrow{L_1 \rightarrow M_1 \rightarrow M_2} M_2$ mit $M_2 \subseteq L_3 \Rightarrow$

$f_3 := L_1 \rightarrow L_3$ mit f_3 bildet alle Ja/Nein-Instanzen von L_1 auf L_3 ab. Die Korrektheit der Funktionen bleibt wie die Ursprünglichen f_1, f_2 . Somit gilt $L_1 \leq L_3$ für beliebige L_1, L_2, L_3 mit $L_1 \leq L_2$ und $L_2 \leq L_3$. Das Reduktionskonzept ist also transitiv. \square

(b). $L_1 \leq L_2 \Rightarrow \exists f = \begin{cases} L_{ja} \rightarrow M_{ja} \\ L_{nein} \rightarrow M_{nein} \end{cases}$
 $L_{ja} = \overline{L_{nein}}, L_{nein} = \overline{L_{ja}}, M_{ja} = \overline{M_{nein}}, M_{nein} = \overline{M_{ja}} \Rightarrow$
 $\exists \overline{f} = \begin{cases} \overline{L_{nein}} \rightarrow \overline{M_{nein}} \\ \overline{L_{ja}} \rightarrow \overline{M_{ja}} \end{cases} \Rightarrow \overline{L_1} \leq \overline{L_2}$
Somit gilt $L_1 \leq L_2 \Rightarrow \overline{L_1} \leq \overline{L_2} \square$

2 Aufgabe 5.3

(a). Die Sprache A_{62} ist rekursiv aufzählbar aber nicht rekursiv.
Beweis in zwei Schritten durch reduktion von H_ϵ auf A_{62} :

I. A_{62} ist nicht rekursiv:

Es gibt eine berechenbare Funktion f mit folgenden Eigenschaften:

- $\langle M \rangle$ ist keine gültige Gödelnummer $f(\langle M \rangle) = \bar{w}$ mit $\bar{w} \in \overline{H_\epsilon}$
- Falls $w = \langle M \rangle$ für eine TM M , so sei $f(w)$ die Gödelnummer einer TM M_{neu} mit folgenden Eigenschaften:

M_{neu} prüft Eingabelänge $l \begin{cases} l > 62 & | \text{ verwerfe} \\ l \leq 62 & | \text{ verwerfe die Eingabe und Simuliere } M \text{ mit Eingabe } \epsilon \end{cases}$

f bildet H_ϵ korrekt auf A_{62} ab:

Falls w keine Gödelnummer ist, so ist die Korrektheit klar, denn in diesem Fall gilt $w \notin H_\epsilon$ und $f(w) \notin A_{62}$

Sei $w = \langle M \rangle$ für eine TM M und sei $f(w) = \langle M_{neu} \rangle$

Es gilt:

$w \in H_\epsilon \Rightarrow M$ hält auf ϵ

$\Rightarrow \langle M_{neu} \rangle$ hält auf jeder Eingabe mit $l \leq 62$

$\Rightarrow \langle M_{neu} \rangle$ hält auf mindestens zwei Wörtern der Länge höchstens 62

$\Rightarrow \langle M_{neu} \rangle \in A_{62}$

$\Rightarrow f(w) \in A_{62}$

$w \notin H_\epsilon \Rightarrow M$ hält nicht auf ϵ

$\Rightarrow \langle M_{neu} \rangle$ hält auf keiner Eingabe

$\Rightarrow \langle M_{neu} \rangle \notin A_{62}$

$\Rightarrow f(w) \notin A_{62}$

Daher ist $H_\epsilon \leq A_{62} \Rightarrow A_{62}$ ist nicht Rekursiv.

II. A_{62} ist aufzählbar:

Es gibt eine TM M die A_{62} erkennt mit folgenden Eigenschaften:

- M Simuliert alle Gödelnummern parallel. Also jeweils einen Schritt auf allen Nummern pro Iteration.
- Die Simulation funktioniert indem jeweils wieder ein Schritt auf jedem Wort mit $l \leq 62$ Simuliert wird.
- Werden dabei zwei Wörter Akzeptiert "druckt" M die korrespondierende Gödelnummer.

(b). Beweis dass B_1 nicht rekursiv aufzählbar ist:

$$\overline{B_1} = \{\langle M \rangle \mid M \text{ akzeptiert kein Wort}\} \geq \{\langle M \rangle \mid M \text{ hält auf keinem Wort}\} = \overline{H_{all}}$$

$$\overline{B_1} \geq \overline{H_{all}}$$

$$H_{all} \text{ ist nicht rekursiv aufzählbar} \Rightarrow \overline{H_{all}} \text{ ist nicht rekursiv aufzählbar}$$

$$\Rightarrow \overline{B_1} \text{ ist nicht rekursiv aufzählbar}$$

$$\Rightarrow B_1 \text{ ist nicht rekursiv aufzählbar}$$

$$\Rightarrow B_1 \text{ ist nicht rekursiv aufzählbar} \quad \square$$

3 Aufgabe 5.4

Es existiert eine berechenbare Funktion $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$, sodass $x \in \bar{H}_\epsilon \Leftrightarrow f(x) \in L_{101}$ gilt.

Wir definieren die Funktion f :

$$w \mapsto \begin{cases} \langle M' \rangle w = \langle M \rangle \\ \epsilon w \neq \langle M \rangle \end{cases}$$

wobei M' die TM ist, die wie folgt arbeitet:

- M' simuliert die ersten i Schritte von M auf der Eingabe ϵ , erst ab $i = 3$.
- Falls M innerhalb dieser i Schritte hält, dann geht M' in die Endlosschleife, ansonsten hält M'

Berechenbarkeit:

Offensichtlich ist die Funktion berechenbar.

Korrektheit:

Fall 1: Wenn $w \neq \langle M \rangle$, dann ist $w \notin \bar{H}_\epsilon$, und damit $f(w) = \epsilon \notin L_{101}$

Fall 2:

$w \in \bar{H}_\epsilon$

$\Rightarrow M$ hält nicht auf der Eingabe ϵ

\Rightarrow Für alle i angefangen bei $i = 3$: M hält nicht innerhalb von i Schritten auf ϵ .

\Rightarrow Für alle i angefangen bei $i = 3$: M' hält auf all Eingaben der Länge i .

$\Rightarrow M'$ hält auf jeder Eingabe.

$\Rightarrow M'$ hält auf jeder Eingabe die mit 101 beginnt.

$\Rightarrow f(w) = \langle M' \rangle \in L_{101}$

$w \notin \bar{H}_\epsilon$

$\Rightarrow M$ hält auf der Eingabe ϵ

\Rightarrow Es gibt ein i , beginnend bei $i = 3$: M hält innerhalb von i Schritten auf ϵ

\Rightarrow Es gibt ein i , beginnend bei $i = 3$: M' hält auf keiner Eingabe der Länge mindestens i .

$\Rightarrow M'$ hält nicht auf jeder Eingabe die mit 101 anfängt.

$\Rightarrow f(w) \notin L_{101}$

Damit haben wir gezeigt dass $\bar{H}_\epsilon \leq L_{101}$ gilt.