Gruppe $\boxed{\textbf{17}}$
Berechenbarkeit & Komplexität

Philipp Hochmann, 356148
Felix Kiunke, 357322

Blatt 5

Aufgabe 5.1

(a)

$$
\begin{aligned}
& L_1 \leq L_2 \wedge L_2 \leq L_3 \\
\Leftrightarrow \quad & (\exists f : x \in L_1 \Leftrightarrow f(x) \in L_2) \wedge (\exists g : x \in L_2 \Leftrightarrow g(x) \in L_3)
\end{aligned}
$$

Also existiert eine solche Funktion $h(x) = f(g(x))$ wenn f und g existieren, für die gilt:
$x \in L_1 \Leftrightarrow f(x) \in L_2 \Leftrightarrow h(x) \in L_3$

(b)

zu zeigen: $\quad (L_1 \leq L_2) \implies (\overline{L_1} \leq \overline{L_2})$

$$
\begin{aligned}
& L_1 \leq L_2 \\
\Leftrightarrow \quad & \exists f : x \in L_1 \Leftrightarrow f(x) \in L_2 \\
\implies \quad & \exists f : \neg(x \in L_1) \Leftrightarrow \neg(f(x) \in L_2) \\
\implies \quad & \exists f : x \notin L_1 \Leftrightarrow f(x) \notin L_2 \\
\implies \quad & \exists f : x \in \overline{L_1} \Leftrightarrow f(x) \in \overline{L_2} \\
\implies \quad & \overline{L_1} \leq \overline{L_2}
\end{aligned}
$$

Aufgabe 5.2

Zu widerlegen:

$$\overline{\mathbf{H}_\varepsilon} \leq \mathbf{H}_\varepsilon \tag{1}$$

$$\mathbf{H}_\varepsilon \leq \overline{\mathbf{H}_\varepsilon} \tag{2}$$

Aus der Vorlesung ist bekannt, dass $\mathbf{H}_\varepsilon$ rekursiv aufzählbar ist und dass $\overline{\mathbf{H}_\varepsilon}$ *nicht* rekursiv aufzählbar ist. Des Weiteren wurde gezeigt, dass

$$L_1 \leq L_2 \text{ und } L_2 \text{ rekursiv aufzählbar} \implies L_1 \text{ rekursiv aufzählbar.}$$

Damit ist die (1) widerlegt: $\mathbf{H}_\varepsilon$ ist rekursiv aufzählbar, $\overline{\mathbf{H}_\varepsilon}$ allerdings nicht, was einen Widerspruch zu dieser Aussage darstellen würde. Damit kann (1) nicht gelten. $\square$

Die Widerlegung von (2) ist damit sehr einfach: Da nach 5.1

$$\mathbf{H}_\varepsilon \leq \overline{\mathbf{H}_\varepsilon} \implies \overline{\mathbf{H}_\varepsilon} \leq \mathbf{H}_\varepsilon$$

gelten müsste und letztere Aussage oben bereits widerlegt wurde, kann auch (2) nicht gelten. □

Aufgabe 5.3

(a)

Gegeben sei die Sprache

$$A_{62} = \{\langle M \rangle \mid M \text{ akzeptiert mindestens zwei Wörter der Länge höchstens } 62\}$$

Diese Sprache ist nicht entscheidbar, also **nicht rekursiv**.

Beweis. mit dem Satz von Rice. Sei

$$\mathcal{S} = \{f_M \mid \exists (w_1, w_2) \in \{0,1\}^* \times \{0,1\}^*, |w_1| \leq 62, |w_2| \leq 62, w_1 \neq w_2 : f_M(w_1) = 1 \wedge f_M(w_2) = 1\}$$

Dann ist $\mathcal{S}$ nicht leer, denn eine passende Funktion f_M existiert, zum Beispiel die Funktion zur Turingmaschine, die alle Wörter aus $\{0,1\}^*$ akzeptiert. Außerdem ist $\mathcal{S} \neq \mathcal{R}$, da nicht alle Turingmaschinen die gewünschte Bedingung erfüllen.

Damit ist

$$\begin{aligned} L(\mathcal{S}) &= \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\} \\ &= \{\langle M \rangle \mid M \text{ akzeptiert mindestens zwei Wörter der Länge höchstens } 62\} \\ &= A_{62} \end{aligned}$$

und somit ist A_{62} nach Satz von Rice nicht entscheidbar. □

Allerdings ist A_{62} **rekursiv aufzählbar**.

Beweis. Wir konstruieren einen Aufzähler A zur TM M, der auf zwei Bändern für $i = 1, 2, 3, \ldots$ jeweils i Schritte für alle Wörter $\{w_1, \ldots, w_{\min\{i,62\}}\}$ simuliert. Dabei sind die für die beiden Bänder gewählten Eingabewörter aus dieser Menge unabhängig voneinander. Die Kombinationen sind analog zum Diagonalverfahren abzählbar. Auf beiden Bändern wird jeweils die Turingmaschine M simuliert. Wenn in einem Schritt beide Bänder akzeptieren, wird auf dem Drucker $\langle M \rangle$ ausgegeben.

Anschließend konstruieren wir einen weiteren Aufzähler A', der für $i' = 1, 2, 3, \ldots$ jeweils i' Schritte des obigen Aufzählers A für alle möglichen Gödelnummern der Länge i' ausführt. Jede Ausgabe von A wird auch von A' ausgegeben. Damit gibt A' alle Gödelnummern aus der Sprache A_{62} aus.

Dieser Aufzähler gibt dann genau A_{62} aus. Da die Konstruktion eines Aufzählers möglich ist, ist A_{62} rekursiv aufzählbar. □

(b)

Gegeben sei die Sprache

$$B_1 = \{\langle M \rangle \mid M \text{ akzeptiert mindestens ein Wort}\}$$

Diese Sprache ist ebenfalls **nicht rekursiv**.

Beweis. mit dem Satz von Rice. Sei

$$\mathcal{S} = \{f_M \mid \exists w \in \{0,1\}^* : f_M(w) = 1\}$$

Dann ist $\mathcal{S}$ nicht leer, denn eine passende Funktion f_M existiert, zum Beispiel die Funktion zur Turingmaschine, die alle Wörter aus $\{0,1\}^*$ akzeptiert. Außerdem ist $\mathcal{S} \neq \mathcal{R}$, da z.B. die Funktion zur Turingmaschine, die alle Wörter verwirft, nicht in $\mathcal{S}$ ist.

Damit ist

$$\begin{aligned} L(\mathcal{S}) &= \{\langle M \rangle \mid M \text{ berechnet eine Funktion aus } \mathcal{S}\} \\ &= \{\langle M \rangle \mid M \text{ akzeptiert mindestens ein Wort}\} \\ &= B_1 \end{aligned}$$

und somit ist B_1 nach Satz von Rice nicht entscheidbar. □

Allerdings ist auch B_1 **rekursiv aufzählbar**.

Beweis. Wir konstruieren wieder einen Aufzähler, der für $i = 1, 2, 3, \ldots$ alle Wörter $\{w_1, \ldots, w_i\}$ für i Schritte simuliert, wobei diese Wörter als Gödelnummern mit anschließenden Eingabewörtern betrachtet werden und ungültige Gödelnummern zuvor verworfen bzw. ignoriert werden. Wenn eine dieser Simulationen akzeptiert, wird die entsprechende Gödelnummer auf dem Drucker ausgegeben (wobei das dazugehörige Eingabewort natürlich nicht mit ausgegeben wird).

Dieser Aufzähler gibt also alle Gödelnummern aus B_1, aus, also genau die Sprache B_1. Da ein Aufzähler existiert, ist B_1 offentsichtlich rekursiv aufzählbar. □

Aufgabe 5.4

f verwirft, wenn die Eingabe $w = \langle M \rangle$ keine gültige Gödelnummer ist. Sonst gibt f die Gödelnummer $\langle M^* \rangle$ aus. M^* hat folgende Eigenschaften: wenn die Eingabe w nicht mit 101 beginnt, wird verworfen. Sollte w mit 101 beginnen, wird M auf dem leeren Wort ε simuliert. Mit jedem Schritt dieser Simulation wird das nächste Wort von der Gestalt $u = 101x$ mit $x \in \Sigma^*$ abgezählt und akzeptiert, wenn dieses die Eingabe ist. Wenn die Simulation von M abgeschlossen ist wird in einen nicht-terminierenden Kreislauf gegangen. Diese Funktion ist *trivialerweise* berechenbar.

Beweis der Korrektheit:
$\langle M \rangle \notin \overline{H_\varepsilon} \implies M$hält auf $\varepsilon \implies M^*$ hält auf keinem w, welches mit 101 beginnt $\implies f(\langle M \rangle) \notin L_{101}$

$\langle M \rangle \in \overline{H_\varepsilon} \implies M$ hält nicht auf $\varepsilon \implies M^*$ simuliert abzählbar-unendlich viele Schritte lang $\implies M^*$ akzeptiert jedes w, welches mit 101 beginnt. Insbesondere hält M^* dann. $\implies f(\langle M \rangle) \in L_{101}$.