# Theoretische Informatik: Blatt 5

Abgabe bis 30. Oktober 2015
Assistent: Sascha Krug, CHN D 42

**Linus Fessler, Markus Hauptner, Philipp Schimmelfennig**

# Aufgabe 13

(a) Seien die Zimmer mit 1, 2, 3, ... durchnummeriert und das Tupel $(i, j)$ beschreibe den $j$-ten Gast aus dem $i$-ten Bus ($i \in \{1, 2, 3\}$, $j \in \mathbb{N}_0$). Zuerst zieht jeder Gast im Hotel mit Zimmernummer $k$ in Zimmernummer $6k$. $(0, j)$ beschreibe die bereits bestehenden Gäste. Nun weisen wir den Gästen aus den Bussen folgendermassen ihre Zimmer zu:

- Gast $(1, j)$ weisen wir Zimmer $6j + 1$ (ungerade) zu.
- Dem Gast $(2, j)$ weisen wir Zimmer $\begin{cases} 6j + 2, & \text{für j gerade} \\ 6j + 3, & \text{für j ungerade} \end{cases}$ zu.
- Dem Gast $(3, j)$ weisen wir Zimmer $6j + 4$ zu.

Damit wiederholt sich das Muster $(1, j)$, $(2, j)$, $(2, j)$, $(3, j)$, leer, $(0, j)$ für aufsteigende Zimmernummern. Gäste aus Bus 1 sind in ungeraden Zimmernummern, Gäste aus Bus 2 immer paarweise nebeneinander und Gäste aus Bus 3 werden auch untergebracht.

(b) Zuerst weisen wir den Bussen eine Reihenfolge zu, je nach dem, wann sie angekommen sind. Der erste Bus, der ankommt, bekommt Nummer 1, usw. Wir beschreiben wieder den j-ten Gast aus Bus i mit dem Tupel $(i, j)$, mit $i, j \in \mathbb{N}$. Dann weisen wir dem Gast $(i, j)$ die Zimmernummer $p_i^j$ zu, wobei $p$ die $i$-te Primzahl ist (nach Grösse geordnet). So hat jeder Gast eindeutig ein Zimmer, weil jede gegebene Primzahl und ihre Potenzen teilerfremd zu jeder anderen Primzahl und ihren Potenzen ist (für positive Exponenten) und weil es abzählbar unendlich viele Primzahlen gibt.

# Aufgabe 14

Sei $A = [d_{ij}]_{i,j=1,\ldots,\infty}$ eine unendliche Boole'sche Matrix mit $d_{ij} = 1 \iff M_i$ akzeptiert $w_j$. Dann ist

$$L_1 = \{w \in \{0, 1\}^* \mid w = w_{i^2} \text{ für ein } i \in \mathbb{N} \text{ und } d_{ii^2} = 0\}$$

und

$$L_2 = \{w \in \{0, 1\}^* \mid w = w_i \text{ für ein } i \in \mathbb{N} \text{ und } d_{i^2 i} = 0\}.$$

Wir beweisen $L_2 \in \mathcal{L}_{RE}$ indirekt. Sei $L_2 \in \mathcal{L}_{RE}$. Dann ist $L_2 = L(M)$ für eine TM $M$. Weil $M$ eine der Turingmaschinen in der Nummerierung aller Turingmaschinen sein muss, existiert ein $i \in \mathbb{N} - \{0\}$, so dass $M = M_{i^2}$. Aber $L_2$ kann nicht gleich $L(M_{i^2})$ sein, weil folgende Äquivalenz gilt:

$$w_i \in L_2 \iff d_{i^2 i} = 0 \iff w_i \notin L(M_{i^2}),$$

d.h., $w_i$ ist in genau einer der Sprachen $L_2$ oder $L(M_{i^2})$. Also ist $L_2 \notin \mathcal{L}_{RE}$.

Man kann keinen analogen Beweis für $L_1$ führen, weil der Beweis nur für die Wörter $w_{i^2}$ möglich ist (wegen $d_{ii^2} = 0$). Zum Beispiel ist $w_2$ sowohl in $L_1$ als auch in $L(M_2)$. Die Äquivalenz:

$$w_{i^2} \in L_1 \iff d_{ii^2} = 0 \iff w_{i^2} \notin L(M_i),$$

gilt also nicht allgemein.

# Aufgabe 15

(a) Wir definieren $(L_{\text{diag'}})^{\complement} = (L_2)^{\complement}$ wie folgt:

$$(L_{\text{diag'}})^{\complement} = \{w \in \{0,1\}^* \mid w = w_i \text{ für ein } i \in \mathbb{N} \text{ und } M_{i^2} \text{ akzeptiert } w\}.$$

(b) Wir zeigen $(L_{\text{diag'}})^{\complement} \leq_R L_U$. Dafür beschreiben wir einen Algorithmus $B$, der mit $A$ als Teilprogramm die Sprache $(L_{\text{diag'}})^{\complement}$ entscheidet:

Algorithmums $B$ für $L_U$