Theoretische Informatik 2 | **Sommer 2026**
Dr. Arne Schmidt | **Abgabe:** 08.06.
Besprechung: ab 17.06.

Übungsblatt 3

Abgabe der Lösungen bis zum 08.06. um 13:00 Uhr in den Hausaufgabenkasten der Algorithmik zwischen Raum IZ 337 und IZ 338. Beschrifte deine Abgabe mit Namen, Matrikelnummer. Achtet darauf, die Abgabe in das mit der korrekten Übungsgruppe beschriftete Fach zu legen. Andernfalls wird die Abgabe unter Umständen nicht gewertet.

Pflichtaufgabe 1 (Unentscheidbarkeit): **(4+4 Punkte)**
Sei $\mathcal{L} = \left\{ w_1\#w_2 \in \{0,1\}^*\#\{0,1\}^* \mid \mathcal{L}(M_{w_2}) \nsubseteq \overline{\mathcal{L}(M_{w_1})} \right\}$. Dabei ist für eine Sprache $\mathcal{L}'$ das Komplement $\overline{\mathcal{L}'} = \Sigma^* \setminus \mathcal{L}'$.

a) Zeige: $\mathcal{L}$ ist semi-entscheidbar.

b) Zeige ohne Verwendung des Satzes von Rice: $\mathcal{L}$ ist nicht entscheidbar.

Pflichtaufgabe 2 (PCP-Varianten): **(2+2+2 Punkte)**
Betrachte Abwandlungen des PCP.

a) Im `Unary PCP` sind Paare $(x_1, y_1), \ldots, (x_m, y_m)$ gegeben mit $x_i, y_i \in \{1\}^*$. Gibt es eine endliche Sequenz $i_1, \ldots, i_n$ mit $x_{i_1}x_{i_2}\cdots x_{i_n} = y_{i_1}y_{i_2}\cdots y_{i_n}$?

   Zeige oder widerlege: `Unary PCP` ist unentscheidbar.

b) Im $\texttt{PCP}_{\geq k}$ sind Paare $(x_1, y_1), \ldots, (x_m, y_m)$ über Σ^*. Gibt es eine endliche Sequenz $i_1, \ldots, i_n$ mit $x_{i_1}x_{i_2}\cdots x_{i_n} = y_{i_1}y_{i_2}\cdots y_{i_n}$ und $n \geq k$.

   Zeige oder widerlege: $\texttt{PCP}_{\geq k}$ ist unentscheidbar.

c) Im `Last-PCP` sind Paare $(x_1, y_1), \ldots, (x_m, y_m)$ über Σ^*. Gibt es eine endliche Sequenz $i_1, \ldots, i_n$ mit $x_{i_1}x_{i_2}\cdots x_{i_n} = y_{i_1}y_{i_2}\cdots y_{i_n}$ und $i_n = m$.

   Zeige oder widerlege: `Last-PCP` ist unentscheidbar.

Pflichtaufgabe 3 (Satz von Rice): **(6 Punkte)**
Beweise mit dem Satz von Rice, ob folgende Sprachen unentscheidbar sind. Begründe dazu, ob die Eigenschaft trivial (durch eine formale Begründung) oder nicht-trivial (durch Angabe einer positiven und negativen Sprache) ist. Begründe mit Hilfe einer positiven und sprach-äquivalenten negativen Turing-Maschine, falls der Satz nicht angewendet werden kann.

a) $\mathcal{L}_1 = \{w \mid M_w \text{ kehrt vor dem Akzeptieren immer zum linken Ende der Eingabe zurück.}\}$

b) $\mathcal{L}_2 = \{w \mid M_w \text{ lehnt jedes Wort mit weniger als 50 Buchstaben ab.}\}$

c) $\mathcal{L}_3 = \{w \mid \forall x \in \{0,1\}^* : x \in \mathcal{L}(M_w) \wedge x^{reverse} \notin \mathcal{L}(M_w)\}$

   Dabei entsteht $x^{reverse} = x_n x_{n-1} \ldots x_1$ durch Umkehrung des Wortes $x = x_1 \ldots x_{n-1}x_n$.