René Maseli
Prof. Dr. Roland Meyer

Theoretical Computer Science 2
Exercise sheet 3

TU Braunschweig
Summer semester 2023

Release: 05/16/2023 Due: 05/24/2023, 18:00

Hand in your answers to the Vips directory of the Stud.IP course until wednesday, 24.05.2023 06:30 pm. You should provide your answers either directly as PDF file or as a readable scan or photo of your handwritten notes. Submit your results as a group of four. On the front page, state the **degree programme, name, surname and student id** of each member of your group.

Remark: On May 19th, exercise 2 a) has been altered, because in that form, it was unsolvable.

Homework Exercise 1: Computability [10 points]

A (deterministic, 1-tape-) Turing machine is called **regular**, if all its transitions are either

- halting ($\delta(q, a) = \langle q, a, N \rangle$) or
- moving right and into a non-accepting state ($\delta(q, a) \in (Q \setminus Q_F) \times \{a\} \times \{R\}$ for $a \in \Sigma$) or
- accept at the end of the input ($\delta(q, \sqcup) \in Q_F \times \{\sqcup\} \times \{N\}$).

a) [6 points] Given the following total function shortest : TM $\to \mathbb{N} \cup \{\bot\}$.

$$\text{shortest}(M) = \begin{cases} v & \text{if } M \text{ is regular and } v \text{ is the only shortest word in } \mathcal{L}(M). \\ \bot & \text{otherwise.} \end{cases}$$

Show that the function is computable by constructing an algorithm that computes shortest.

Nonregular-Computation	
Given:	A Turing machine M with input alphabet $\{0, 1\}$ and an input word $x \in \{0, 1\}^*$.
Question:	Does M at least once move left in the computation on x?

b) [4 points] Use the universal Turing machine U from the lecture as a subroutine to show that Nonregular-Computation is semi-decidable.
Keep in mind, that U is specialized solely on the acceptance condition.

Hinweis: Es lässt sich sogar zeigen, dass Nonregular-Computation co-semi-entscheidbar ist.

Homework Exercise 2: Undecidability [7 points]

Disprove the decidability of the following problems.

1-Bounded-Acceptance	
Given:	Eine Turing-Maschine M mit Eingabealphabet $\{0, 1\}$.
Question:	Akzeptiert M nicht mehr als ein Wort?

a) [3 points] Show that 1-Bounded-Acceptance is not semi-decidable, by reducing EMPTINESS to the language BACC := $\{w \in \{0,1\}^* \mid |\mathcal{L}(M_w)| < 2\}$.

Remark: EMPTINESS := $\{w \in \{0,1\}^* \mid \mathcal{L}(M_w) = \varnothing\}$ is not semi-decidable, but co-semi-decidable. This will be further discussed on monday.

Safe-Leading	
Given:	Two Turing machines M_0 and M_1 with input alphabet $\{0,1\}$ and an input word $x \in \{0,1\}^*$.
Question:	Is M_1 always in a state with equal or higher index than M_0, if computed simultaneously on x?

b) [4 points] Show that Safe-Leading is not semi-decidable.

Exercise 3:

In der Vorlesung haben Sie gesehen, dass das Akzeptanzproblem unentscheidbar ist. Nun wollen wir ein Problem kennenlernen, das nicht einmal semi-entscheidbar ist.

Non-Self-Acceptance	
Given:	Turing-Maschine M mit Eingabealphabet $\{0,1\}$
Question:	Lehnt M ihre eigene Kodierung $\langle M \rangle$ ab?

Als Sprache lässt sich das Problem wie folgt auffassen:

$$L_{NSACC} = \left\{ w \in \{0,1\}^* \;\middle|\; w \text{ kodiert } M_w \text{ und } w \notin \mathcal{L}(M_w) \right\}.$$

Beweisen Sie unter Verwendung einer Diagonalisierung, dass L_{NSACC} nicht semi-entscheidbar ist: Nehmen Sie also die Existenz einer TM M mit $\mathcal{L}(M) = L_{NSACC}$ an, betrachten Sie ihre Kodierung $\langle M \rangle$ und leiten Sie einen Widerspruch her.

Zeigen Sie, dass L_{NSACC} co-semi-entscheidbar ist, d.h. beweisen Sie, dass das Komplementproblem semi-entscheidbar ist.

Non-ε-Acceptance	
Given:	Turing-Maschine M mit Eingabealphabet $\{0,1\}$
Question:	Lehnt M die Eingabe ε ab?

Das Problem sei aufgefasst als die Sprache:

$$L_{N\varepsilon ACC} = \left\{ w \in \{0,1\}^* \;\middle|\; w \text{ kodiert } M_w \text{ und } \varepsilon \notin \mathcal{L}(M_w) \right\}$$

Zeigen Sie $L_{NSACC} \leq L_{N\varepsilon ACC}$. Was können Sie daraus in Kombination mit den anderen beiden Teilaufgaben schließen?

Exercise 4:

Das Universalitätsproblem ist das folgende Entscheidungsproblem.

Universality	
Given:	Turing-Maschine M mit Eingabealphabet $\{0, 1\}$
Question:	Akzeptiert M alle Eingaben?

Dieses Problem lässt sich als folgende Sprache auffassen:

$$L_{\text{Universality}} = \left\{ w \in \{0, 1\}^* \mid \mathcal{L}(M_w) = \{0, 1\}^* \right\} .$$

Zeigen Sie, dass $L_{\text{Universality}}$ nicht co-semi-entscheidbar ist. Betrachten Sie dazu das Komplementproblem $\overline{L_{\text{Universality}}} = \{0, 1\}^* \setminus L_{\text{Universality}}$. Beschreiben Sie zunächst diese Sprache. Was bedeutet es für die Maschine M_w, wenn $w \in \overline{L_{\text{Universality}}}$ gilt?

Reduzieren Sie ein Problem, von dem bekannt ist, dass es nicht semi-entscheidbar ist, auf $\overline{L_{\text{Universality}}}$.

Hinweis. Die uns bereits bekannten Halte- und Akzeptanzprobleme sind semi-entscheidbar, aber nicht entscheidbar. Gemäß Theorem 3.20 im Skript sind sie also nicht co-semi-entscheidbar.

Eine akzeptierende Berechnung einer DTM A lässt sich als Wort kodieren, indem man die Konfigurationen der Berechnung mit #-Symbolen getrennt hintereinander schreibt, also $c_0 \# c_1 \# c_2 \# \ldots \# c_k$. Geben Sie an, wie man einen Entscheider B konstruiert, welcher die Kodierung einer solchen Berechnung genau dann akzeptiert, wenn es sich **nicht** um eine akzeptierende Berechnung von A zur Eingabe ε handelt.

Hinweis. Eine Eingabe für B kann entweder keine valide Kodierung einer Berechnung, oder eine akzeptierende, oder eine nicht-akzeptierende Berechnung sein.

Beweisen Sie, dass das Universalitätsproblem nicht semi-entscheidbar ist. Führen Sie dazu eine Reduktion mit Hilfe der vorherigen Teilaufgabe durch.

Hinweis. Wenn eine Maschine A die Eingabe ε nicht akzeptiert, wie verhält sich dann die dazugehörige Maschine B?

Exercise 5:

Zeigen Sie, dass das 1-PCP, also das PCP für Instanzen, bei denen die x_i und y_i Wörter über dem unären Alphabet $\{1\}$ sind, entscheidbar ist.

Es sei $|\Sigma| \geq 2$. Das $\mathrm{PCP}_{\geq k}$ ist folgendes Entscheidungsproblem.

PCP$_{\geq k}$

Given:	Eine endliche Menge von Wort-Paaren $(x_1, y_1), \ldots, (x_m, y_m)$ über Σ^*
Question:	Gibt es eine endliche Sequenz von Indizes $i_1, \ldots, i_n$ mit $n \geq k$ und $x_{i_1} x_{i_2} \ldots x_{i_n} = y_{i_1} y_{i_2} \ldots y_{i_n}$?

Zeigen Sie, dass das PCP$_{\geq k}$ unentscheidbar ist.

Es sei $|\Sigma| \geq 2$. Das Last-PCP ist folgendes Entscheidungsproblem.

Last-PCP

Given:	Eine endliche Menge von Wort-Paaren $(x_1, y_1), \ldots, (x_m, y_m)$ über Σ^*
Question:	Gibt es eine endliche Sequenz von Indizes $i_1, \ldots, i_n$ mit $i_n = m$ und $x_{i_1} x_{i_2} \ldots x_{i_n} = y_{i_1} y_{i_2} \ldots y_{i_n}$?

Zeigen Sie, dass das Last-PCP unentscheidbar ist.