Algorithmen und Datenstrukturen 2

ALGO2 · WiSe-2023/24 · tcs.uni-frankfurt.de/algo2/ · 2024-02-05 · 7ae310c

Übungen zu Woche 10: Berechenbarkeit

Zur Erinnerung:

- $\textsc{Accept}(M) := \{w \in \Sigma^* \mid M \text{ akzeptiert } w\}$
- $\textsc{Reject}(M) := \{w \in \Sigma^* \mid M \text{ verwirft } w\}$
- $\textsc{Halt}(M) := \textsc{Accept}(M) \cup \textsc{Reject}(M)$
- $\textsc{Diverge}(M) := \Sigma^* \setminus \textsc{Halt}(M)$

Dienstag

🏟️👍 **10.1 Turingmaschinen.** Sei M eine beliebige Turingmaschine.

a) Beschreibe eine Turingmaschine M^R, sodass:

$$\textsc{Accept}(M^R) = \textsc{Reject}(\mathrm{M}) \qquad \text{und} \qquad \textsc{Reject}(M^R) = \textsc{Accept}(M)$$

b) Beschreibe eine Turingmaschine M^A, sodass:

$$\textsc{Accept}(M^A) = \textsc{Accept}(M) \qquad \text{und} \qquad \textsc{Reject}(M^A) = \varnothing$$

c) Beschreibe eine Turingmaschine M^H, sodass:

$$\textsc{Accept}(M^H) = \textsc{Halt}(M) \qquad \text{und} \qquad \textsc{Reject}(M^H) = \varnothing$$

🏟️🔑 **10.2 Entscheidbarkeit I.** Beweise für jede der folgenden vier Sprachen, dass sie unentscheidbar ist:

a) $\textsc{Halt} := \{\langle M, w\rangle \mid M \text{ hält auf Eingabe } w\}$
b) $\textsc{Diverge} := \{\langle M, w\rangle \mid M \text{ hält nicht auf Eingabe } w\}$
c) $\textsc{NeverHalt} := \{\langle M\rangle \mid \textsc{Halt}(M) = \varnothing\}$
d) $\textsc{AlwaysHalt} := \{\langle M\rangle \mid \textsc{Halt}(M) = \Sigma^*\}$

Donnerstag

Moodle 🐣 🔑 **10.3 Entscheidbarkeit II.** Wir beschränken uns im Folgenden auf deterministische Turingmaschinen.

a) Zeige über eine Reduktion ausgehend von der Sprache ACCEPT, dass die Sprache

$$L_1 = \left\{ \langle M, w \rangle \;\middle|\; \begin{array}{l} M \text{ ist eine Turingmaschine} \\ \text{und } M \text{ erreicht Zustand 1 für Eingabe } w \end{array} \right\}$$

nicht entscheidbar ist.

b) Zeige unter Anwendung des Satzes von Rice, dass die folgende Sprache nicht entscheidbar ist. Begründe die Nicht-Trivialität der Menge S.

$$L_2 = \left\{ \langle M \rangle \;\middle|\; \begin{array}{l} M \text{ ist eine Turingmaschine} \\ \text{und } |L(M)| \leq 2 \end{array} \right\}$$

Hierbei ist $L(M)$ die von der deterministischen Turingmaschine M akzeptierte Sprache.

🏟️ 🔑 **10.4 Entscheidbarkeit III.** Entwerfe für jedes der folgenden Entscheidungsprobleme entweder einen Algorithmus oder beweise, dass das Problem unentscheidbar ist. Die Eingabe ist für jedes Entscheidungsproblem die Kodierung $\langle M \rangle$ einer Turingmaschine M.

a) Akzeptiert M die Eingabe $\langle M \rangle \langle M \rangle$?
b) Akzeptiert M alle Palindrome?
c) Akzeptiert M die Sprache $\{\langle M \rangle \mid M$ hat min. 100 Zustände und hält auf Eingabe $\langle M \rangle\}$?
d) Gibt es einen Eingabestring, der M zu einer Bewegung nach links zwingt?

Weitere Aufgaben und Projekte ✒️

10.5 Entscheidbarkeit IV. Argumentiere für jedes der folgenden Entscheidungsprobleme, dass es unentscheidbar ist.

a) Für ein als Eingabe gegebenes Python-Programm `fib.py` wollen wir verifizieren, ob `fib.py` ein korrektes Programm für die Fibonacci-Zahlen ist, also ob `fib.py` bei Eingabe n die n-te Fibonacci-Zahl ausgibt.
b) Kann ein als Eingabe gegebenes Python-Programm `p.py` in eine Endlosschleife gelangen?
c) Berechnen zwei als Eingabe gegebene Python-Programme, `p1.py` und `p2.py`, die gleiche Funktion?

10.6 Entscheidbarkeit V. Entwerfe für jedes der folgenden Entscheidungsprobleme entweder einen Algorithmus oder beweise, dass das Problem unentscheidbar ist. Die Eingabe ist für jedes Entscheidungsproblem die Kodierung $\langle M, w \rangle$ einer Turingmaschine M und ihrem Eingabestring w.

a) Akzeptiert M entweder w oder w^R?
b) Akzeptiert M die Eingabe w in höchstens $2^{|w|}$ Schritten?
c) Wenn M auf der Eingabe w ausgeführt wird, gelangt M jemals wieder in den Startzustand?