8. Aufgabenblatt zur Vorlesung

Grundlagen der theoretischen Informatik — SoSe 2020

Wolfgang Mulzer

Abgabe bis zum 22. Juni 2020, 10 Uhr, im Whiteboard

Bitte erläutern und begründen Sie alle Ihre Antworten.

Aufgabe 1 Reduktionen *4+3+3 Punkte*

(a) Sei $H_0 := \{\langle M \rangle \mid M \text{ hält bei Eingabe } \varepsilon\}$. Zeigen Sie: $H \leq H_0$.

(b) Seien X und R Sprachen, so dass R regulär ist und $X \leq R$ gilt. Muss X dann ebenfalls regulär sein? Begründen Sie Ihre Antwort.

(c) Sei D die Diagonalsprache und U die Universalsprache. Zeigen Sie, dass es keine Reduktion von D auf U geben kann.

Hinweis: Zeigen Sie, dass D sonst entscheidbar wäre.

Aufgabe 2 Rekursiv aufzählbare Sprachen *5+5 Punkte*

(a) Beweisen Sie: Eine Sprache L ist genau dann entscheidbar bzw. rekursiv, wenn es eine Turingmaschine M gibt, die alle Wörter von L *in lexikographischer Reihenfolge* durch Blankzeichen getrennt auf ein Ausgabeband ausgibt, auf das nur geschrieben werden kann.

Hinweis: Sie müssen alle Wörter $w \in \Sigma^*$ aufzählen und jeweils überprüfen, ob w in L enthalten ist.

(b) Beweisen Sie: Eine Sprache L ist genau dann semientscheidbar bzw. rekursiv aufzählbar, wenn es eine Turingmaschine M gibt, die alle Wörter von L *in beliebiger Reihenfolge* durch Blankzeichen getrennt auf ein Ausgabeband ausgibt, auf das nur geschrieben werden kann.

Hinweis: Gehen Sie vor wie bei (a). Wie können Sie dies bewerkstelligen, ohne dass ein Problem entsteht, wenn die Turingmaschine für L nicht anhält?

Aufgabe 3 Unäres PKP *10 Punkte*

Zeigen Sie: Das PKP über dem unären Alphabet $\Sigma = \{0\}$ ist entscheidbar.

Aufgabe 1 Reduktionen von Thore Brehmer und David Jey *4+3+3 Punkte*

(a) Sei $H_0 := \{\langle M\rangle \mid M \text{ hält bei Eingabe } \varepsilon\}$. Zeigen Sie: $H \leq H_0$.

(b) Seien X und R Sprachen, so dass R regulär ist und $X \leq R$ gilt. Muss X dann ebenfalls regulär sein? Begründen Sie Ihre Antwort.

(c) Sei D die Diagonalsprache und U die Universalsprache. Zeigen Sie, dass es keine Reduktion von D auf U geben kann.

Hinweis: Zeigen Sie, dass D sonst entscheidbar wäre.

a) $H := \{(\langle M\rangle, w) \mid \text{Die TM } M \text{ hält bei Eingabe } w \in \Sigma^*\}$

z.z. $H \leq H_0$

- Eingabe: $(\langle M\rangle, w)$
- Funktion: Konstruiere M' wie folgt:
  - Falls Eingabewort $= \varepsilon$, schreibe w auf das Band
  - Führe M auf w aus
  - Wenn M hält, halte auch

Das ist irrelevant.

Warum schreibt ihr nur dann w auf das Band, wenn ihr sowieso immer die Simulation von M auf w ausführt?

- Ausgabe: $(\langle M'\rangle)$
- Es gilt f ist berechenbar. $(\langle M\rangle, w) \in H \iff (\langle M'\rangle) \in H_0$

Warum?

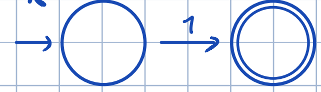
Warum? Das müsst ihr zeigen. -1P

$\Rightarrow H \leq H_0$

b) Nein. Bsp. sei $X := \{0^n 1^n \mid n \in \mathbb{N}\}$. X ist nicht regulär (aus Vorlesung)

sei $R := \{1\}$ offensichtlich ist R regulär

R



zeige $X \leq R$

- Eingabe: $w \in \Sigma^*$
- Funktion: $f \begin{cases} 1, & \text{falls } w \in X \text{ (in form von } 0^n1^n) \\ 0, & \text{sonst} \end{cases}$ $\quad f(w) = w'$
- Ausgabe: $w' \in \{0,1\}$
- Es gilt f ist berechenbar. $w \in X \iff w' \in R$

Warum gilt das? Zeigen! -1P

$\Rightarrow X \leq R$

wobei X nicht regulär obwohl R regulär!

c)

c) $D := \{ \langle M \rangle \mid$ Die TM M akzeptiert die Eingabe $\langle M \rangle$ nicht $\}$

$U := \{ (\langle M \rangle, w) \mid$ Die TM M akzeptiert die Eingabe $w \in \Sigma^* \}$

z.z. $D \nleq U$

fehlt -3P

5/10

Aufgabe 2 Rekursiv aufzählbare Sprachen *5+5 Punkte*

(a) Beweisen Sie: Eine Sprache L ist genau dann entscheidbar bzw. rekursiv, wenn es eine Turingmaschine M gibt, die alle Wörter von L *in lexikographischer Reihenfolge* durch Blankzeichen getrennt auf ein Ausgabeband ausgibt, auf das nur geschrieben werden kann.

Hinweis: Sie müssen alle Wörter $w \in \Sigma^*$ aufzählen und jeweils überprüfen, ob w in L enthalten ist.

(b) Beweisen Sie: Eine Sprache L ist genau dann semientscheidbar bzw. rekursiv aufzählbar, wenn es eine Turingmaschine M gibt, die alle Wörter von L *in beliebiger Reihenfolge* durch Blankzeichen getrennt auf ein Ausgabeband ausgibt, auf das nur geschrieben werden kann.

Hinweis: Gehen Sie vor wie bei (a). Wie können Sie dies bewerkstelligen, ohne dass ein Problem entsteht, wenn die Turingmaschine für L nicht anhält?

a) L ist entscheidbar $\Leftrightarrow$ es gibt TM M für L

1. L ist entscheidbar $\Rightarrow$ es gibt TM M für L

Konstruiere M wie folgt aus M_1 und M_2:

- M_1 erstellt alle $w \in \Sigma^*$ nacheinander in lexikographischer Reihenfolge und nutzt
- M_2 zum überprüfen, ob $w \in L$. Falls $w \in L$ schreibe w auf das Ausgabeband, sonst verwerfe w. Danach erstellt M_1 das nächste Wort
(M₂ nur möglich da L entscheidbar)

$\Rightarrow$ M zählt alle Wörter von L in lexikographischer Reihenfolge auf.

2. L ist entscheidbar $\Leftarrow$ es gibt TM M für L

Konstruiere die TM $M' := \{ w' \mid w' \text{ ist in } L \}$ (die L entscheidet) wie folgt:

- Simuliere M und vergleiche jedes neue Wort m, welches M erstellt, mit der Eingabe w'
- Falls $m = w'$ gib q_{ja} aus
- Falls m größer ist als w' (mehr Buchstaben besitzt) gib q_{nein} aus.
- Falls alle Wörter aus M überprüft wurden gib q_{nein} aus

$\Rightarrow$ M' terminiert immer $\Rightarrow$ L ist entscheidbar

b) L ist semientscheidbar $\Leftrightarrow$ es gibt TM M für L

1. L ist semientscheidbar $\Rightarrow$ es gibt TM M für L

Idee: da L semientscheidbar ist, können wir die Wörter von L nicht einfach nacheinander abarbeiten wie in a, da die Überprüfung von einem Wort unendlich lang dauern könnte. Daher werden wir die Wörter im "Zickzack" mit begrenzten Rechenschritten überprüfen.

Konstruiere M wie folgt aus M_1 und M_2:

- M_1 erstellt für alle $w \in \Sigma^*$ nacheinander "Zickzack" Tupel wie folgt:
  - (w_i, j) wobei j für Rechenschritte steht und w_i für ein Wort aus Σ^*
    für (w_i, j) muss gelten $(w_i, j) < (w_i', j')$ wobei das gilt wenn $i+j < i'+j'$ gilt (Zickzack)
    Bsp. die ersten Tupel sehen wie folgt aus: $(w_1, 1), (w_1, 2), (w_2, 1), (w_1, 3), (w_2, 2), (w_3, 1)$
- Nun kann M_2 das erstellte Wort von M_1 überprüfen. Falls w_i nach genau j Schritten von M_2 angenommen wird, schreibe w_i auf das Ausgabeband. Sonst verwerfe das Wort.
  Nach dem Überprüfen des (w_i, j) Tupels, gehe zur nächsten Überprüfung über.
  (Da L semientscheidbar gibt es eine TM M_2 welche L semientscheidet)

$\Rightarrow$ M zählt alle Wörter von L auf.

2. L ist semientscheidbar $\Leftarrow$ es gibt TM M für L

Konstruiere die TM $M' := \{ w' \mid w' \text{ ist in } L \}$ (die L semientscheidet) wie folgt:

- Simuliere M und vergleiche jedes neue Wort m, welches M erstellt, mit der Eingabe w'
- Falls $m = w'$ gib q_{ja} aus
- sonst gehe zum nächsten Wort. Da L unendlich lang sein kann und M keine Reihenfolge besitzt $\Rightarrow$ kann es dazu kommen, dass M' nicht anhält.

$\Rightarrow$ M' terminiert nicht immer $\Rightarrow$ L ist semientscheidbar

Aufgabe 3 Unäres PKP *10 Punkte*

Zeigen Sie: Das PKP über dem unären Alphabet $\Sigma = \{0\}$ ist entscheidbar.

Die PKP Sprache mit dem Alphabet wird aus Wörtern der folgenden Form bestehen: aus Paaren von Wörtern aus $(0^{x_1}, 0^{y_1}), \ldots (0^{x_i}, 0^{y_i}) \in (0^*)^2$

Für bessere Lesbarkeit werden wir aber nur (x_i, y_i) schreiben, wobei x und y für die Anzahlen von 0 stehen.

Konstruiere eine TM M welche

Fall 1. Für alle Paare (x_i, y_i) gilt $|x_i| > |y_i|$

$\Rightarrow$ Dann werden auch alle möglich Wörter Kombinationen aus x_i > alle möglichen Wörter Kombinationen aus y_i

$\Rightarrow$ Es gibt keine Korrespodenz

Fall 2. Für alle Paare (x_i, y_i) gilt $|x_i| < |y_i|$

$\Rightarrow$ Dann werden auch alle möglich Wörter Kombinationen aus x_i < alle möglichen Wörter Kombinationen aus y_i

$\Rightarrow$ Es gibt keine Korrespodenz

Fall 3. Es gibt ein Paar (x_i, y_i) für das gilt $|x_i| = |y_i|$

$\Rightarrow x_i = y_i \Rightarrow$ Es gibt eine Korrespodenz i

Fall 4. Es gibt je mindestens ein Paar (x_i, y_i) für das gilt $|x_i| < |y_i|$ und ein anderes Paar (x_j, y_j) für das gilt $|x_j| > |y_j|$

Existiert eine Korrespodenz? Es muss gelten $|x_i| + |x_j| = |y_i| + |y_j|$ oder auch $|x_i| - |y_i| = |y_j| - |x_j|$

Für welche $u, v \in \mathbb{N}$ gilt $u(|x_i| - |y_i|) = v(|y_j| - |x_j|)$

wahr für $u = (|y_j| - |x_j|)$ und $v = (|x_i| - |y_i|)$

$\Rightarrow$ Es existiert eine Korrespodenz aus $|x_j| - |y_j|$ - oft das i Paar und $|y_i| - |x_i|$ - oft dem j Paar.
(Reihenfolge ist egal, da für addition und multiplikation das Kommutativgesetz gilt)

Bsp. Sei die Eingabe Paare $i = (0^4, 0^1)$ und $j = (0^2, 0^3)$

Aus j: $|y| - |x| = 3 - 2 = 1$ mal das Paar i

Aus i: $|x| - |y| = 4 - 1 = 3$ mal das Paar j

Korrespodenz: $1 \cdot i + 3 \cdot j = 1 \cdot (0^4, 0^1) + 3 \cdot (0^2, 0^3) = (0^4, 0^1) + (0^6, 0^9) = (0^{10}, 0^{10})$

Für alle möglichen Fälle gibt es immer eine Ausgabe.
(Entweder gibt es eine Korrespodenz oder nicht)

$\Rightarrow$ Das PKP über dem unären Alphabet $\Sigma = \{0\}$ ist entscheidbar

TM angeben, die diese Bedingungen überprüft! -1P

9/10