Übungsblatt 06

Til Mohr, 405959
Andrés Montoya, 405409
Marc Ludevid Wulf, 405401

Aufgabe 4

1) $\left[\frac{ab}{abb}\right]\left[\frac{bb}{b}\right]$ ist eine mögliche Lösung dieser PKP-Instanz.

2)
   - $\left[\frac{a}{ba}\right], \left[\frac{a}{bb}\right], \left[\frac{aab}{ab}\right], \left[\frac{abab}{aa}\right]$ sind keine möglichen Startdominos, da sie mit unterschiedlichen Buchstaben beginnen.
   - $\left[\frac{ab}{abb}\right], \left[\frac{aa}{aab}\right]$ erzeugen zwar selber gleich beginnende Worte oben und unten, jedoch endet bei beiden jedes Wort unten mit einem b mehr als oben. Zudem gibt es keinen Dominostein, der oben mit einem b beginnt. Daher sind diese auch keine möglichen Startdominos.

   Da es keine möglichen Startdominos gibt, hat diese PKP-Instanz keine Lösung.

Aufgabe 5

Um die Aussage zu beweisen, zeigen wir erst, dass L_{01} rekursiv ist:
Sei T eine TM mit folgender Funktionsweise:

(i) Gehe an den Anfang des Eingabewortes.

(ii) Ist der Buchstabe unter dem Kopf eine 1, so verwirf die Eingabe. Ist der Buchstabe unter dem Kopf ein B, so verwirf die Eingabe (Eingabe war ϵ). Ist der Buchstabe unter dem Kopf eine 0, so lösche das Zeichen, und gehe zu dem Ende des Eingabewortes.

(iii) Ist der Buchstabe unter dem Kopf eine 0, so verwirf die Eingabe. Ist der Buchstabe unter dem Kopf eine 1, so lösche das Zeichen und gehe einen Schritt nach Links.

(iv) Ist das Zeichen unter dem Kopf B, akzeptiere die Eingabe. Sonst fahre bei Schritt *(i)* fort.

Korrektheit:

- Sei ϵ das Eingabewort, $\epsilon \notin L_{01} \Rightarrow T$ verwirft die Eingabe sofort.
- Sei w das Eingabewort, $w \notin L_{01} \Rightarrow$ Durch das gleichmäßige Abbauen des Eingabewortes an beiden Seiten erkennt T in Schritten *(ii)* bzw. *(iii)* irgendwann, dass die Eingabe nicht dem Format $0^n1^n, n > 0$ entspricht $\Rightarrow T$ verwirft
- Sei w das Eingabewort, $w \in L_{01} \Rightarrow$ Duch das gleichmäßige Abbauen des Eingabewortes an beiden Seiten wird T keine Fehler des Formates $0^n1^n, n > 0$ an w entdecken $\Rightarrow T$ akzeptiert

T erkennt offensichtlich L_{01}, daher ist L_{01} rekursiv.
Sei nun L eine Sprache.

$\Rightarrow$ Sei L rekursiv. Dann gibt es eine berechenbare Funktion f die L entscheidet. Diese Funktion f ändern wir nun so zu f' ab, dass sie 01 ausgibt falls f ja ausgibt und 11 falls f nein ausgibt. Somit gilt nun, dass wenn $x \in L$ dann auch $f'(x) \in L_{01}$ liegt und wenn $x \notin L$ dann auch $f'(x) \notin L_{01}$ liegt. Also gilt $L \leq L_{01}$.

$\Leftarrow$ Gelte $L \leq L_{01}$. Da ja L_{01} rekursiv ist, muss auch L rekursiv sein. (VL)

Damit gilt die Aussage.

Aufgabe 6

a) Ansatz: Seien A und B zwei Tupel $\mathbb{Z}^k$ wobei k die Anzahl der Dominosteine ist. Gegeben das i-te Dominostein $\left[\frac{w_1}{w_2}\right]$ dann ist $A_i = |w_1|_a - |w_2|_a$ und $B_i = |w_1|_b - |w_2|_b$.
Beispiel: $\{\left[\frac{abbb}{bbaa}\right], \left[\frac{baa}{a}\right], \left[\frac{a}{ab}\right]\} \Rightarrow A = (-1, 1, 0); B = (1, 1, -1)$

Nun gilt es eine Lösung p für das folgende Gleichungssystem zu finden. p ist wiederum ein Tupel der Form $\mathbb{N}^k$.
$\sum_{1 \leq j \leq k}(p_j * A_j) = 0$
$\sum_{1 \leq j \leq k}(p_j * B_j) = 0$
Für unser Beispiel wäre z.B. $p = (1, 1, 2)$ eine Lösung.

Sei nun q ebenfalls ein Tupel der Form $\mathbb{N}^k$:
$q_1 \coloneqq p_1$
$q_n \coloneqq q_{n-1} + p_n$

Sei nun so eine Lösung gegeben, dann ist die Folge I der Länge q_k folgendermassen definiert:
$I_1 ... I_{q_1} \coloneqq 1$
$I_{q_1} ... I_{q_2} \coloneqq 2$
...
$I_{q_{k-1}} ... I_{q_k} \coloneqq k$

Da lineare Gleichungssysteme algorithmisch lösbar sind und jeder der beschriebenen Schritte ebenfalls berechenbar ist, ist das Problem entscheidbar.

Korrektheit:

- p eine Lösung des GS $\Rightarrow$ I ist Folge sodass Variante des PKP gelöst ist.
  Sei nun dieses q gegeben. Dann berechne wie oben definiert I.
  $\sum_{1 \leq j \leq k}(p_j * A_j) = 0$ und $\sum_{1 \leq j \leq k}(p_j * B_j) = 0$
  $\Rightarrow \sum_{1 \leq j \leq q_k}(|x_{I_j}|_a) = \sum_{1 \leq j \leq q_k}(|y_{I_j}|_a)$ und $\sum_{1 \leq j \leq q_k}(|x_{I_j}|_b) = \sum_{1 \leq j \leq q_k}(|y_{I_j}|_b)$

- GS hat keine Lösung $\Rightarrow$ es gibt keine Folge I sodass Variate des PKP gelöst ist.
  Kontraposition: I ist Folge sodass Variante des PKP gelöst ist $\Rightarrow$ p eine Lösung des GS
  Sei nun das I gegeben. Dann ist p_j die Anzahl an Vorkommen von j in I.
  Dann ist p eine Lösung des GS: $\sum_{1 \leq j \leq k}(p_j * A_j) = 0$ und $\sum_{1 \leq j \leq k}(p_j * B_j) = 0$, denn es gilt:
  $\sum 1 \leq j \leq q_k(|x_{I_j}|_a) = \sum 1 \leq j \leq q_k(|y_{I_j}|_a)$ und $\sum 1 \leq j \leq q_k(|x_{I_j}|_b) = \sum 1 \leq j \leq q_k(|y_{I_j}|_b)$

b) Dieses Problem ist gleich dem PKP.
Sei K eine PKP-Instanz, wobei hier k_i der i-ten Stein aus K ist.
Aus K können wir nun eine Instanz K' unserer PKP-Variante konstruieren. Dazu wende folgendes Verfahren an:
Sei $\# \notin \Sigma$.

- Füge den Universalstein $u \coloneqq \left[\frac{\#\#}{\#}\right]$ zu K' hinzu.
- Hat der k_i Stein oben und unten verschieden lange Wörter, dann ist der Stein der Form $\left[\frac{a_1 ... a_n}{b_1 ... b_q}\right]$ mit $a_j, b_l \in \Sigma$ für $0 < j \leq n, 0 < l \leq q, n \neq q$.

So füge K' folgenden Stein hinzu:
$x_i := \left[\frac{a_1\#\#a_2\#\#...a_n\#\#}{b_1\#\#b_2...b_q\#\#}\right]$

- Hat der k_i Stein gleich lange Wörter oben und unten, dann ist der Stein der Form $\left[\frac{a_1...a_n}{b_1...b_n}\right]$ mit $a_j, b_j \in \Sigma$ für $0 < j \leq n$.
  Füge dann einen Stein der folgender Form K' hinzu:
  $y_i := \left[\frac{a_1\#\#a_2\#\#...a_n}{b_1\#\#b_2\#\#...b_n\#}\right]$
  Damit muss immer auf ein y_i ein u folgen

Gibt es also eine Folge $\langle o_1, ..., o_m \rangle$, sodass $k_{o_1}...k_{o_m}$ eine Lösung von K ist, dann gibt es eine Folge $\langle o'_1, ..., o'_p \rangle, p \geq m$, sodass $k'_{o_1}...k'_{o_p}$ eine Lösung von K' ist:

Input: $\langle o_1, ..., o_m \rangle$
i = 1
I = $\langle\rangle$
while(i $\leq$ m)
    if(k_{o_i} oben und unten verschieden lang)
        I += index(x_{o_i})
    else
        I += index(y_{o_i})
        I += index(u)
    i++
return I

, wobei hier `index(k)` den Index von Dominostein `k` aus K' zurückgibt.
Also können wir das PKP auf dieses Problem ableiten. Da das PKP nicht berechenbar ist, ist dieses Problem auch nicht berechenbar.