

Aufgabe 4

- 1) $\begin{bmatrix} ab \\ abb \end{bmatrix} \begin{bmatrix} bb \\ b \end{bmatrix}$ ist eine mögliche Lösung dieser PKP-Instanz.
- 2)
 - $\begin{bmatrix} a \\ ba \end{bmatrix}, \begin{bmatrix} a \\ bb \end{bmatrix}, \begin{bmatrix} aab \\ ab \end{bmatrix}, \begin{bmatrix} abab \\ aa \end{bmatrix}$ sind keine möglichen Startdominos, da sie mit unterschiedlichen Buchstaben beginnen.
 - $\begin{bmatrix} ab \\ abb \end{bmatrix}, \begin{bmatrix} aa \\ aab \end{bmatrix}$ 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 $0^n 1^n, n > 0$ entspricht $\Rightarrow T$ verwirft
- Sei w das Eingabewort, $w \in L_{01} \Rightarrow$ Durch das gleichmäßige Abbauen des Eingabewortes an beiden Seiten wird T keine Fehler des Formates $0^n 1^n, n > 0$ 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 gilt auch $L \leq L_{01}$, da ja auch L_{01} rekursiv ist. (VL)

\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)

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 $u := \begin{bmatrix} \#\# \\ \# \end{bmatrix}$ zu K' hinzu.
- Hat der k_i Stein oben und unten verschieden lange Wörter, dann ist der Stein der Form $\begin{bmatrix} a_1 \dots a_n \\ b_1 \dots b_m \end{bmatrix}$ mit $a_j, b_l \in \Sigma$ für $0 < j \leq n, 0 < l \leq m, n \neq m$.

So füge K' folgenden Stein hinzu:

$$x_i := \begin{bmatrix} a_1 \# \# a_2 \# \# \dots a_n \# \# \\ b_1 \# \# b_2 \# \# \dots b_m \# \# \end{bmatrix}$$

- Hat der k_i Stein gleich lange Wörter oben und unten, dann ist der Stein der Form $\begin{bmatrix} a_1 \dots a_n \\ b_1 \dots b_n \end{bmatrix}$ mit $a_j, b_j \in \Sigma$ für $0 < j \leq n$.

Füge dann einen Stein der folgender Form K' hinzu:

$$y_i := \begin{bmatrix} a_1 \# \# a_2 \# \# \dots a_n \\ b_1 \# \# b_2 \# \# \dots b_n \# \end{bmatrix}$$

Damit muss immer auf ein y_i ein u folgen

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

```

Input:  $\langle o_1, \dots, o_m \rangle$ 
i = 1
I =  $\langle \rangle$ 
while(i ≤ 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 $\text{index}(\mathbf{k})$ den Index von Dominostein \mathbf{k} aus K' zurückgibt.

Also können wir dieses Problem auf das PKP ableiten. Da das PKP nicht berechenbar ist, ist dieses Problem auch nicht berechenbar.