

Übungsblatt 2 mit Lösungen

Abgabetermin: Mittwoch, der 2. November 2022 um 14:30

Aufgabe 4 (Palindrome mit 2-Band-TM)**4 Punkte**

Für ein Wort $w = w_1 w_2 \dots w_n$ mit $w_i \in \Sigma$ bezeichnen wir mit $w^{-1} = w_n w_{n-1} \dots w_1$ die "Spiegelung" von w . Sei nun $L = \{ww^{-1} \mid w \in \Sigma^*\}$ die Sprache der Palindrome gerader Länge über dem Alphabet $\Sigma = \{0, 1\}$.

Beschreiben (in Worten) Sie eine möglichst effiziente 2-Band-TM, die L entscheidet. Analysieren Sie den Zeit- und den Speicherplatzbedarf der von Ihnen entworfenen Maschine.

Hinweis: Überlegen Sie sich zuerst, wie ein zweites Band die Erkennung eines Wortes in L schneller machen kann.

Lösung: _____

Die Idee besteht darin, die Eingabe zunächst auf das zweite Band zu kopieren und dann die Köpfe beim Vergleich in entgegengesetzte Richtungen zu bewegen. Alternativ kann auch das Wort in umgekehrter Reihenfolge auf das zweite Band geschrieben werden. Es ist nicht möglich den Schritt des kopierens (egal ob in umgekehrter oder normaler Reihenfolge) auszulassen, da das zweite Band mit B initialisiert ist. Während dem kopieren auf das zweite Band überprüft man außerdem ob das Wort gerade Länge hat, indem man zwei unterschiedliche Zustände zum kopieren verwendet, zwischen denen man hin und her wechselt.

Für das kopieren, zählen und zurücklaufen sind etwa $2n$ Schritte nötig. In einer zweiten Phase werden beide Zeichenketten miteinander verglichen. Durch die Verwendung von 2 Bändern (mit 2 Schreib-/Lese-Köpfen) ist Phase 2 ebenfalls in etwa n Schritten möglich. Zusammen dauert der Vergleich $3n + O(1)$ Schritte, also liegt die Gesamtlaufzeit in $O(n)$. Sowohl beim kopieren als auch beim vergleichen sind es nicht genau n bzw $2n$ Schritte, da die TM das Ende des Wortes erkennen muss, also jeweils ein Zeichen mehr betrachtet und dann wieder auf das Wort zurückgeht was etwa 2 Schritte extra kostet.

- 1) Kopiere Band 1 auf Band 2 und prüfe währenddessen ob die Eingabelänge gerade ist. Falls ungerade, terminiere mit NEIN.
- 2) Bewege Kopf 1 zurück auf den Start und lasse dabei Kopf 2 auf dem letzten Zeichen der Eingabe (auf Band 2) stehen.
- 3) Lese die Zeichen auf beiden Bändern und vergleiche sie. Falls ungleich, terminiere mit NEIN. Andernfalls bewege Kopf 1 nach rechts und Kopf 2 nach links.
- 4) Wiederhole Schritt 3 bis beide Köpfe ein Blank lesen. Dann terminiere mit JA.

Erklärung zu Schritt 4: Beim lesen des Blank-Symbols (das nicht in der Eingabe vorkommen kann) wissen wir, dass wir das Ende des Wortes erreicht haben und wenn wir vorher noch nicht mit NEIN terminiert haben, dann war das Wort ein Palindrom. Ohne den Test

auf gerade Eingabelänge in Schritt 1 würden wir beliebige Palindrome akzeptieren, was aber nicht der Aufgabenstellung entspricht.

Aufgabe 5 (Einseitig beschränktes Band)**5 Punkte**

Eine TM mit einseitig beschränktem Band ist eine 1-Band-TM, die die Positionen $p < 0$ nie benutzt (und mit dem Kopf auf Position 1) startet. Zeigen Sie, dass jede 1-Band-TM M durch eine 1-Band-TM M' mit einseitig beschränktem Band simuliert werden kann. Sie müssen nur das Akzeptanzverhalten der simulierten TM M übernehmen, das heißt es ist nur das erste Zeichen der Ausgabe relevant. Geben Sie eine möglichst effiziente Simulation.

Geben Sie den Zeitverlust ihrer Simulation an, d.h. falls die simulierte TM M $t(n)$ Schritte läuft, wie viele Schritte benötigt ihre Simulation M' (asymptotisch) abhängig von $t(n)$?

Hinweis: Überlegen Sie sich zuerst wie die Turingmaschine erkennen kann dass das Ende des Bandes erreicht ist.

Lösung:

Wir simulieren eine 1-Band-TM mit beidseitig unendlichem Band auf einer 1-Band-TM mit einseitig unendlichem Band. Dazu benutzen wir eine zweite Spur. Zugriffe auf Bandpositionen $p < 0$ werden auf Position $1 - p$ umgelenkt. An Position 0 steht ein Sonderzeichen, das sonst nicht verwendet wird, z.B. ein \$. Das neue Bandalphabet ist also $\Gamma_{\text{neu}} := \Gamma \cup \Gamma^2 \cup \{\$\}$. Die Beschreibung der Zustandsübergangsfunktion wird dabei doppelt so groß, sie enthält nämlich einmal das ursprüngliche und unveränderte δ , und einmal ein gespiegeltes δ' . Wo δ ein L enthält, enthält δ' ein R , und umgekehrt.

Falls der Kopf auf einer Position $p > 0$ steht, verhält sich die neue TM wie die ursprüngliche TM, verwendet also die Übergänge aus δ , wobei die Symbole in Spur 2 ignoriert werden. Dasselbe gilt an Position 0, sofern der Kopf nach rechts oder gar nicht bewegt wird. Wenn die ursprüngliche TM aber nach links gehen würde und sich der Kopf an Position 0 befindet (was am \$ zu erkennen ist), so findet stattdessen ein Schritt nach rechts statt und es wird bis zum nächsten Besuch der Position 0 die Übergangsfunktion δ' statt δ bzw. δ' statt δ benutzt.

Erreicht die simulierte TM den Endzustand, schreibt die neue TM das entsprechende Zeichen von Spur 1 oder 2 auf das Band (statt dem Zeichen (a, b) schreibt die neue TM a wenn sie aktuell auf Spur 1 arbeitet, bzw b wenn sie aktuell auf Spur 2 arbeitet).

Sei also $M = (Q, \Sigma, \Gamma, B, q_0, \bar{q}, \delta)$. Wir schreiben $\delta(q, a) = (\delta_Q(q, a), \delta_\Gamma(q, a), \delta_{\text{move}}(q, a))$ und, für $X \in \{R, L, N\}$, schreiben wir X^{-1} für die Bewegung in die entgegen gesetzte Richtung (R statt L usw.). Dann ist die neue TM $M^* = (Q^*, \Sigma, \Gamma^*, B, q_0^*, \bar{q}^*, \delta^*)$, wobei gilt:

- $Q^* := Q \times \{1, 2\} \uplus \{q_0^*, \bar{q}^*\}$,
- $\Gamma^* := \Gamma_{\text{neu}} = \Gamma \cup \Gamma^2 \cup \{\$\}$,
- q_0^*, \bar{q}^* sind neue Zustände, die nicht in Q vorkommen, und
- δ^* wird definitert durch die folgenden Regeln:
 - $\delta^*(q_0^*, a) := (q_0^*, a, L)$, für alle $a \in \Sigma$,

- $\delta^*(q_0^*, B) := ((q_0, 1), \$, R)$,
- $\delta^*((q, 1), a) := ((\delta_Q(q, a), 1), (\delta_\Gamma(q, a), B), \delta_{\text{move}}(q, a))$, für alle $a \in \Gamma$ und $q \in Q \setminus \{\bar{q}\}$,
- $\delta^*((q, 2), a) := ((\delta_Q(q, B), 2), (a, \delta_\Gamma(q, B)), \delta_{\text{move}}^{-1}(q, B))$, für alle $a \in \Gamma$ und $q \in Q \setminus \{\bar{q}\}$,
- $\delta^*((q, 1), (a, b)) := ((\delta_Q(q, a), 1), (\delta_\Gamma(q, a), b), \delta_{\text{move}}(q, a))$, für alle $(a, b) \in \Gamma^2$ und $q \in Q \setminus \{\bar{q}\}$,
- $\delta^*((q, 2), (a, b)) := ((\delta_Q(q, b), 2), (a, \delta_\Gamma(q, b)), \delta_{\text{move}}^{-1}(q, b))$, für alle $(a, b) \in \Gamma^2$ und $q \in Q \setminus \{\bar{q}\}$,
- $\delta^*((q, \alpha), \$) := ((q, 3 - \alpha), \$, R)$, für alle $q \in Q$,
- $\delta^*((\bar{q}, 1), a) := (\bar{q}^*, a, N)$, für alle $a \in \Gamma$,
- $\delta^*((\bar{q}, 2), a) := (\bar{q}^*, B, N)$, für alle $a \in \Gamma$,
- $\delta^*((\bar{q}, 1), (a, b)) := (\bar{q}^*, a, N)$, für alle $(a, b) \in \Gamma^2$, und
- $\delta^*((\bar{q}, 2), (a, b)) := (\bar{q}^*, b, N)$, für alle $(a, b) \in \Gamma^2$.

Diese Simulation verändert die asymptotische Laufzeit nicht. Es gibt initial konstanten Aufwand um das \$ auf das Band zu schreiben, und sonst wird jeder Schritt von der ursprünglichen TM durch maximal zwei Schritte simuliert. Die Ausgabe des korrekten Zeichens ist wiederum in konstanter Zeit möglich.

(Dieses Verfahren bezieht sich auf Turingmaschinen, die nur akzeptieren bzw. verwerfen. Wird ein Ausgabewort geschrieben, so müsste dies wieder „einspurig“ hingeschrieben werden. Dabei ist der Zeitverlust quadratisch in der Größe des Ausgabewortes.)

Aufgabe 6 (RAM-Programm für gewünschte Laufzeit)**6 Punkte**

Geben Sie ein kurzes RAM-Programm an, welches für eine Eingabe $m \in \mathbb{N}_{>0}$ in Register 1 mit n Bits im *uniformen Kostenmaß* eine Laufzeit von $\Theta(\sqrt{2^n})$ hat (die Ausgabe spielt keine Rolle). Geben Sie auch die Laufzeit im *logarithmischen Kostenmaß* an. Erläutern Sie die Funktionsweise des RAM-Programms und zeigen Sie, dass ihr RAM-Programm die gewünschte Laufzeit hat.

Lösung: _____

```

1: LOAD 2
2: CADD 1
3: STORE 2
4: MULT 2
5: CADD 1
6: SUB 1
7: IF C(0)=0 THEN GOTO 1
8: END

```

Die RAM initialisiert Register 2 mit dem Wert $c(2) = 0$. Dann durchläuft sie eine Schleife: Als erstes wird die Zahl $c(2)$ um 1 erhöht und abgespeichert (durch die Befehle LOAD 2; CADD 1; STORE 2). Es gilt nun $c(0) = c(2)$ und im nächsten Schritt wird $c(0)$ dann quadriert (durch den Befehl MULT 2). Danach wird 1 addiert und $c(1)$ subtrahiert (durch die Befehle CADD 1; SUB 1). Nun gilt $c(0) = c(2)^2 + 1 - c(1)$. Ist das Ergebnis 0 (also gilt $c(2)^2 + 1 \leq c(1)$), dann wird die Schleife wiederholt. Wird die Schleife abgebrochen steht im Register 2 die Zahl k , die die Anzahl der Schleifendurchläufe angibt, mit $k^2 + 1 - m > 0$ bzw. $k^2 \geq m$

Es gilt:

$$\begin{array}{lll}
 k \leq \sqrt{m} + 1 & \leq \sqrt{2^n} + 1 & \in O(\sqrt{2^n}) \\
 k \geq \sqrt{m} & \geq \sqrt{2^{n-1}} & \in \Omega(\sqrt{2^n})
 \end{array}$$

Dies ergibt im **uniformen Kostenmaß** eine Laufzeit von $k \in \Theta(\sqrt{2^n})$.

Im **logarithmischen Kostenmaß** werden in jedem Schleifendurchlauf Kosten in Höhe von $\Theta(n)$ verursacht. Dies geschieht durch das Lesen der n -Bit Zahl in Register 1 und der k -Bit Zahl in Register 2, wobei $k \leq n$. Dies ergibt eine Gesamtlaufzeit von $\Theta(\sqrt{2^n} \cdot n)$