

Übung “Grundbegriffe der Informatik”

Karlsruher Institut für Technologie

Matthias Schulz, Gebäude 50.34, Raum 247

email: schulz@ira.uka.de

Matthias Janke, Gebäude 50.34, Raum 249

email: matthias.janke@kit.edu

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Vorgehensweise: ???

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

Überprüfe, ob Wort Binärdarstellung ist!

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

Schreibe nacheinander Wörter **10, 11, 101, 110, 111, ...** hinter ursprüngliches Wort.

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

Schreibe nacheinander Wörter **10, 11, 101, 110, 111, ...** hinter ursprüngliches Wort. Nach einem Trennsymbol **:**

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

Teile gegebenes Wort durch Wort hinter :

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

Falls Rest 0 und Wort hinter $:$ ungleich w : Nicht prim.

.

Turing-Maschinen: Design

Gegeben: Problem für Wörter (z.B.: Überprüfe, ob Wort Binärdarstellung einer Primzahl ist)

Gesucht: Turingmaschine, die das Problem löst.

Erste Überlegung: Wie würde ich das mit Bleistift und Papier lösen?

Falls Rest 0 und Wort hinter $:$ gleich w : prim.

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

Gehe ans rechte Ende des Bandes:

$$\forall x \in X \setminus \{\square\} : (f, g, m)(S, x) = (S, x, 1)$$

$$(f, g, m)(S, \square) = (S_1, \square, -1)$$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

Addiere 1 zu letzter Ziffer, merke Übertrag:

$$(S_1, 0) \mapsto (S_0, 1, -1)$$

$$(S_1, 1) \mapsto (S_1, 0, -1)$$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

Addiere Übertrag zu Ziffer, merke Übertrag:

$$(S_0, 0) \mapsto (S_0, 0, -1)$$

$$(S_0, 1) \mapsto (S_0, 1, -1)$$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

Was passiert bei $(S_1, :)$?

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

Was passiert bei $(S_1, :)$?

Entweder Wort vorne dran 1 nach links verschieben,
oder Wort hinter : 1 nach rechts verschieben.

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Binärdarstellung um 1 erhöhen.

Was passiert bei $(S_1, :)$?

Entweder Wort vorne dran 1 nach links verschieben,
oder Wort hinter $:$ 1 nach rechts verschieben.

Siehe Übungsblatt 12!

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Teilen:

100001 : 11 ($w : w'$)

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Teilen:

100001 : 11 ($w : w'$)

Vergleiche ersten $|w'|$ Ziffern von w mit w' .

Falls Ziffer von w immer \geq Ziffer von w' : w' abziehen.

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Teilen:

100001 : 11 ($w : w'$)

Vergleiche ersten $|w'|$ Ziffern von w mit w' .

Sonst w' von den ersten $|w'| + 1$ Ziffern von w abziehen.

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Merke letztes Zeichen; **markiere** letztes Zeichen!

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Merke letztes Zeichen; **markiere** letztes Zeichen!

$(S, 0) \mapsto (S_0, \bar{0}, -1)$

$(S, 1) \mapsto (S_1, \bar{1}, -1)$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Gehe zu Wort vor dem :

$$\forall x \in X \setminus \{:\} \forall i \in \mathbb{G}_2 : (S_i, x) \mapsto (S_i, x, -1)$$

$$(S_i, :) \mapsto (Z_i, :, -1)$$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Gehe zu erster unmarkierter Ziffer vor dem :

$$(Z_i, \bar{0}) \mapsto (Z_i, \bar{0}, -1)$$

$$(Z_i, \bar{1}) \mapsto (Z_i, \bar{1}, -1)$$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Ziehe Ziffern von einander ab; merke Übertrag:

$(Z_0, i) \mapsto (U_0, \bar{i}, 1)$

$(Z_1, 1) \mapsto (U_0, \bar{0}, 1)$ $(Z_1, 0) \mapsto (U_1, \bar{1}, 1)$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Gehe zu letzter nicht markierter Ziffer nach :

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Addiere Übertrag zu gelesener Ziffer:

$$(U'_i, j) \mapsto (S_{i+j}, \bar{j}, -1)$$

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Wenn Übertrag 1 und keine unmarkierte Ziffer hinter :

.

Turing-Maschinen: Design

Jeden Schritt in Turingmaschine übersetzen!

Z.B. Abziehen:

100 : 11 ($w : w'$)

Wenn Übertrag 1 und keine unmarkierte Ziffer hinter :
→ Fahre zur letzten unmarkierten Ziffer des ersten Wortes,
ziehe 1 ab.

.

Turing-Maschinen: Design

Merke:

- Gelesene Zeichen in Index speichern hilft, zu wissen, was man eigentlich tun will!
- Gelesene Zeichen als gelesen markieren entspricht “sich merken, wo man eigentlich gerade war” .

.

Strukturelle Induktion - Wörter

Induktionsanfang: Zeige: X gilt für $w = \epsilon$.

Induktionsvoraussetzung: Schreibe: X gilt für beliebiges, aber festes $w \in A^*$.

Induktionsschritt: Schreibe: Sei $x \in A$ beliebig.

Zeige: Dann gilt X auch für wx .

Strukturelle Induktion - Strukturen

Es gibt atomare Elemente und Operationen, die aus maximal k Elementen ein größeres Element “zusammensetzen”.

Induktionsanfang: Zeige: X gilt für **alle** atomaren Elemente.

Induktionsvoraussetzung: Schreibe: X gilt für beliebige, aber feste Elemente e_1, \dots, e_k .

Induktionsschritt: Zeige für **jede** Operation \circ mit $j \leq k$

Argumenten:

Dann gilt X auch für $\circ(e_1, e_2, \dots, e_j)$.

Strukturelle Induktion - Strukturen

$$Z \subseteq \mathbb{N}_0$$

$$10, 15 \in Z$$

$$z_1, z_2 \in Z \Rightarrow z_1 \cdot z_2 \in Z \wedge z_1 + z_2 \in Z$$

Keine anderen Zahlen in Z .

.

Strukturelle Induktion - Strukturen

$$Z \subseteq \mathbb{N}_0$$

$$10, 15 \in Z$$

$$z_1, z_2 \in Z \Rightarrow z_1 \cdot z_2 \in Z \wedge z_1 + z_2 \in Z$$

Keine anderen Zahlen in Z .

Zeige: Alle Zahlen $z \in Z$ durch 5 teilbar!

.

Strukturelle Induktion - Strukturen

$$Z \subseteq \mathbb{N}_0$$

$$10, 15 \in Z$$

$$z_1, z_2 \in Z \Rightarrow z_1 \cdot z_2 \in Z \wedge z_1 + z_2 \in Z$$

Keine anderen Zahlen in Z .

IA: 10, 15 durch 5 teilbar ✓

.

Strukturelle Induktion - Strukturen

$$Z \subseteq \mathbb{N}_0$$

$$10, 15 \in Z$$

$$z_1, z_2 \in Z \Rightarrow z_1 \cdot z_2 \in Z \wedge z_1 + z_2 \in Z$$

Keine anderen Zahlen in Z .

IV: Für beliebige, aber feste $z_1, z_2 \in Z$ gilt: z_1, z_2 durch 5 teilbar.

.

Strukturelle Induktion - Strukturen

$$Z \subseteq \mathbb{N}_0$$

$$10, 15 \in Z$$

$$z_1, z_2 \in Z \Rightarrow z_1 \cdot z_2 \in Z \wedge z_1 + z_2 \in Z$$

Keine anderen Zahlen in Z .

IS: Dann gilt auch $z_1 + z_2$ durch 5 teilbar und $z_1 \cdot z_2$ durch 5 teilbar (offensichtlich).

.

Turing-Maschinen: Verstehen

	B	L_b	A	L_a	U
a	$(a, B, 1)$	$(a, L_b, -1)$	$(\bar{a}, L_a, -1)$	$(a, L_a, -1)$	$(a, U, -1)$
b	$(\bar{b}, L_b, -1)$	$(b, L_b, -1)$	$(b, A, 1)$	$(b, L_a, -1)$	$(b, U, -1)$
\bar{a}	$(\bar{a}, B, 1)$	$(\bar{a}, L_b, -1)$	$(\bar{a}, A, 1)$	$(\bar{a}, L_a, -1)$	$(\bar{a}, U, -1)$
\bar{b}	$(\bar{b}, B, 1)$	$(\bar{b}, L_b, -1)$	$(\bar{b}, A, 1)$	$(\bar{b}, L_a, -1)$	$(b, U, -1)$
\square	$(\square, U, -1)$	$(\square, A, 1)$	$(\square, D, -1)$	$(\square, B, 1)$	$(\square, B, 1)$

	D	I_b	R
a	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
b	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
\bar{a}	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
\bar{b}	$(\square, I_b, -1)$	$(\bar{b}, I_b, -1)$	$(\bar{b}, R, 1)$
\square	-	-	$(\square, D, -1)$

Turing-Maschinen: Verstehen

Schritt 1: Suche Zustände, die einfach nur nach rechts oder links durchlaufen, ohne etwas zu ändern!

.

Turing-Maschinen: Verstehen

Schritt 1: Suche Zustände, die einfach nur nach rechts oder links durchlaufen, ohne etwas zu ändern!

Das heißt: $\forall x \in X : (z, x) \mapsto (x, z, 1)$

oder $\forall x \in X : (z, x) \mapsto (x, z, -1)$

.

Turing-Maschinen: Verstehen

	B	L_b	A	L_a	U
a	$(a, B, 1)$	$(a, L_b, -1)$	$(\bar{a}, L_a, -1)$	$(a, L_a, -1)$	$(a, U, -1)$
b	$(\bar{b}, L_b, -1)$	$(b, L_b, -1)$	$(b, A, 1)$	$(b, L_a, -1)$	$(b, U, -1)$
\bar{a}	$(\bar{a}, B, 1)$	$(\bar{a}, L_b, -1)$	$(\bar{a}, A, 1)$	$(\bar{a}, L_a, -1)$	$(\bar{a}, U, -1)$
\bar{b}	$(\bar{b}, B, 1)$	$(\bar{b}, L_b, -1)$	$(\bar{b}, A, 1)$	$(\bar{b}, L_a, -1)$	$(b, U, -1)$
\square	$(\square, U, -1)$	$(\square, A, 1)$	$(\square, D, -1)$	$(\square, B, 1)$	$(\square, B, 1)$

	D	I_b	R
a	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
b	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
\bar{a}	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
\bar{b}	$(\square, I_b, -1)$	$(\bar{b}, I_b, -1)$	$(\bar{b}, R, 1)$
\square	-	-	$(\square, D, -1)$

Turing-Maschinen: Verstehen

Schritt 2: Suche Zustände, die einfach nur nach rechts oder links durchlaufen, bis sie ein bestimmtes Zeichen y gefunden haben

.

Turing-Maschinen: Verstehen

Schritt 2: Suche Zustände, die einfach nur nach rechts oder links durchlaufen, bis sie ein bestimmtes Zeichen y gefunden haben

Das heißt: $\forall x \in X : x \neq y \Rightarrow (z, x) \mapsto (x, z, 1)$

oder $\forall x \in X : x \neq y \Rightarrow (z, x) \mapsto (x, z, -1)$

.

Turing-Maschinen: Verstehen

Schritt 2: Suche Zustände, die einfach nur nach rechts oder links durchlaufen, bis sie ein bestimmtes Zeichen y gefunden haben

Das heißt: $\forall x \in X : x \neq y \Rightarrow (z, x) \mapsto (x, z, 1)$

oder $\forall x \in X : x \neq y \Rightarrow (z, x) \mapsto (x, z, -1)$

(Falls Zeichen nicht gefunden wird, landet TM in Zustand z auf \square)

.

Turing-Maschinen: Verstehen

	<i>B</i>	<i>L_b</i>	<i>A</i>	<i>L_a</i>	<i>U</i>
<i>a</i>	$(a, B, 1)$	$(a, L_b, -1)$	$(\bar{a}, L_a, -1)$	$(a, L_a, -1)$	$(a, U, -1)$
<i>b</i>	$(\bar{b}, L_b, -1)$	$(b, L_b, -1)$	$(b, A, 1)$	$(b, L_a, -1)$	$(b, U, -1)$
\bar{a}	$(\bar{a}, B, 1)$	$(\bar{a}, L_b, -1)$	$(\bar{a}, A, 1)$	$(\bar{a}, L_a, -1)$	$(\bar{a}, U, -1)$
\bar{b}	$(\bar{b}, B, 1)$	$(\bar{b}, L_b, -1)$	$(\bar{b}, A, 1)$	$(\bar{b}, L_a, -1)$	$(b, U, -1)$
□	$(\square, U, -1)$	$(\square, A, 1)$	$(\square, D, -1)$	$(\square, B, 1)$	$(\square, B, 1)$

	<i>D</i>	<i>I_b</i>	<i>R</i>
<i>a</i>	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
<i>b</i>	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
\bar{a}	$(\square, D, -1)$	$(\bar{b}, R, 1)$	-
\bar{b}	$(\square, I_b, -1)$	$(\bar{b}, I_b, -1)$	$(\bar{b}, R, 1)$
□	-	-	$(\square, D, -1)$

Turing-Maschinen: Verstehen

Erkenntnis:

In Zustand B sucht TM nach b , markiert dieses, geht zurück nach links,
sucht a , markiert dieses, geht nach links zurück,
wiederhole, so lange es a und b gibt.

.

Turing-Maschinen: Verstehen

Was machen die restlichen Zustände?

.

Turing-Maschinen: Verstehen

Was machen die restlichen Zustände?

U : Hebt Markierungen von b auf und geht nach links zurück.

.

Turing-Maschinen: Verstehen

Was machen die restlichen Zustände?

U : Hebt Markierungen von b auf und geht nach links zurück.

D : Löscht alle Zeichen $\neq \bar{b}$, merkt sich \bar{b} .

.

Turing-Maschinen: Verstehen

Was machen die restlichen Zustände?

U : Hebt Markierungen von b auf und geht nach links zurück.

D : Löscht alle Zeichen $\neq \bar{b}$, merkt sich \bar{b} .

I_b : Geht nach links durch die \bar{b} und ersetzt erstes Zeichen $\neq \bar{b}$ durch \bar{b} .

.

Turing-Maschinen: Verstehen

Was machen die restlichen Zustände?

U : Hebt Markierungen von b auf und geht nach links zurück.

D : Löscht alle Zeichen $\neq \bar{b}$, merkt sich \bar{b} .

I_b : Geht nach links durch die \bar{b} und ersetzt erstes Zeichen $\neq \bar{b}$ durch \bar{b} .

R : Geht nach rechts durch die \bar{b}

.

Turing-Maschinen: Verstehen

Beispiel:

Baaaababb

(Schreibweise: Zustand steht vor Symbol, auf dem sich Kopf befindet).

.

Turing-Maschinen: Verstehen

Beispiel:

$$Baaaababb \rightarrow^* aaL_b a\bar{b}abb$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$Baaaababb \rightarrow^* aaL_b a\bar{b}abb \rightarrow^* Aaaa\bar{b}abb$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ \rightarrow^* B\bar{a}a\bar{a}babb$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} Baaababb &\rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}aa\bar{b}abb \\ &\rightarrow^* B\bar{a}aa\bar{b}abb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}aa\bar{b}abb \\ \rightarrow^* B\bar{a}aa\bar{b}abb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} Baaababb &\rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}aa\bar{b}abb \\ &\rightarrow^* B\bar{a}aa\bar{b}abb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} Baaababb &\rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ &\rightarrow^* B\bar{a}a\bar{a}babb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} Baaababb &\rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ &\rightarrow^* B\bar{a}a\bar{a}babb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}Db \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} &Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ &\rightarrow^* B\bar{a}a\bar{a}babb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}Db \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}I_b\bar{a} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} &Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ &\rightarrow^* B\bar{a}a\bar{a}babb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}Db \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}I_b\bar{a} \\ &\rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}D\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} &Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ &\rightarrow^* B\bar{a}a\bar{a}babb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}Db \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}I_b\bar{a} \\ &\rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}D\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}I_b\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} &Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaaa\bar{b}abb \rightarrow^* L_a\Box\bar{a}aa\bar{b}abb \\ &\rightarrow^* B\bar{a}aa\bar{b}abb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}Db \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}I_b\bar{a} \\ &\rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}D\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}I_b\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{b}R\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} &Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaaa\bar{b}abb \rightarrow^* L_a\Box\bar{a}aa\bar{b}abb \\ &\rightarrow^* B\bar{a}aa\bar{b}abb \rightarrow^* B\bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}a\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}\bar{a}\bar{b}Db \rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}I_b\bar{a} \\ &\rightarrow^* \bar{a}\bar{a}\bar{a}\bar{b}D\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}I_b\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{b}R\bar{b} \\ &\rightarrow^* \bar{a}\bar{a}\bar{b}D\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Beispiel:

$$\begin{aligned} &Baaababb \rightarrow^* aaL_b\bar{a}babb \rightarrow^* Aaa\bar{a}babb \rightarrow^* L_a\Box\bar{a}a\bar{a}babb \\ &\rightarrow^* B\bar{a}a\bar{a}babb \rightarrow^* B\bar{a}\bar{a}\bar{a}b\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}b\bar{a}\bar{b}U\bar{b} \\ &\rightarrow^* B\bar{a}\bar{a}\bar{a}babb \rightarrow^* A\bar{a}\bar{a}\bar{a}b\bar{a}\bar{b}\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}b\bar{a}\bar{b}Db \rightarrow^* \bar{a}\bar{a}\bar{a}bI_b\bar{a} \\ &\rightarrow^* \bar{a}\bar{a}\bar{a}bD\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{a}I_b\bar{b} \rightarrow^* \bar{a}\bar{a}\bar{b}R\bar{b} \\ &\rightarrow^* \bar{a}\bar{a}\bar{b}D\bar{b} \rightarrow^* I_b\Box\bar{b} \end{aligned}$$

.

Turing-Maschinen: Verstehen

Allgemein:

Solange (Anzahl unmarkierte $a \geq$ Anzahl b): Markiere so viele a , wie es b gibt.

.

Turing-Maschinen: Verstehen

Allgemein:

Solange (Anzahl unmarkierte $a \geq$ Anzahl b): Markiere so viele a , wie es b gibt.

Falls (Anzahl unmarkierte $a <$ Anzahl b):

Markiere ein mehr b , als es unmarkierte a gibt.

.

Turing-Maschinen: Verstehen

Allgemein:

Solange (Anzahl unmarkierte $a \geq$ Anzahl b): Markiere so viele a , wie es b gibt.

Falls (Anzahl unmarkierte $a <$ Anzahl b):

Markiere ein mehr b , als es unmarkierte a gibt.

Lösche Zeichen außer markierte b

.

Turing-Maschinen: Verstehen

Allgemein:

Solange (Anzahl unmarkierte $a \geq$ Anzahl b): Markiere so viele a , wie es b gibt.

Falls (Anzahl unmarkierte $a <$ Anzahl b):

Markiere ein mehr b , als es unmarkierte a gibt.

Lösche Zeichen außer markierte b

Setze letztes markiertes b an Anfang des \bar{b} Blockes

.

Turing-Maschinen: Verstehen

Allgemein:

Solange (Anzahl unmarkierte $a \geq$ Anzahl b): Markiere so viele a , wie es b gibt.

Falls (Anzahl unmarkierte $a <$ Anzahl b):

Markiere ein mehr b , als es unmarkierte a gibt.

Lösche Zeichen außer markierte b

Setze letztes markiertes b an Anfang des \bar{b} Blockes
oder lösche letztes markiertes b , falls kein Zeichen un-
gleich \bar{b} vor dem Block steht.

.

Turing-Maschinen: Verstehen

Allgemein: n mal a , k mal b :

Markiere alle a und $n \bmod k + 1$ b

Lösche Zeichen außer markierte b

Schiebe alle bis auf ein \bar{b} am Anfang des Wortes zusammen.

.

Turing-Maschinen: Verstehen

Allgemein: n mal a , k mal b :

Markiere alle a und $n \bmod k + 1$ b

Lösche Zeichen außer markierte b

Schiebe alle bis auf ein \bar{b} am Anfang des Wortes zusammen.

→ Am Ende steht $\bar{b}^{n \bmod k}$ auf dem Band.

.

Lerntipp für Klausur

Überlegen Sie sich, wie sie ein Problem mit einer Turingmaschine lösen würden!

Entwerfen Sie eine Turingmaschine!

Tauschen Sie Turingmaschinen untereinander und finden Sie heraus, was die Turingmaschinen Ihrer Kommilitonen machen!

.