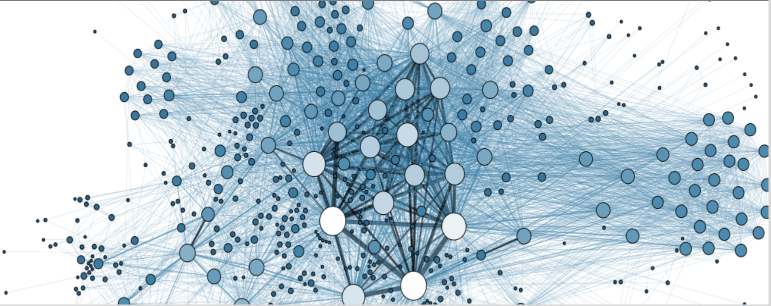
Grundbegriffe der Informatik
Tutorium 33

Lukas Bach, lukas.bach@student.kit.edu | 09.02.2017



Lukas Bach, lukas.bach@student.kit.edu

Gliederung

Lukas Bach, lukas.bach@student.kit.edu

Turingmaschinen

Karlsruher Institut für Technologie

Was sind Turingmaschinen?

- Sehr mächtige Erweiterung Automat
  - Was heißt mächtig?
  - Turingmaschinen können eine große Vielfalt von Problemen lösen, einschließlich vieler in GBI besprochener Probleme
- Gesteuert durch einen endlichen Automaten, aber mit einem unendlichen Arbeitsband zum Zwischenspeichern von Informationen
- Besitzen einen Kopf um auf dem Band zu lesen und zu schreiben
- Turingmaschinen sind sozusagen genauso mächtig wie Computer
  - können also benutzt werden, um für Probleme zu entscheiden, ob sie gelöst werden können oder nicht

Definition von Turingmaschinen

Lukas Bach, lukas.bach@student.kit.edu

Definition von Turingmschinen

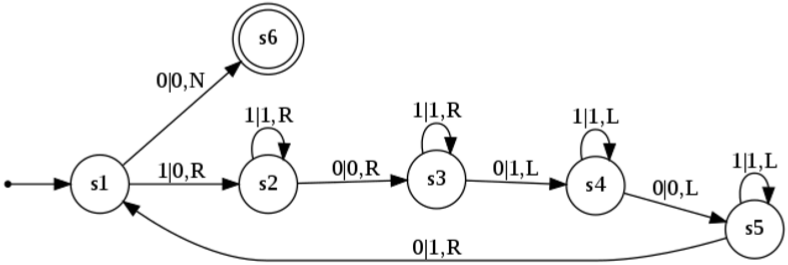
Eine Turingmaschine $T = (Z, z_0, X, f, g, m)$ besteht aus:

- Z Zustandsmenge
- $z_0 \in Z$ Startzustand
- X Bandalphabet
- $\square$ Blanksymbol (sozusagen Markierung für Leerzeichen)
- $f : Z \times X \to Z$ partielle Zustandsübergangsfunktion
- $g : Z \times X \to X$ partielle Ausgabefunktion
- $m : Z \times X \to \{L, N, R\}$ partielle Bewegungsfunktion

Anmerkung: partielle Funktionen sind nicht linkstotal, also manche Elemente des Definitionsbereichs werden nicht abgebildet.

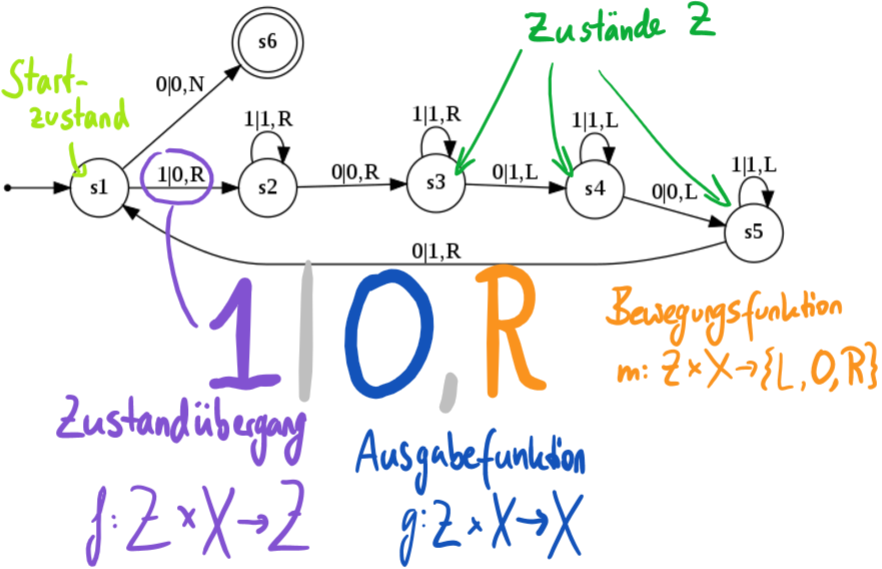
Lukas Bach, lukas.bach@student.kit.edu

Beispiel einer Turingmaschine



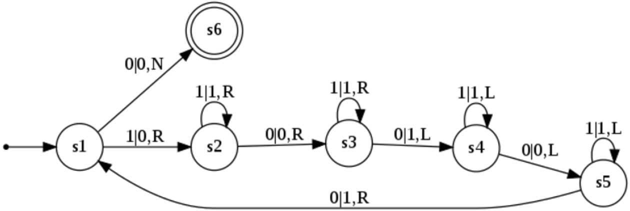
Lukas Bach, lukas.bach@student.kit.edu

Beispiel einer Turingmaschine



Lukas Bach, lukas.bach@student.kit.edu

Funktionen von Turing Maschinen



Wie sehen die konkreten Abbildungsvorschriften der linken vier Pfeile aus?

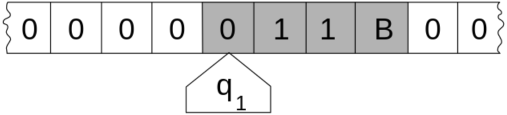
- $f : Z \times X \to Z$
- $g : Z \times X \to X$
- $m : Z \times X \to \{L, N, R\}$

	f	g	m
$0\|0, N$	$(s1, 0) \mapsto s6$	$(s1, 0) \mapsto 0$	$(s1, 0) \mapsto N$
$1\|0, R$	$(s1, 1) \mapsto s2$	$(s1, 1) \mapsto 0$	$(s1, 1) \mapsto R$
$1\|1, R$	$(s2, 1) \mapsto s2$	$(s2, 1) \mapsto 1$	$(s2, 1) \mapsto R$
$0\|0, R$	$(s2, 1) \mapsto s3$	$(s2, 1) \mapsto 0$	$(s2, 1) \mapsto R$

Lukas Bach, lukas.bach@student.kit.edu

Das Band einer Turingmaschine

- Unendliche Anreihung von Zeichen, die nach links und rechts unbegrenzt weiter geht



- Die Turingmaschine hat einen Kopf, mit dem sie das aktuelle Zeichen lesen oder überschreiben kann, oder kann ihn nach links oder rechts bewegen.
- Das Band einer Turingmaschine wird benutzt als...
  - Erhalten der Eingabe: Bevor die Turingmaschine startet, steht das Eingabewort auf dem Band, der Kopf steht auf dem ersten Zeichen der Eingabe.
  - Rückgabe der Ausgabe: Nach Beenden steht auf dem Band die Ausgabe (und der Kopf irgendwo).
  - Zwischenspeicher: Die Turingmaschine kann überall Informationen zwischenspeichern, diese müssen von der TM am Ende aber gelöscht werden.

Beispielabarbeitungen

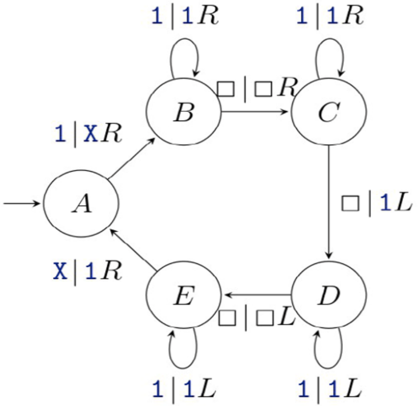
Lukas Bach, lukas.bach@student.kit.edu

Gemeinsame Übung

Arbeite folgende Wörter mit der Turingmaschine ab:

- 0
- 1
- 11
- 111

Was macht die Turingmaschine?



Die Turingmaschine macht aus 1^k die Ausgabe $1^k \square 1^k$.

Lukas Bach, lukas.bach@student.kit.edu

Halten von Turingmaschinen

Halten einer Turingmaschine

Wenn eine Turingmaschine in einem Zustand ist, für den das nächste Eingabezeichen durch die Übergangsfunktion f nicht definiert ist, hält die Maschine.

Wann hält also eine Turingmaschine nicht?

Nicht-Halten einer Turingmmaschine

Wenn eine Turingmaschine in eine endlose Schleife gerät, so hält sie nicht.

Entscheidbarkeit

Lukas Bach, lukas.bach@student.kit.edu

Durch Turingmaschine akzeptierte Sprache

Eine Turingmaschine akzeptiert eine formale Sprache L, wenn sie für jedes Wort $w \in L$ in einem akzeptierenden Zustand hält.

Entscheidbare Sprache

Eine formale Sprache L heißt entscheidbar, wenn es eine Turingmaschine gibt, die immer hält und L akzeptiert.

Aufzählbare Sprache

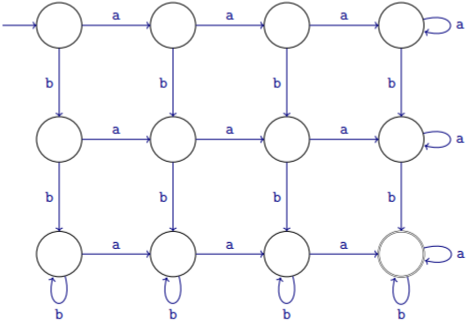
Eine formale Sprache L heißt aufzählbar, wenn es eine Turingmaschine gibt, die L akzeptiert.

Vom endlichen Akzeptor zur Turingmaschine

Lukas Bach, lukas.bach@student.kit.edu

Akzeptieren von Turingmaschinen

Wie kann man aus dem gegebenen endlichen Akzeptor eine Turingmaschine machen, die dieselbe Sprache akzeptiert?



Lukas Bach, lukas.bach@student.kit.edu

Lösung

Einfach gesagt: mache aus jedem Übergang a einen Turingmaschinen-Übergang der Art $a|a, R$, also bei jedem Zeichen mache den Zustandsübergang, überschreibe aber das Zeichen nicht und gehe zum nächsten Zeichen.

Formaler ausgedrückt?

- Für allgemeinen endlichen Akzeptor (Z, z_0, X, f, Y, h), definiere eine Turingmaschine $T := (Z, z_0, X \cup Y, f, g, h)$, also Z, z_0, f gleich und mit Bandalphabet $=$ Eingabealphabet $\cup$ Ausgabealphabet
- $g(z, x) := x \quad \forall (z, x)$ in f definiert
- $m(z, x) := R \quad \forall (z, y)$ in f definiert

Jeder endliche Akzeptor kann so zu einer Turingmaschine umgeformt werden, die dieselbe Sprache akzeptiert.

Über endliche Akzeptoren hinaus

Lukas Bach, lukas.bach@student.kit.edu

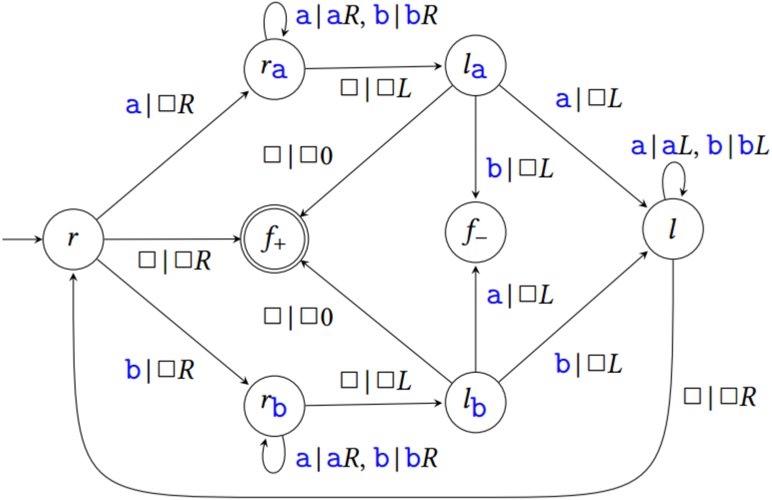
Sei L die Sprache von Palindromen über $\{a, b\}$ ($L = \{aabaa, bbababb, aa, \varepsilon\}$).

- Ist die Sprache regulär, also gibt es einen endlichen Akzeptor, der diese akzeptiert? Nein.
- Ist die Sprache entscheidbar, also gibt es eine stets haltende Turingmaschine, die L akzeptiert?

Lukas Bach, lukas.bach@student.kit.edu

Palindromerkennung mit Turingmaschinen

Ja, nämlich:



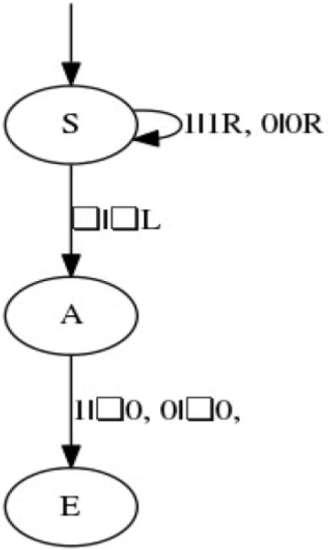
Lukas Bach, lukas.bach@student.kit.edu

Turingmaschinen Entwurfsaufgabe

Turingmaschine Entwurf

Entwerfe eine Turingmaschine, die...

- als Eingabe eine Binärzahl auf dem Band erhält
- als Ausgabe diese Zahl restlos durch zwei teilt und auf dem Band stehen lässt



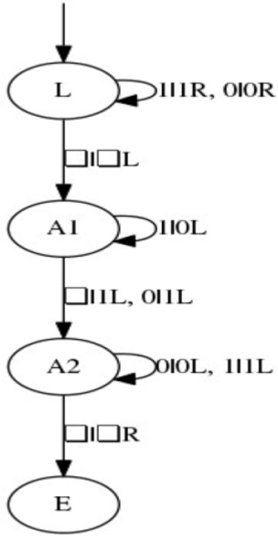
Turingmaschinen Entwurfsaufgabe

Lukas Bach, lukas.bach@student.kit.edu

Turingmaschine Entwurf

Entwerfe eine Turingmaschine, die...

- als Eingabe eine Binärzahl auf dem Band erhält
- als Ausgabe diese Zahl um eins erhöht auf dem Band stehen lässt
- den Kopf der Turingmaschine auf dem ersten Zeichen der Ausgabe stehen hat.

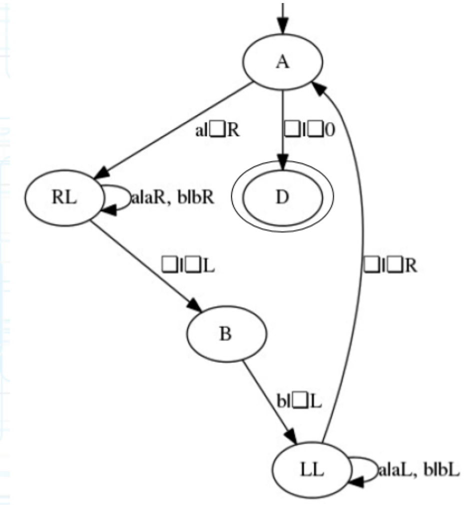


Turingmaschine Entwurfsaufgabe

Lukas Bach, lukas.bach@student.kit.edu

Turingmaschine Entwurf

Entwerfe eine Turingmaschine, die die Sprache $\{a^k b^k : k \in \mathbb{N}_0\}$ erkennt.



Lukas Bach, lukas.bach@student.kit.edu

Konfiguration von Turingmaschinen

Angenommen, man kennt eine Turingmaschine, hat mit der Abarbeitung eines Wortes angefangen, will aber pausieren, um später weiterzumachen...

Was muss man sich alles merken, um später weiter zu machen?

- Derzeitiger Zustand, in dem die Turingmaschine steht
- Inhalt des Bandes

Konfiguration von Turingmaschinen

Wenn während dem Arbeiten einer Turingmaschine auf dem Band das Wort $w_1 a w_2$ mit $w_1, w_2 \in X^*, a \in X$ steht, der Kopf der Turingmaschine auf das Zeichen a zeigt und die Turingmaschine im Zustand z ist, so schreibt man die Konfiguration der Turingmaschine als $\square w_1 (z) a w_2 \square$.

Lukas Bach, lukas.bach@student.kit.edu

Konfiguration von Turingmaschinen

Beispiel:

$$\square\square\square\square\square\square\square\square abcbabb\textcolor{blue}{d}aabc\square\square\square\square\square$$

$$\uparrow$$

$$KOPF$$

...sei das Band der Turingmaschine während Abarbeitung der Eingabe, dazu steht sie im Zustand z_4.

Dann sieht sieht die Konfiguration der Turingmaschine so aus: $\square abcbabb(z_4)daabc\square$

Grundbegriffe der Informatik

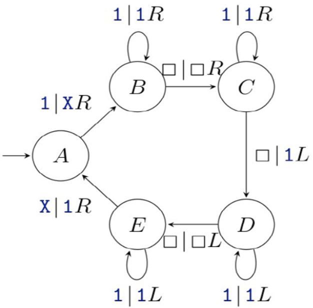
Lukas Bach, lukas.bach@student.kit.edu

Turingmaschinen

Dokumentation einer Abarbeitung mit Konfigurationen

Aufgabe zu Konfigurationen

Gebe alle Konfigurationen der Turingmaschine bei Abarbeitung des Wortes 11 an.



$\square(A)11\square$

$\rightarrow \square X(B)1\square$

$\rightarrow \square X1(B)\square$

$\rightarrow \square X1\square(C)\square$

$\rightarrow \square X1(D)\square 1\square$

$\rightarrow \square X(E)1\square 1\square$

$\rightarrow \square(E)X1\square 1\square$

$\rightarrow \square 1(A)1\square 1\square$

$\rightarrow \square 1X(B)\square 1\square$

$\rightarrow \square 1X\square(C)1\square$

$\rightarrow \square 1X\square 1(C)\square$

$\rightarrow \square 1X\square(D)11\square$

$\rightarrow \square 1X(D)\square 11\square$

$\rightarrow \square 1(E)X\square 11\square$

$\rightarrow \square 11(A)\square 11\square$

Lukas Bach, lukas.bach@student.kit.edu

Halteproblem

Halteproblem: Für einen gegebenen Algorithmus, gelingt dieser bei seiner Abarbeitung zu einem Ende und hält?

- Algorithmen können durch Turingmaschinen durchgeführt werden
- Turingmaschinen können durch sogenannte universelle Turingmaschinen simuliert werden
  - Wenn eine Turingmaschine T kodiert ist mit dem Wort w, dann ist $T_w : X \longrightarrow X$ eine Funktion, die Eingaben auf die Ausgabe der Turingmaschine T mappt.
  - Also mit $X = \{1, 0\}$ gibt z.B. $T_w(100101) = 001$ genau dann zurück, wenn, sofern man 100101 als Eingabe an die Turingmaschine mit der Kodierung w gibt, diese hält und als Ausgabe 001 erzeugt.

Dann lässt sich das Halteproblem auch als Sprache formulieren:

$$H = \{w \in A^* : w \text{ ist eine TM-Codierung und } T_w(w) \text{ hält.}\}$$

bzw. als allgemeinerer Fall:

$$\widehat{H} = \{(w, x) \in A^* \times A^* : w \text{ ist eine TM-Codierung und } T_w(x) \text{ hält.}\}$$

Lukas Bach, lukas.bach@student.kit.edu

Halteproblem

Das Halteproblem ist unentscheidbar, dh. es gibt keine Turingmaschine, die H entscheidet.

Lukas Bach, lukas.bach@student.kit.edu

Busy Beaver

Busy Beaver TM ist eine Turingmaschine mit n Zuständen, die möglichst viele Einsen auf das Band schreibt und hält.

- Also nicht einfach ewig Einsen aufschreibt und nie aufhört.

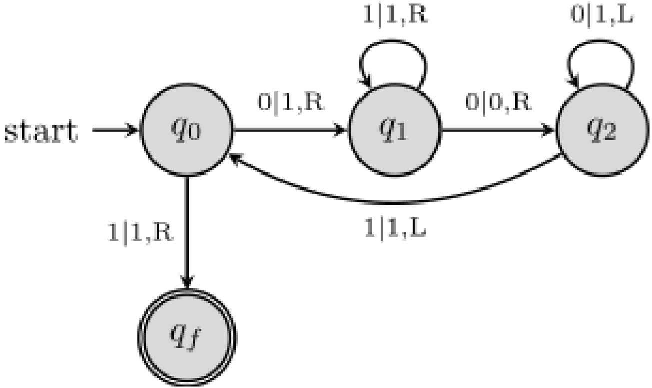
Busy Beaver Problem: Für eine gegebene Turingmaschine mit n Zuständen, die Einsen aufschreibt und hält: Schreibt sie auch maximal viele Einsen auf?

Das Busy Beaver Problem ist nicht entscheidbar, bzw. die Busy Beaver Funktion $bb(n)$, die definiert, wieviele einsen von einer Busy Beaver TM maximal geschrieben werden können, ist nicht berechenbar.

Beispielwerte von bb:
$bb(1) = 1, bb(2) = 4, bb(5) \geq 4098, bb(6) > 10^{18276}$.

Busy Beaver für $n = 3$

Lukas Bach, lukas.bach@student.kit.edu



Organisatorisches

Lukas Bach, lukas.bach@student.kit.edu

- Alle Folien und Folienpaket jetzt online.
- Fragen zur Klausur oder zur Vorbereitung?

Informationen

Lukas Bach, lukas.bach@student.kit.edu

Zum Tutorium

- Lukas Bach
- Tutorienfolien auf:
  - `http://gbi.lukasbach.com`
- Tutorium findet statt:
  - Donnerstags, 14:00 - 15:30
  - 50.34 Informatikbau, -107

Mehr Material

- Ehemalige GBI Webseite:
  - `http://gbi.ira.uka.de`
  - Altklausuren!

Zur Veranstaltung

- Grundbegriffe der Informatik
- Klausurtermin:
  - 06.03.2017, 11:00
  - Zwei Stunden Bearbeitungszeit
  - 6 ECTS für Informatiker und Informationswirte, 4 ECTS für Mathematiker und Physiker

Zum Übungsschein

- Übungsblatt jede Woche
- Ab 50% insgesamt hat man den Übungsschein
- Keine Voraussetzung für die Klausur, aber für das Modul