



Turingmachine

A Turing machine is like a wise old person, sitting at an endless table, playing a complex game. They have a magical pen that reads and writes on the game board. They follow strict rules, do not move from their spot, but the table mysteriously moves back and forth. Their concentration is deep and calm as they perform a complex ballet of reading, writing, and state-changing.

- ChatGPT

Berechenbarkeit

1 Turingmaschine

Wir betrachte das folgende, sehr bekannt, berechnungsmodell. Anschaulich lässt es sich wie folgt beschreiben.

- Es gibt einen "Speicher" \rightsquigarrow k unendlich lange Arrays(**Bänder**)
- Es gibt einen "Arbeitsspeicher" \rightsquigarrow eine endliche Menge von Zuständen, die die Maschine einnehmen kann
- Für jedes Band gibt es einen Schreib- und Lesekopf
- Jeder Schritt ist wie folgt:
Abhängig von Zustand und gelesenen Symbol, Schreiben die Köpfe genau ein Symbol, bewegen sich nun maximal eine Position und der Zustand der Maschine wird geändert.
- Stellt die Maschine ihr schrittweises Arbeiten ein, so wird die Ausgabe entweder den Zustand entnommen oder von einem der Bänder in geeigneter Weise abgelesen.

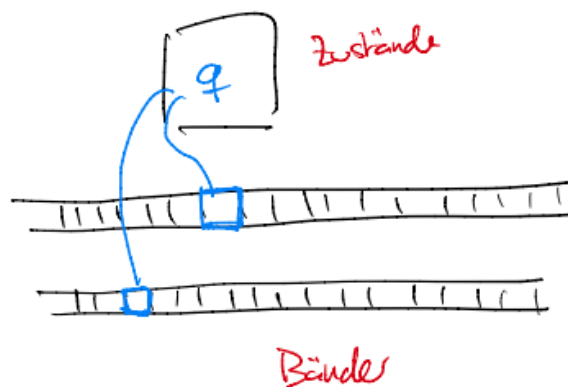


Figure 1: Turingmaschine.

1.1 Definition (Turingmaschine, Alan Turing, 1936)

Sei $k \in \mathbb{N}$ eine **k-Band-Turingmaschine** kurz k-TM, ist ein Tupel $M = (Q, \Sigma, \Gamma, \Delta, s, F)$. Dabei ist:

- Q eine endliche Menge, **Zustandsmenge**
- Σ das **Eingabealphabet**, ein Alphabet $\square \notin \Sigma$
- Γ das **Bandalphabet**, ein Alphabet mit $\Sigma \subseteq \Gamma$ und $\square \in \Gamma \setminus \Sigma$
- $\Delta \subseteq Q \times \Gamma^k \rightarrow \subseteq Q \times \Gamma^k \times L, S, R^k$ die **Übergangsrelation**
- $s \in Q$ der **Startzustand**
- $F \subseteq Q$ die Menge der **akzeptierenden Zustände**

Das Symbol \square heißt **Blank**. Die Elemente von Δ heißen **Instruktionen**. Für eine Instruktion $(q_1, a_1, \dots, a_k, q', a'_1, \dots, a'_k, B_1, \dots, B_k)$ **Anweisungsteil**. Die TM M ist eine **deterministische k-Band Turingmaschine**, kurz k-DTM, wenn es $\forall b \in Q \times \Gamma^k$ höchstens eine Instruktion $i \in \Delta$ mit Bedingungsteil b .

1.2 Definition (Konfiguration)

Sei $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ eine k-TM. Eine **Konfiguration** von M ist ein Tupel

$$C = (q, w_1, \dots, w_k, p_1, \dots, p_k) \in Q \times (p^*)^k \times \mathbb{N}^k$$

Die **Startkonfiguration** von M zur Eingabe $(u_1, \dots, u_n) \in (\Sigma^*)^n$, wobei $n \in \mathbb{N}$, ist die Konfiguration

$$Start_M(u_1, \dots, u_n) = (s, u_1 \sqcup u_2 \sqcup \dots \sqcup u_n, \square, \dots, 1, \dots, 1)$$

Die Konfiguration C ist eine **Stoppkonfiguration** von M , wenn es keine Instruktion $i \in \Delta$ mit Bedingungssteil $(q, w_1(p_1), \dots, w_k(p_k))$ gibt.

1.3 Definition (Nachfolgekonfiguration)

Sei $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ eine k-DTM. Für Konfiguration $C = q_1, w_1, \dots, w_k, p_1, \dots, p_k$ und $C' = q'_1, w'_1, \dots, w'_k, p'_1, \dots, p'_k$ von M ist die Konfiguration C' Nachfolgekonfiguration von C , wenn es eine Instruktion

$$(q, w_1(p_1), \dots, w_k(p_k), a'_1, a'_k, B_1, \dots, B_k) \in \Delta$$

gibt, sodass

$$w'_i = \begin{cases} \square a'_i w_i(2) \dots w_i(|w_i|), & \text{falls } p_i = 1 \text{ und } B_i = L \\ w_i \dots w_i(|w_i| - 1) a'_i \square, & \text{falls } p_i = |w_i| \text{ und } B_i = R \\ w_i \dots w_i(p_i - 1) a'_i w_i(p_i + 1) \dots w_i(|w_i|), & \text{sonst} \end{cases}$$

und

$$p'_i = \begin{cases} 1, & \text{falls } p_i = 1 \text{ und } B_i = L \\ p_i - 1, & \text{falls } p_i \geq 2 \text{ und } B_i = L \\ p_i, & \text{falls } B_i = S \\ p_i + 1, & \text{falls } B_i = R \end{cases}$$

$\forall i \in [k]$ gelten.

Es bezeichnen \rightarrow_M die Relation auf der Menge der Konfiguration von M , sodass $C \rightarrow_M C'$ falls C, C' Konfig von M sind wobei C' eine Nachfolgekonfiguration von C ist.

1.4 Definition (Rechnung)

Sei $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ eine k-DTM. Eine **endliche partielle Rechnung** von M ist eine endliche Folge C_1, \dots, C_n von Konfig von M mit $C_i \rightarrow_M C_{i+1} \forall i \in [n-1]$. Eine **unendliche partielle Rechnung** von M ist eine unendliche Folge C_1, C_2, \dots von Konfiguration von M mit $C_1 \rightarrow_M C_{1+i} \forall i \in \mathbb{N}$. Eine **Rechnung von M zur Eingabe** $(w_1, \dots, w_n) \in (\Sigma^*)^n$ (mit $n \in \mathbb{N}$) ist eine endliche partielle Rechnung $start_M = C_1, \dots, C_m$ bei der C_m eine Stoppkonfiguration von M oder eine unendliche partielle Rechnung $start_M(w_1, \dots, w_n) = C_1, C_2, \dots$

1.5 Bemerkung

Ist M eine k-DTM, so gilt es $\forall n \in \mathbb{N}$ und $(w_1, \dots, w_n) \in (\Sigma^*)^n$ genau eine Rechnung zur Eingabe (w_1, \dots, w_n) .

1.6 Definition (total)

Eine k-DTM $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ **terminiert** bei Eingabe $(w_1, \dots, w_n) \in (\Sigma^*)^n$ wenn die Rechnung von M zur Eingabe (w_1, \dots, w_n) endlich ist. Eine k-TM $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ ist **total**, wenn $\forall n \in \mathbb{N}$ und $(w_1, \dots, w_n) \in (\Sigma^*)^n$ alle Rechnungen von M zur Eingabe (w_1, \dots, w_n) endlich sind.

1.7 Definition (akzeptierte Sprache)

Sei $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ eine k -TM. Eine Stoppkonfiguration $(q, w_1, \dots, w_k, p_1, \dots, p_k)$ von M ist **akzeptierend**, wenn $q \in F$. Die **akzeptierte Sprache** $L(M)$ von M ist die Sprache über dem Alphabet Σ so dass $w \in L(M)$ gilt, wenn es eine endliche Rechnung C_1, \dots, C_n von M zur Eingabe w gibt, bei der C_n eine akzeptierende Stoppkonfiguration von M ist.

Hinweis: Für nicht deterministische TM heißt das insbesondere, dass es für die Wörter w in der akzeptierten Sprache nur mindestens **eine** im einer akzeptierten Stoppkonfiguration endende endliche Rechnung zur Eingabe w geben muss. Für Wörter w , die nicht in $L(M)$ sind, sind **alle** rechnungen von M zur Eingabe am Ende nicht in einer akzeptierten Stoppkonfiguration oder unendlich.

1.8 Definition(entscheidbar)

Eine Sprache L ist genau dann **entscheidbar**, wenn es eine totale k -TM M mit $L(M) = L$ gibt. Wir schreiben **REC** für die Klasse der entscheidbaren Sprachen. Der Begriff entscheidbar für Sprachen ergibt sich hier daraus, dass effektiv entschieden werden kann ob eine gegebene Eingabe in der Sprache liegt oder nicht. Insbesondere steht dies voraus, dass Eingabe, die nicht in der Sprache liegen effektiv als nicht in der Sprache liegend erkannt werden.

Begriff: effektiv \leadsto eine TM erledigt dies in endlicher Zeit. Da sich der durch TM formatierte Berechenbarkeitsbegriff, also die Formalisierung dessen was effektiv durchführbar ist, auch äquivalent durch rekursive Funktion definieren lässt, werden entscheidbare Sprachen auch als rekursiv bezeichnet.

1.9 Definition(rekursiv aufzählbar)

Eine Sprache L ist genau dann **rekursiv aufzählbar**, wenn es eine k -TM mit akzeptierter Sprache L gibt. Wir schreiben **RE** für die Klasse der rekursiv aufzählbaren Sprachen. Die Aufzählbarkeit leitet sich daraus ab, dass es für eine rekursiv aufzählbare Sprache L über einem Alphabet Σ möglich ist effektive Verfahren anzugeben, die die Wörter von L aufzählen, also dass eine endlich oder unendliche Aufzählung von $A = w_1, w_2, \dots$ mit $w_1, w_2, \dots = L$ existiert.

1.10 Bemerkung (normiert)

Eine 1-DTM $M = (Q, \Sigma, \Gamma, \Delta, s, F)$ heißt **normiert**, wenn $Q = 0, \dots, n$ für eine $n \in \mathbb{N}_0$, $\Sigma = 0, 1$, $\Gamma = \square, 0, 1$, $s = 0$, $F = s$. Alle TMs mit Eingabealphabet $0, 1$ lassen sich mit folgenden Schritten in eine normierte TM mit gleicher erkannter Sprache und gleicher berechneter Funktion umwandeln.

- Von Nichtdeterminismus zu Determinismus: Eine DTM kann die Rechnungen einer nichtdeterministischen TM parallel im Sinne von abwechselnd schrittweise durchführen um schließlich das Verhalten der simulierten TM zu ???. Dies entspricht einer **Breitensuche im Rechnungsbaum**.
ADD IMAGE HERE
- Von mehreren Bändern zu einem Band: Intuitiv können k Bänder auf ein Band simuliert werden, indem die Felder des einen Bandes in k -teilerfeldern unterteilt werden, die jeweils die gleiche Bandalphabetbuchstaben wie zuvor als Beschreibung zulassen und es zudem erlaubt zu markieren, dass der simulierte Kopf des simulierten Bandes dort steht. Eine dieser Idee folgende Konstruktion wird als **Spurentechnik** bezeichnet. Formal: Übergang vom Bandalphabet Γ zu

$$((\Gamma \cup \underline{a} : a \in \Gamma)^k / \square^k) \cup \square$$

wobei $\underline{a} \notin \Gamma$ für $a \in \Gamma$. Hierbei bedeutet \underline{a} , dass das simulierte Feld mit a beschriftet ist und dass dort der simulierte Kopf steht. Weiter spielt \square die Rolle des k -Tupels $(\square, \dots, \square)$ um der Tatsache gerecht zu werden, dass alle Felder zu Beginn mit \square beschriftet sind.