Definition:

Sei A ein Alphabet. Ein **Berechnungsproblem** ist eine Relation $R \subseteq A^* \times A^*$. $X_R = \{x \in A^* \mid \exists y \in A^* : (x, y) \in R\}$ ist die Menge der **Instanzen** von R.
Eine Turingmaschine Φ mit Alphabet A **berechnet** R, wenn für alle $x \in X_R$ gilt: $\text{time}(\Phi, x) < \infty$ und $(x, \text{output}(\Phi, x)) \in R$. Wenn es ein Polynom p gibt, so dass $\text{time}(\Phi, x) \leq p(\text{size}(x))$ für alle $x \in X_R$ gilt, dann heißt Φ **polynomielle Turingmaschine für** R, und R heißt **in polynomieller Zeit berechenbar**.

Wichtiger Spezialfall: Wenn R eine Funktion $R : A^* \to \{0, 1\}$ und in polynomieller Zeit berechenbar ist, dann heißt die Sprache $\{x \in A^* \mid R(x) = 1\}$ **in polynomieller Zeit entscheidbar**.

Definition:

Ein **Entscheidungsproblem** ist ein Paar $\mathcal{P} = (X, Y)$ mit $Y \subseteq X$, wobei X eine in polynomieller Zeit entscheidbare Sprache ist.

Definition: Sei A ein Alphabet und $\bar{A} = A \cup \{\sqcup\}$. Eine **Turingmaschine** ist eine Funktion

$$\Phi : \{0, \ldots, N\} \times \bar{A} \quad \to \quad \{-1, \ldots, N\} \times \bar{A} \times \{-1, 0, 1\}$$

für ein $N \in \mathbb{Z}_+$.
Die **Berechnung** von Φ für die Eingabe $x \in A^*$ ist die endliche oder unendliche Folge von Tripeln $(n^{(i)}, s^{(i)}, \pi^{(i)}) \in \{-1, \ldots, N\} \times \bar{A}^{\mathbb{Z}} \times \mathbb{Z}$, die wie folgt rekursiv definiert wird:

- $n^{(0)} := 0$.
  $s_j^{(0)} := x_j$ für $j \in \{1, \ldots, \text{size}(x)\}$ und $s_j^{(0)} := \sqcup$ für $j \in \mathbb{Z} \setminus \{1, \ldots, \text{size}(x)\}$.
  $\pi^{(0)} := 1$

Ist $(n^{(i)}, s^{(i)}, \pi^{(i)})$ bereits definiert, dann gibt es zwei Fälle:

- Ist $n^{(i)} \neq -1$, so sei $(m, \sigma, \delta) := \Phi(n^{(i)}, s^{(i)}_{\pi^{(i)}})$ und setze
  - $n^{(i+1)} := m$,
  - $s^{(i+1)}_{\pi^{(i)}} := \sigma$, $s_j^{(i+1)} := s_j^{(i)}$ für $j \in \mathbb{Z} \setminus \{\pi^{(i)}\}$,
  - $\pi^{(i+1)} := \pi^{(i)} + \delta$
- Ist $n^{(i)} = -1$, so endet hier die Folge. Dann definiert man $\text{time}(\Phi, x) := i$ und $\text{output}(\Phi, x) \in A^k$ mit $k := \min\{j \in \mathbb{N} \mid s_j^{(i)} = \sqcup\} - 1$ durch $\text{output}(\Phi, x)_j := s_j^{(i)}$ für $j = 1, \ldots, k$.

Ist die Folge unendlich, dann setzen wir $\text{time}(\Phi, x) := \infty$. Dann ist $\text{output}(\Phi, x)$ undefiniert.

Definition: Sei A ein Alphabet und $\bar{A} = A \cup \{\sqcup\}$. Eine **2-Band-Turingmaschine** ist eine Funktion

$$\Phi : \{0, \ldots, N\} \times \bar{A} \times \bar{A} \quad \to \quad \{-1, \ldots, N\} \times \bar{A} \times \bar{A} \times \{-1, 0, 1\} \times \{-1, 0, 1\}$$

für ein $N \in \mathbb{Z}_+$.

Die **Berechnung** von Φ für die Eingabe $x \in A^*$ ist die endliche oder unendliche Folge von Quintupeln $(n^{(i)}, s^{(i)}, t^{(i)}, \pi^{(i)}, \rho^{(i)}) \in \{-1, \ldots, N\} \times \bar{A}^{\mathbb{Z}} \times \bar{A}^{\mathbb{Z}} \times \mathbb{Z} \times \mathbb{Z}$, die wie folgt rekursiv definiert wird:

- $n^{(0)} := 0$.
  $s_j^{(0)} := x_j$ für $j \in \{1, \ldots, \text{size}(x)\}$ und $s_j^{(0)} := \sqcup$ für $j \in \mathbb{Z} \setminus \{1, \ldots, \text{size}(x)\}$.
  $t_j^{(0)} := \sqcup$ für alle $j \in \mathbb{Z}$.
  $\pi^{(0)} := 1$.
  $\rho^{(0)} := 1$.

Ist $(n^{(i)}, s^{(i)}, t^{(i)}, \pi^{(i)}, \rho^{(i)})$ bereits definiert, dann gibt es zwei Fälle:

- Ist $n^{(i)} \neq -1$, so sei $(m, \sigma, \tau, \delta, \epsilon) := \Phi(n^{(i)}, s^{(i)}_{\pi^{(i)}}, t^{(i)}_{\rho^{(i)}})$ und setze
  - $n^{(i+1)} := m$,
  - $s^{(i+1)}_{\pi^{(i)}} := \sigma$, $s_j^{(i+1)} := s_j^{(i)}$ für $j \in \mathbb{Z} \setminus \{\pi^{(i)}\}$,
  - $t^{(i+1)}_{\rho^{(i)}} := \tau$, $t_j^{(i+1)} := t_j^{(i)}$ für $j \in \mathbb{Z} \setminus \{\rho^{(i)}\}$,
  - $\pi^{(i+1)} := \pi^{(i)} + \delta$ und
  - $\rho^{(i+1)} := \rho^{(i)} + \epsilon$
- Ist $n^{(i)} = -1$, so endet hier die Folge. Es werden $\text{time}(\Phi, x)$ und $\text{output}(\Phi, x)$ wie bei der (1-Band-)Turingmaschine definiert.

Modellierung einer 2-Band-Turingmaschine Φ als 1-Band-Turingmaschine Φ'

Zu Beginn:

Setze die Markierung für die simulierten Lese-/Schreibköpfe auf das Startfeld:

$$\Phi'(0,(\cdot,0,\cdot,0)) \quad = \quad (1,(\cdot,1,\cdot,1),0)$$

Es bezeichne "$\cdot$" stets einen beliebigen Wert, der unverändert bleibt.

Implementierung einer einzelnen Anweisung

Sei $\Phi(m, \sigma, \tau) = (m', \sigma', \tau', \delta, \epsilon)$ eine Anweisung von Φ mit $m = 0$.

Notation

- Sei $\zeta \in \overline{A} \backslash \{\sigma\}$.
- Sei $\xi \in \overline{A} \backslash \{\tau\}$.
- π und ρ bezeichnen die Bandpositionen der simulierten 2-Band-Maschine.
- M sei der erste Zustand von Φ', der zur Anweisung $\Phi(m' + 1, \sigma, \tau)$ gehört.

Invariante: Die Bandposition der 1-Band-Maschine sei nach jeder simulierten Anweisung $\min\{\pi, \rho\}$.

Definiere für jedes Paar $(\sigma, \tau) \in \overline{A}^2$ einen Anweisungsblock wie folgt (wobei 13 der erste Zustand des zweiten Blocks sei):

Implementierung von $\Phi(m, \sigma, \tau) = (m', \sigma', \tau', \delta, \epsilon)$

$$\left.\begin{aligned} \Phi'(1,(\zeta,1,\cdot,\cdot)) &= (13,(\zeta,1,\cdot,\cdot),0) \\ \Phi'(1,(\cdot,\cdot,\xi,1)) &= (13,(\cdot,\cdot,\xi,1),0) \end{aligned}\right\} \text{Gehe zum nächsten Block.}$$

$$\left.\begin{aligned} \Phi'(1,(\sigma,1,\cdot,\cdot)) &= (2,(\sigma,1,\cdot,\cdot),0) \\ \Phi'(1,(\cdot,0,\tau,1)) &= (6,(\cdot,0,\tau,1),0) \end{aligned}\right\} \text{Richtiger Block für } \sigma \text{ bzw. } \tau \text{ erreicht.}$$

$$\left.\begin{aligned} \Phi'(2,(\cdot,\cdot,\cdot,0)) &= (2,(\cdot,\cdot,\cdot,0),1) \\ \Phi'(2,(\cdot,\cdot,\xi,1)) &= (12,(\cdot,\cdot,\xi,1),-1) \end{aligned}\right\} \text{Suche Position } \rho.$$

$$\left.\begin{aligned} \Phi'(2,(\cdot,\cdot,\tau,1)) &= (3,(\cdot,\cdot,\tau',0),\epsilon) \\ \Phi'(3,(\cdot,\cdot,\cdot,0)) &= (4,(\cdot,\cdot,\cdot,1),1) \\ \Phi'(4,(\cdot,0,\cdot,\cdot)) &= (4,(\cdot,0,\cdot,\cdot),-1) \\ \Phi'(4,(\sigma,1,\cdot,\cdot)) &= (5,(\sigma',0,\cdot,\cdot),\delta) \\ \Phi'(5,(\cdot,0,\cdot,\cdot)) &= (10,(\cdot,1,\cdot,\cdot),-1) \end{aligned}\right\} \text{Führe Anweisung } \Phi(m,\sigma,\tau) \text{ aus.}$$

Definiere Anweisungen 6, 7, 8, 9 analog zu 2, 3, 4, 5.

Sicherstellung der Voraussetzungen für $m+1$

$$
\left.
\begin{array}{lcl}
\Phi'(10,(\cdot,\cdot,\cdot,\cdot)) & = & (11,(\cdot,\cdot,\cdot,\cdot),-1) \\
\Phi'(11,(\cdot,0,\cdot,0)) & = & (11,(\cdot,0,\cdot,0),1) \\
\Phi'(11,(\cdot,1,\cdot,\cdot)) & = & (M,(\cdot,1,\cdot,\cdot),0) \\
\Phi'(11,(\cdot,0,\cdot,1)) & = & (M,(\cdot,0,\cdot,1),0)
\end{array}
\right\}
$$
Setze $\psi := \min\{\pi,\rho\}$. Gehe zur nächsten Anweisung für Φ.

$$
\left.
\begin{array}{lcl}
\Phi'(12,(\cdot,0,\cdot,0)) & = & (12,(\cdot,0,\cdot,0),-1) \\
\Phi'(12,(\cdot,1,\cdot,\cdot)) & = & (13,(\cdot,1,\cdot,\cdot),0) \\
\Phi'(12,(\cdot,\cdot,\cdot,1)) & = & (13,(\cdot,\cdot,\cdot,1),0)
\end{array}
\right\}
$$
Setze $\psi := \min\{\pi,\rho\}$. Gehe zum nächsten Block.

Definition:

Sei X eine endliche Menge Boolescher Variablen. Eine **Wahrheitsbelegung** für X ist eine Funktion $T : X \to \{\text{true}, \text{false}\}$. Wir erweitern T auf die Menge $L := X \dot{\cup} \{\bar{x} : x \in X\}$ der **Literale** über X, indem wir $T(\bar{x}) := \text{true}$ genau dann setzen, wenn $T(x) = \text{false}$. Eine **Klausel** über X ist eine Menge von Literalen über X. Sie wird von T genau dann **erfüllt**, wenn mindestens eines ihrer Literale durch T auf true gesetzt wird. Eine Familie $\mathcal{Z}$ von Klauseln über X ist genau dann **erfüllbar**, wenn es eine Wahrheitsbelegung gibt, die alle Klauseln in $\mathcal{Z}$ gleichzeitig erfüllt.

Notation: Statt z.B. $\{\{x_1, \bar{x_2}\}, \quad \{\bar{x_1}, x_3\}, \quad \{x_1, x_2, x_3\}\}$ für eine Familie von Klauseln schreibt man $(x_1 \vee \bar{x_2}) \wedge (\bar{x_1} \vee x_3) \wedge (x_1 \vee x_2 \vee x_3)$.

SATISFIBILITY (SAT)

Instanz: Eine Menge X von Variablen und eine Familie $\mathcal{Z}$ von Klauseln über X.
Frage: Ist $\mathcal{Z}$ erfüllbar?