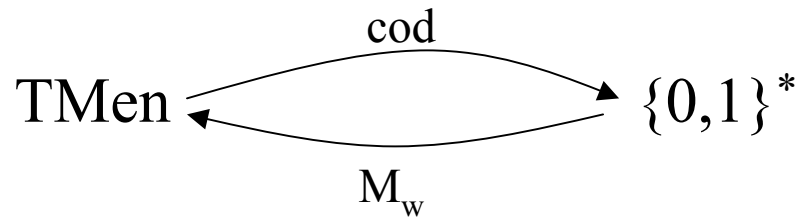
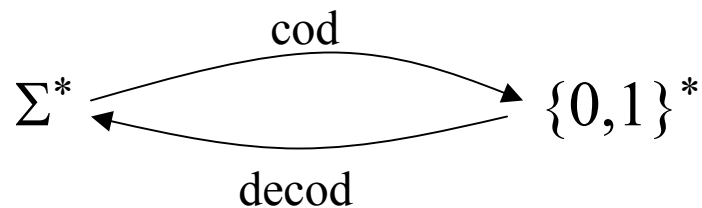


Kodieren/Dekodieren von Turing Maschinen und Strings



$$M_{\text{cod}(M)} \approx M$$



$$\text{decode}(\text{cod}(x)) = x$$

Selbst-Akzeptierende Maschinen

$$\text{SAM} = \{ w \in \{0,1\}^* \mid w \in L(M_w) \}$$

$$\overline{\text{SAM}} = \{ w \in \{0,1\}^* \mid w \notin L(M_w) \}$$

Satz: $\overline{SAM} = \{ w \in \{0,1\}^* \mid w \notin L(M_w) \}$

ist **keine** TM-Sprache (keine semi-entscheidbare Sprache,
keine rekursiv aufzählbare Sprache).

Satz: $\overline{SAM} = \{ w \in \{0,1\}^* \mid w \notin L(M_w) \}$

ist **keine** TM-Sprache (keine semi-entscheidbare Sprache,
keine rekursiv aufzählbare Sprache).

Beweis: Müssen zeigen:

Es gibt **keine** TM M mit $L(M) = \overline{SAM}$.

Satz: $\overline{SAM} = \{ w \in \{0,1\}^* \mid w \notin L(M_w) \}$

ist **keine** TM-Sprache (keine semi-entscheidbare Sprache,
keine rekursiv aufzählbare Sprache).

Beweis: Müssen zeigen:

Es gibt **keine** TM M mit $L(M) = \overline{SAM}$.

Für **jede** TM M gilt $L(M) \neq \overline{SAM}$.

Satz: $\overline{SAM} = \{ w \in \{0,1\}^* \mid w \notin L(M_w) \}$
ist **keine** TM-Sprache (keine semi-entscheidbare Sprache,
keine rekursiv aufzählbare Sprache).

Beweis: Müssen zeigen:

Es gibt **keine** TM M mit $L(M) = \overline{SAM}$.

Für **jede** TM M gilt $L(M) \neq \overline{SAM}$.

Für **jedes** $w \in \{0,1\}^*$ gilt $L(M_w) \neq \overline{SAM}$.

Satz: $\overline{SAM} = \{ w \in \{0,1\}^* \mid w \notin L(M_w) \}$
ist **keine** TM-Sprache (keine semi-entscheidbare Sprache,
keine rekursiv aufzählbare Sprache).

Beweis: Müssen zeigen:

Es gibt **keine** TM M mit $L(M) = \overline{SAM}$.

Für **jede** TM M gilt $L(M) \neq \overline{SAM}$.

Für **jedes** $w \in \{0,1\}^*$ gilt $L(M_w) \neq \overline{SAM}$.

$L(M_w)$ und $\overline{SAM}$ unterscheiden sich in w !!

$$w \notin L(M_w) \Leftrightarrow w \in \overline{SAM}$$

Satz: **SAM** = $\{ w \in \{0,1\}^* \mid w \in L(M_w) \}$
ist **nicht** entscheidbar.

Satz: **SAM** = $\{ w \in \{0,1\}^* \mid w \in L(M_w) \}$
ist **nicht** entscheidbar.

Beweis:

Wäre **SAM** entscheidbar, dann müsste

SAM TM akzeptierbar und auch

SAM TM akzeptierbar sein.

Widerspruch.

Analog: „spezielles Halteproblem“

Ob eine SML Funktion mit
seinem eigenen Programmtext als Argument
terminiert oder nicht,
ist im Allgemeinen **nicht entscheidbar**.

Analog: „spezielles Halteproblem“

Ob eine SML Funktion mit
seinem eigenen Programmtext als Argument
terminiert oder nicht,
ist im Allgemeinen **nicht entscheidbar**.

Das heißt, es gibt kein automatisches Werkzeug,
das aus dem Programmtext einer SML Funktion schließen
kann, ob diese Funktion ihren eigenen Programmtext
akzeptiert oder nicht.

Das „universelle“ (allgemeine) Halteproblem

$$\text{UNIV} = \{ w\$x \in \{0,1\}^* \$ \{0,1\}^* \mid \text{decode}(x) \in L(M_w) \}$$

Satz: Die Sprache

$$\text{UNIV} = \{ w\$x \in \{0,1\}^*\$ \{0,1\}^* \mid \text{decod}(x) \in L(M_w) \}$$

ist **nicht** entscheidbar.

Satz: Die Sprache

$$\text{UNIV} = \{ w\$x \in \{0,1\}^* \$ \{0,1\}^* \mid \text{decod}(x) \in L(M_w) \}$$

ist **nicht** entscheidbar.

Beweis:

Annahme, es gäbe einen Entscheider E für UNIV .

Dies würde einen Entscheider für SAM liefern, was unmöglich ist. Also Widerspruch.

Satz: Die Sprache

$$\text{UNIV} = \{ w\$x \in \{0,1\}^* \$ \{0,1\}^* \mid \text{decod}(x) \in L(M_w) \}$$

ist **nicht** entscheidbar.

Beweis:

Annahme, es gäbe einen Entscheider E für UNIV .

Dies würde einen Entscheider für SAM liefern, was unmöglich ist. Also Widerspruch.

$$w \in \text{SAM} \quad \text{g.d.w.} \quad w\$ \text{cod}(w) \in \text{UNIV}$$

Satz: Die Sprache

$$\text{UNIV} = \{ w\$x \in \{0,1\}^* \$ \{0,1\}^* \mid \text{decod}(x) \in L(M_w) \}$$

ist **nicht** entscheidbar.

Beweis:

Annahme, es gäbe einen Entscheider E für UNIV .

Dies würde einen Entscheider für SAM liefern, was unmöglich ist. Also Widerspruch.

$$w \in \text{SAM} \quad \text{g.d.w.} \quad w\$ \text{cod}(w) \in \text{UNIV}$$

$$w\$ \text{cod}(w) \in \text{UNIV} \Leftrightarrow \text{decod}(\text{cod}(w)) \in L(M_w) \Leftrightarrow w \in L(M_w) \Leftrightarrow w \in \text{SAM}$$

Analog: „allgemeines Halteproblem“

Ob eine SML Funktion mit einem vorgegebenen Argument terminiert oder nicht,
ist im Allgemeinen **nicht entscheidbar**.

Analog: „allgemeines Halteproblem“

Ob eine SML Funktion mit einem vorgegebenen Argument terminiert oder nicht,
ist im Allgemeinen **nicht entscheidbar**.

Das heißt, es gibt kein automatisches Werkzeug,
das aus dem Programmtext einer SML Funktion schließen
kann, ob diese Funktion ein vorgegebenes Argument
akzeptiert oder nicht.

Reduktionen

Def.: Man sagt

“Sprache $A \subset \Sigma^*$ reduziert sich auf Sprache $B \subset \Gamma^*$ ”

“ A ist leichter als B ”

und schreibt

$A \preceq B$,

wenn es eine überall berechenbare Funktion $F: \Sigma^* \rightarrow \Gamma^*$ gibt,
sodass für alle $x \in \Sigma^*$ gilt

$$x \in A \Leftrightarrow F(x) \in B$$

Satz: Wenn $A \preceq B$ dann gilt:

(i) B entscheidbar $\Rightarrow A$ entscheidbar

(ii) A nicht entscheidbar $\Rightarrow B$ nicht entscheidbar

Satz: Wenn $A \preceq B$ dann gilt:

(i) B entscheidbar $\Rightarrow A$ entscheidbar

(ii) A nicht entscheidbar $\Rightarrow B$ nicht entscheidbar

Beweis: (zeige nur (i), da (i) und (ii) äquivalent)

B entscheidbar, d.h. es gibt Entscheider E für B

$A \preceq B$, d.h. es gibt Funktion F , berechnet durch TM M_F
mit $x \in A \Leftrightarrow F(x) \in B$

Entscheide, ob $x \in A$, indem E auf $F(x)$ angewandt wird.