

# **Einführung in die Theoretische Informatik**

**Tutorium – Woche 12**

**Esther Ney**

Sommersemester 2025

## Zulip

Für die diesjährige THEO-Vorlesung wurden **inoffizielle** Streams auf der ZULIP-Instanz der TUM INFORMATIK erstellt. Dieser wird von den meisten Tutor\*innen, aber nicht von PROF. ALBERS gelesen; wir versuchen natürlich trotzdem, organisatorische Fragen bei Bedarf weiterzuleiten. Man kann mich dort auch in Direktnachrichten anschreiben.

- Tutorin: ESTHER NEY
- Tutorien: Dienstag 16-18 (Di-16-2) und Donnerstag 16-18 (Do-16-1)
- E-Mail: [esther.ney@tum.de](mailto:esther.ney@tum.de)
- Aufbau einer Tutorstunde: kurze Wiederholung wichtiger Definitionen und Lemmata, dann selbständiges Bearbeiten der Aufgaben
- Folien: auf ZULIP oder <https://reflect.moe/uni/theo25/>

### Hinweis

Diese Folien sind keine offizielle Musterlösung und dienen lediglich als zusätzliches Material für die Nachbereitung der Tutorien.

Tutorienlösungen können fehlerhaft sein.

Im Zweifel gilt immer das, was in den offiziellen Vorlesungsfolien und in der Musterlösung steht.

### Urheberrechtlicher Hinweis

Die Folien sind nur für die Teilnehmer der entsprechenden Tutorien gedacht und dürfen nicht ohne die ausdrückliche Erlaubnis der Urheber vervielfältigt oder weitergegeben werden. Das Urheberrecht zu den Aufgaben liegt bei den Aufgabenerstellern.

### Definitionen 5.32, 5.34, 5.40 – Halteproblem

Wir sagen, dass  $M[w] \downarrow$ , wenn  $M$  für die Eingabe  $w$  hält. Bezeichne nun mit  $M_w$  die zum Wort  $w$  zugehörige Turingmaschine. Dann definieren wir die drei *Halteprobleme*:

$$K := \{w \in \{0, 1\}^* \mid M_w[w] \downarrow\} \quad (\text{spezielles Halteproblem})$$

$$H := \{w \# x \mid M_w[x] \downarrow\} \quad (\text{allgemeines Halteproblem})$$

$$H_0 := \{w \in \{0, 1\}^* \mid M_w[\varepsilon] \downarrow\} \quad (\text{Halteproblem auf leerem Band})$$

### Satz 5.49 – Satz von Rice

Sei  $\mathcal{F}$  eine Menge berechenbarer Funktionen, sodass  $\mathcal{F}$  nicht trivial ist, d.h. weder leer noch die Menge aller berechenbarer Funktionen ist.

Dann ist  $\{w \mid \varphi_{M_w} \in \mathcal{F}\}$  unentscheidbar.

Intuitiv: es ist unentscheidbar, ob die von  $M$  berechnete Funktion in  $\mathcal{F}$  liegt.

## Semi-Entscheidbarkeit

### Definition 5.42 – Semi-Entscheidbarkeit

Wir sagen, dass eine Menge  $A \subseteq \Sigma^*$  *semi-entscheidbar* ist, falls ihre semi-charakteristische Funktion  $\chi'_A(a) := \begin{cases} 1 & \text{wenn } a \in A \\ \perp & \text{wenn } a \notin A \end{cases}$  berechenbar ist.

Hierbei steht  $\perp$  dafür, dass eine entsprechende Turingmaschine nicht hält.

Sind  $A$  und  $\bar{A}$  semi-entscheidbar, so ist  $A$  entscheidbar.

### Definition 5.45 – rekursiv aufzählbar

Wir sagen, dass eine Menge  $A \subseteq \Sigma^*$  *rekursiv aufzählbar* ist, falls  $A = \emptyset$  gilt oder eine berechenbare totale Funktion  $f : \mathbb{N} \rightarrow A$  existiert, dass

$$A = \{f(0), f(1), \dots\} \quad (\text{rekursiv aufzählbar})$$

Semi-Entscheidbarkeit und rekursive Aufzählbarkeit sind äquivalent.

## It's complicated (get it, complexity)

### Definitionen 6.1 und 6.4 – time und ntime, TIME und NTIME

Wir sagen, dass  $\text{time}_M(w) \in \mathbb{N} \cup \infty$  die Anzahl an Schritten ist, damit die deterministische Turingmaschine (DTM)  $M$  für Eingabe  $w$  hält. Weiter sagen wir, dass die Menge  $A$  in  $\text{TIME}(f(n))$  liegt, falls es eine DTM  $M$  mit  $A = L(M)$  gibt, sodass  $\text{time}_M(w) \leq f(|w|)$  für alle  $w \in \Sigma^*$  gilt.

Wir sagen, dass  $\text{ntime}_M(w) \in \mathbb{N} \cup \infty$  die minimale Anzahl an Schritten ist, damit die nichtdeterministische Turingmaschine (NTM)  $M$  die Eingabe  $w$  akzeptiert. Weiter sagen wir, dass die Menge  $A$  in  $\text{NTIME}(f(n))$  liegt, falls es eine NTM  $M$  mit  $A = L(M)$  gibt, sodass  $\text{ntime}_M(w) \leq f(|w|)$  für alle  $w \in \Sigma^*$  gilt.

### Definitionen 6.2 und 6.4 – P und NP

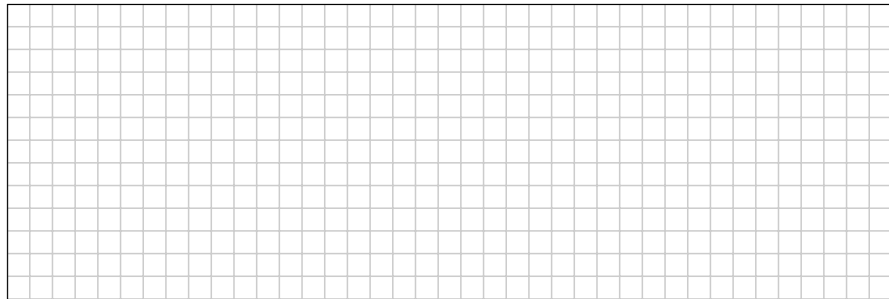
Wir sagen, dass  $A$  in P liegt, falls ein Polynom  $p(n)$  existiert, sodass  $A$  in  $\text{TIME}(p(n))$  liegt, oder äquivalent: falls ein  $k \in \mathbb{N}$  existiert, dass  $A \in \text{TIME}(O(n^k))$ .

Wir sagen, dass  $A$  in NP liegt, falls ein Polynom  $p(n)$  existiert, sodass  $A$  in  $\text{NTIME}(p(n))$  liegt, oder äquivalent: falls ein  $k \in \mathbb{N}$  existiert, dass  $A \in \text{NTIME}(O(n^k))$ .

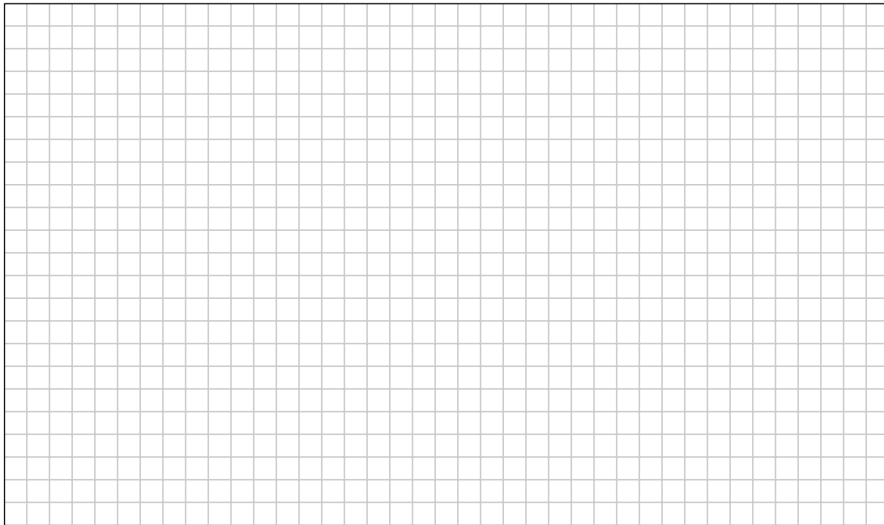
## Aufgabe T 12.1

Betrachten Sie  $H := \{w\#x \mid M_w[x] \downarrow\}$  (allgemeines Halteproblem) und  $H_0 := \{w \in \{0, 1\}^* \mid M_w[\varepsilon] \downarrow\}$  (Halteproblem auf leerem Band).

- (a) Zeigen Sie, dass  $H_{uvu} := \{w\#u\#v \mid M_w[uvu] \downarrow\}$  unentscheidbar ist, indem Sie  $H \leq H_{uvu}$  reduzieren.
- (b) Zeigen Sie, dass  $H_{\Sigma^*} := \{w \in \{0, 1\}^* \mid \forall x \in \Sigma^* : M_w[x] \downarrow\}$  unentscheidbar ist, indem Sie  $H_0 \leq H_{\Sigma^*}$  reduzieren.



## Aufgabe T 12.1





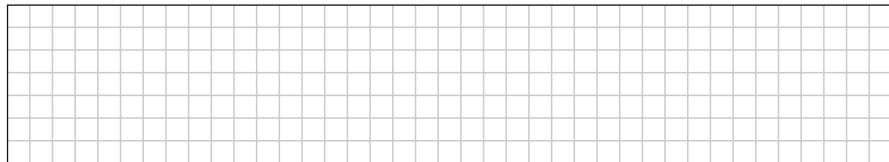
## Aufgabe T 12.2

Sei  $\Sigma$  ein beliebiges Alphabet. Entscheiden Sie, ob die folgenden Mengen entscheidbar sind. Nutzen Sie hierfür den Satz von Rice.

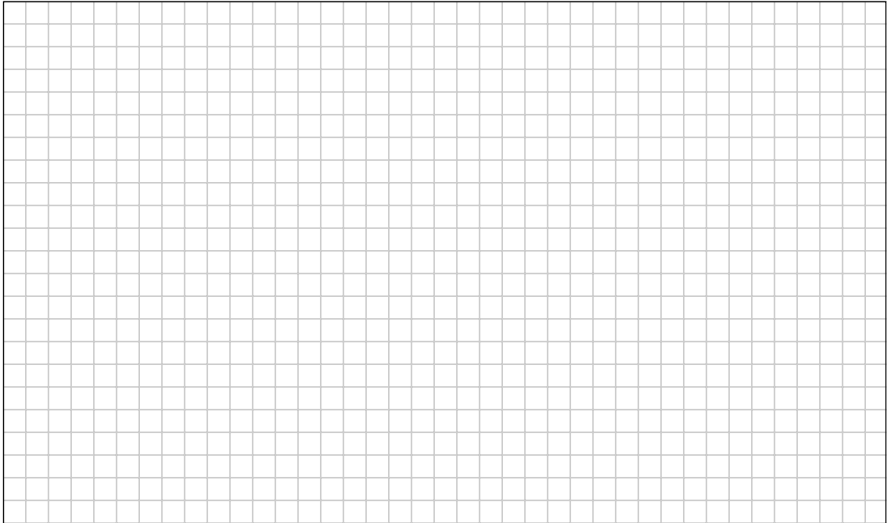
Falls der Satz von Rice anwendbar ist, geben Sie die Menge  $\mathcal{F}$  genau an und nennen Sie ein  $f \in \mathcal{F}$  und ein  $g \notin \mathcal{F}$ .

Falls der Satz von Rice nicht anwendbar ist, erläutern Sie, weshalb, und erklären Sie anderweitig, ob die Menge entscheidbar ist.

- (a)  $L_1 := \{w \in \{0, 1\}^* \mid \{u \in \Sigma^* \mid \varphi_w(u) = 1\} \text{ ist regulär}\}$
- (b)  $L_2 := \{w \in \{0, 1\}^* \mid \forall n \in \mathbb{N}_0 : \varphi_w(n) = n \cdot (n + 23) + 42\}$
- (c)  $L_3 := \{w \in \{0, 1\}^* \mid \forall x \in \Sigma^* : \varphi_w(x) \neq |w|\}$
- (d)  $L_4 := \{w \in \{0, 1\}^* \mid \forall p \in \mathbb{N}_0 : (|w| > p \wedge p \text{ prim}) \implies w_p = 0\}$



## Aufgabe T 12.2



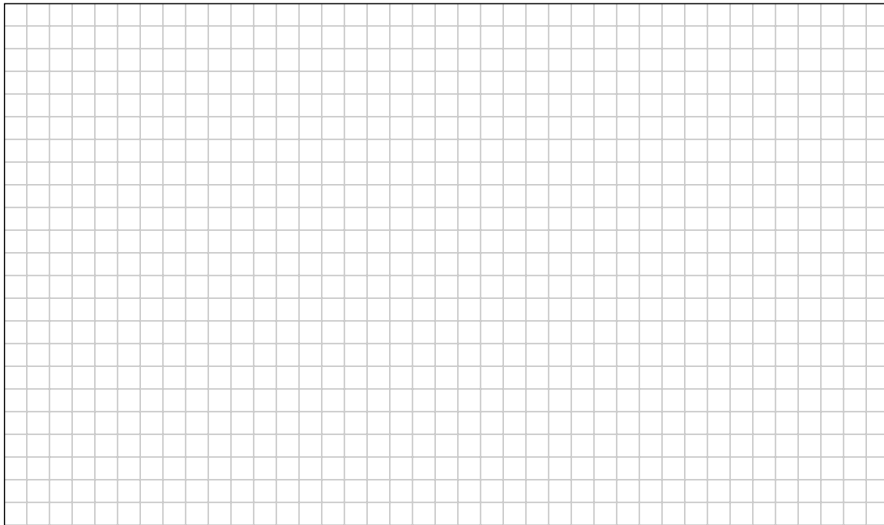


## Aufgabe T 12.4

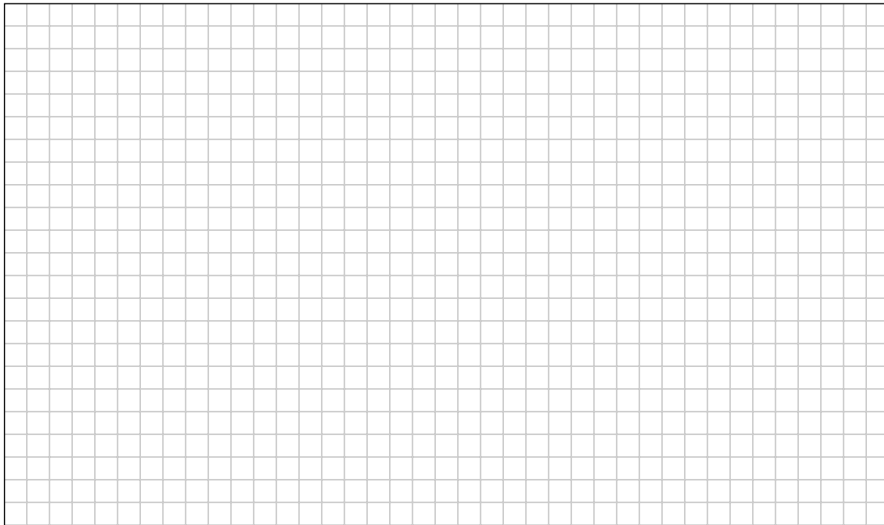
Entscheiden Sie jeweils, ob die folgenden Algorithmen korrekterweise das gegebene Problem in deterministisch, polynomieller Zeit entscheiden. Begründen Sie Ihre Antwort.

- (a) Gegeben: Eine aussagenlogische Formel  $F$ . Frage: Ist  $F$  erfüllbar?  
 Algorithmus: Konstruiere die Wahrheitstabelle für  $F$  und werte  $F$  für jede Belegung aus der Tabelle aus.
- (b) Gegeben: Eine natürliche Zahl  $n > 1$ . Frage: Ist  $n$  prim?  
 Algorithmus: Überprüfe für alle Zahlen  $k$  zwischen 2 und  $n-1$ , ob  $k$   $n$  teilt. Wenn keine dieser Zahlen  $n$  teilt, ist  $n$  prim. (Nehmen Sie an, dass ein Teilbarkeitstest in polynomieller Zeit durchgeführt werden kann.)
- (c) Sei  $k \in \mathbb{N}$  fixiert. Gegeben: Eine endliche Menge  $S \subseteq \mathbb{N}$  mit  $n$  Elementen und eine Zahl  $z$ . Frage: Gilt  $\sum_{s \in K} s > z$  für alle  $(n - k)$ -elementigen Teilmengen  $K$  von  $S$ ?  
 Algorithmus: Wiederhole, solange es noch neue  $(n - k)$ -elementige Teilmengen von  $S$  gibt: Berechne die nächste solche Teilmenge  $K$ , summiere ihre Elemente und überprüfe, ob diese Summe größer als  $z$  ist. Wenn nein, antworte nein. Wenn es keine weitere  $(n - k)$ -elementige Teilmengen von  $S$  gibt, antworte ja. (Nehmen Sie an, dass die nächste Teilmenge in polynomieller Zeit berechnet werden kann.)
- (d) Frage (c), aber mit  $\frac{n}{k}$ -elementigen Teilmengen, wobei  $k$  ein Teiler von  $n$  ist.

## Aufgabe T 12.4



## Aufgabe T 12.4



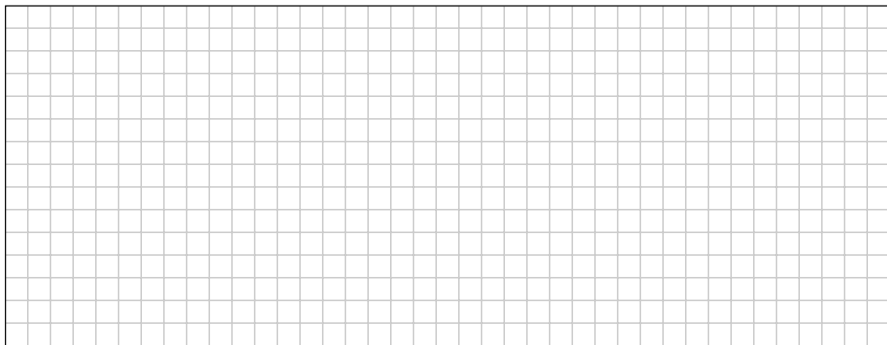
## Aufgabe T 12.5

*Hinweis:* Diese Aufgabe ist optional. Sie ist als kleine Knobelaufgabe gedacht und wird nicht besprochen.

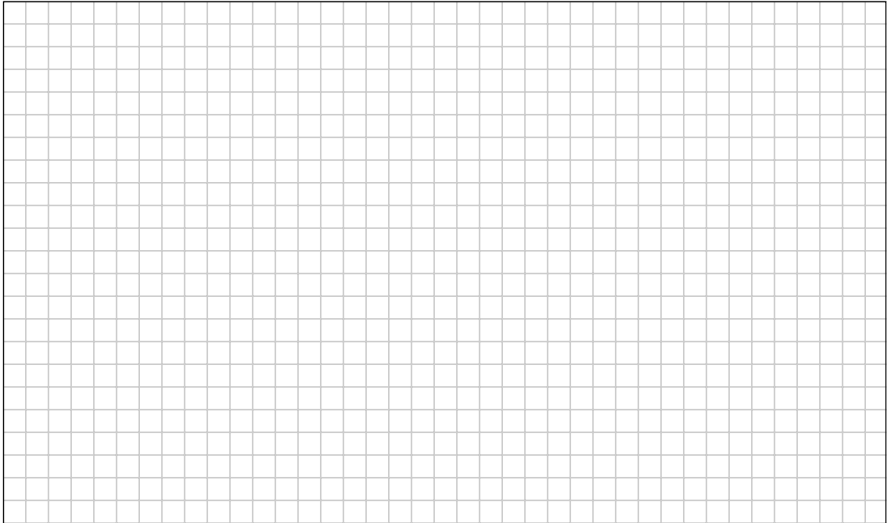
Sei  $\Sigma$  ein beliebiges Alphabet. Zeigen Sie: Die Menge

$$H_{NEQ} := \{w_1 \# w_2 \mid w_1, w_2 \in \{0, 1\}^* \text{ und } \forall x \in \Sigma^*. M_{w_1}[x] \downarrow \iff \neg M_{w_2}[x] \downarrow\}$$

ist unentscheidbar.



## Aufgabe T 12.5





**Heute keine Altklausuraufgabe!**

Wegen des Stoffumfangs **entfällt** heute die Altklausuraufgabe.