

Theoretische Grundlagen der Informatik

Übung

4. Übungstermin · 29. November 2016
Guido Brückner

INSTITUT FÜR THEORETISCHE INFORMATIK · LEHRSTUHL ALGORITHMIK

Gliederung

Turingmaschinen und Berechenbarkeit

- Universelle Turingmaschinen
- Entscheidbarkeit und Semi-Entscheidbarkeit
- Satz von Rice
- Post'sches Korrespondenzproblem

Komplexitätsklassen

- Sprachen, Problem und Zeitkomplexität
- Klasse NP
- Über die Klasse P und NP hinaus

Wiederholung



Entscheidbarkeit

Satz: Eine Sprache $L \subseteq \Sigma^*$ heißt **rekursiv** oder **entscheidbar**, wenn es eine Turing-Maschine gibt, die auf allen Eingaben stoppt und eine Eingabe w genau dann akzeptiert, wenn $w \in L$ gilt.

$\Rightarrow \mathcal{M}$ entscheidet L .

Semi-Entscheidbarkeit

Satz: Eine Sprache $L \subseteq \Sigma^*$ heißt **rekursiv-aufzählbar** oder **semi-entscheidbar**, wenn es eine Turing-Maschine gibt, die genau die Eingaben w akzeptiert für die $w \in L$. Das Verhalten der Turing-Maschine für Eingaben $w \notin L$ ist damit nicht definiert. D.h., die Turing-Maschine stoppt entweder nicht in einem Endzustand oder aber stoppt gar nicht.

$\Rightarrow \mathcal{M}$ akzeptiert L .

Beziehung zwischen Entscheidbarkeit und Semi-Entscheidbarkeit

Satz: Eine Sprache L ist genau dann entscheidbar, wenn L und deren Komplement L^c semi-entscheidbar sind.

Universelle Turingmaschine

Bisher:

- Bislang beschriebenen DTMs sind für spezielle Aufgaben

Intuitiver Wunsch:

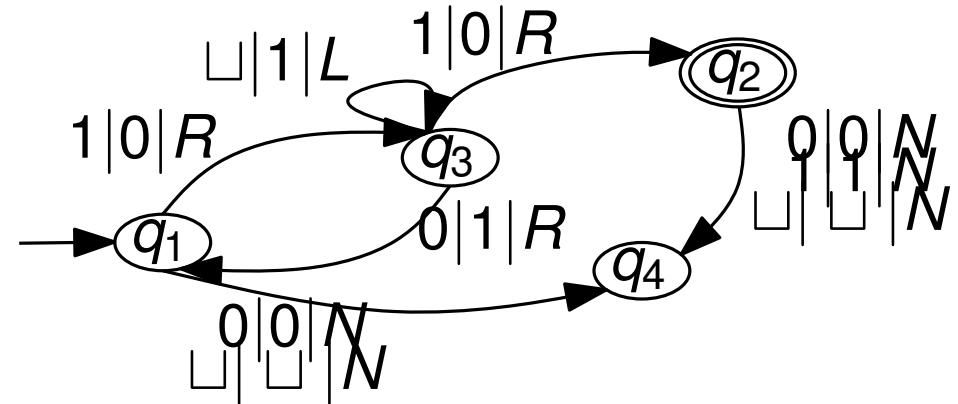
- Eine Art programmierbarer Rechner, der als Eingabe ein Programm und die Eingabe für dieses Programm bekommt

Beschreibung einer TM

- $\mathcal{M} := (Q, \Sigma, \Gamma, \delta, s, F)$
- Gödelnummer $\langle \mathcal{M} \rangle$ von \mathcal{M} , ist definiert durch folgende Kodierungsvorschrift:
 1. Kodiere $\delta(q_i, a_j) = (q_r, a_s, d_t)$ durch $0^i 10^j 10^r 10^s 10^t$, mit $d_t \in \{d_1, d_2, d_3\}$, d_1 für L , d_2 für R und d_3 für N
 2. Turing-Maschine wird kodiert durch:
 $111\text{code}_1 11\text{code}_2 11 \dots 11\text{code}_z 111$,
mit code_i für $i = 1, \dots, z$ entspricht allen Funktionswerten von δ

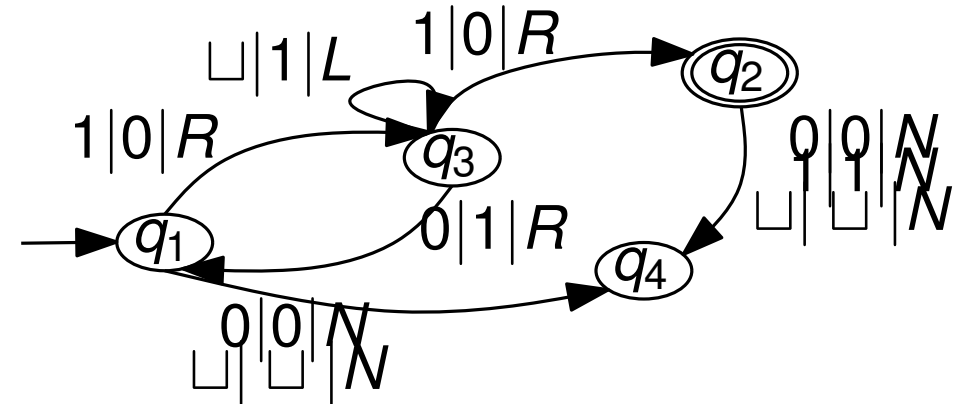
Universelle Turingmaschine

Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ :



Universelle Turingmaschine

Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ :



- Gödelnummer $\langle \mathcal{M} \rangle$ von \mathcal{M} , ist definiert durch folgende Kodierungsvorschrift:

1. Kodiere $\delta(q_i, a_j) = (q_r, a_s, d_t)$ durch $0^i 1 0^j 1 0^r 1 0^s 1 0^t$, mit $d_t \in \{d_1, d_2, d_3\}$, d_1 für L , d_2 für R und d_3 für N

2. Turing-Maschine wird kodiert durch:

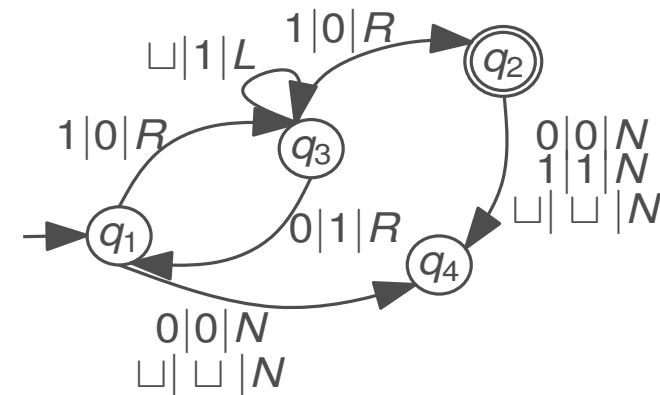
$111\text{code}_1 11\text{code}_2 11 \dots 11\text{code}_z 111$,

mit code_i für $i = 1, \dots, z$ entspricht allen Funktionswerten von δ

Universelle Turingmaschine

Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

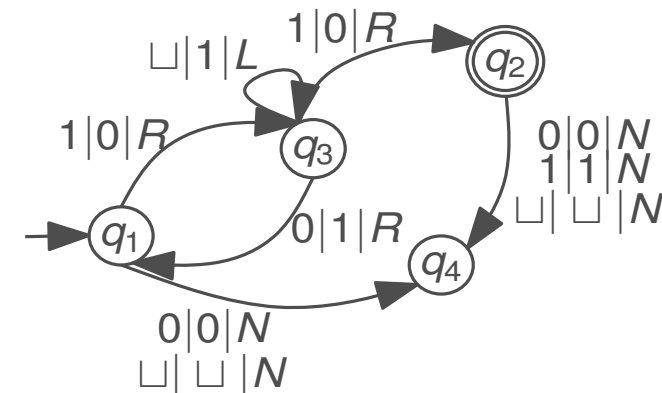
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup$$

$$D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N$$

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



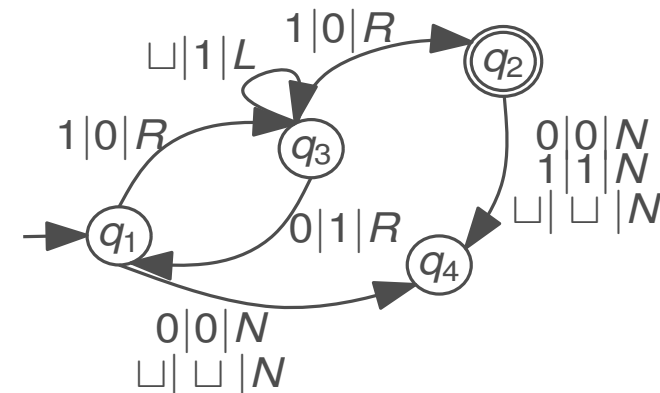
Universelle Turingmaschine

Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m)$$

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



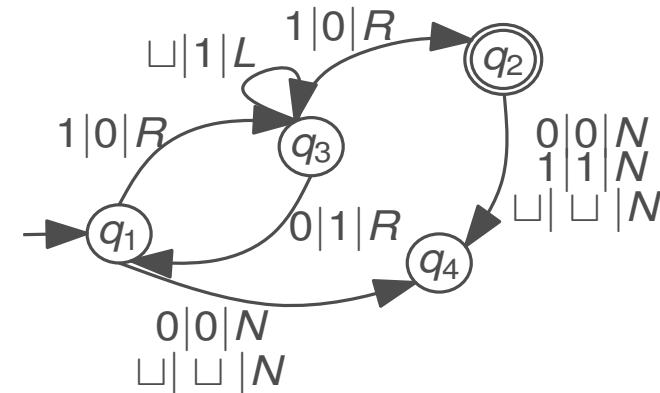
Universelle Turingmaschine

Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 10^j 1 0^k 10^\ell 10^m$$

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

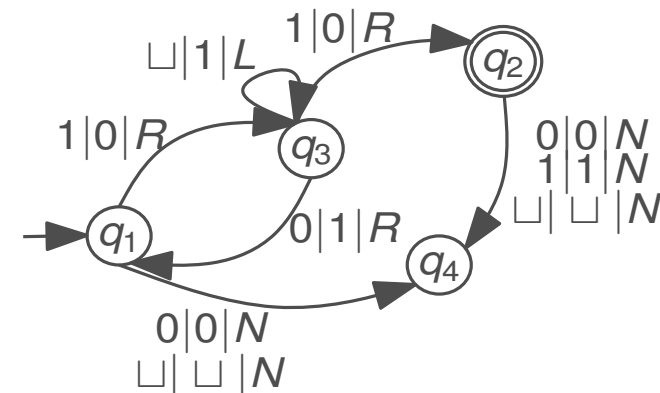
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 10^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

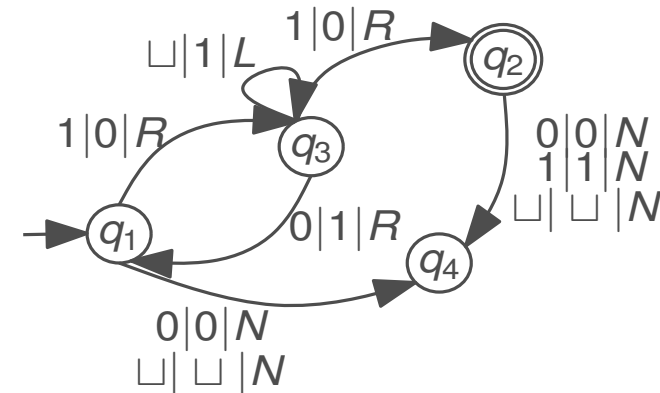
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 10^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

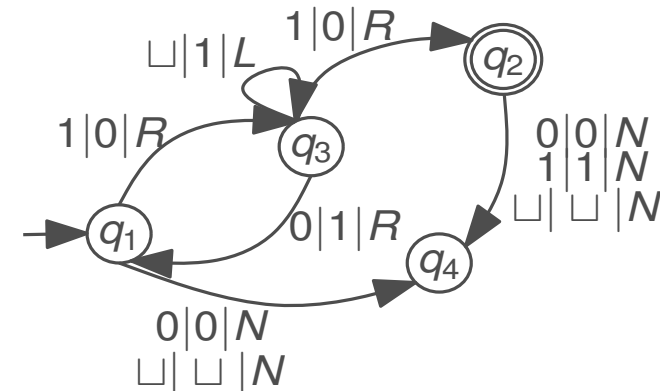
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 10^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

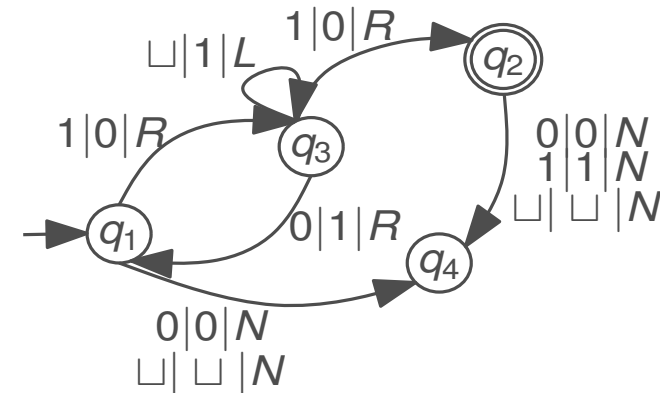
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 10^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

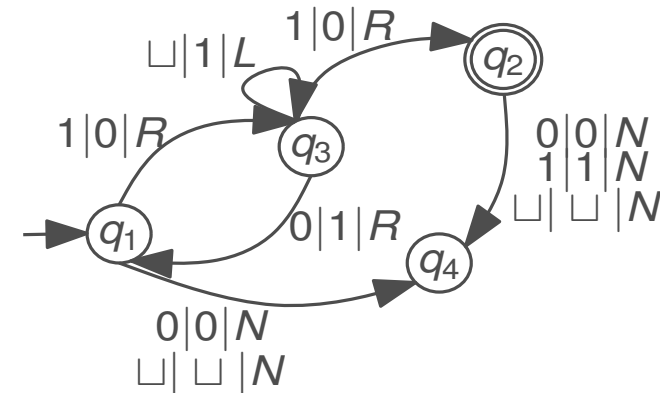
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

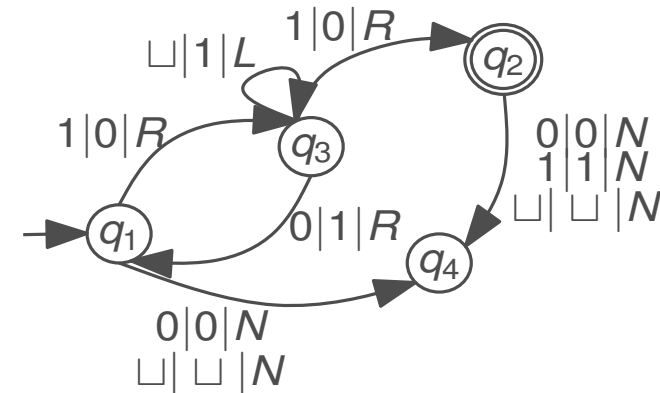
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 1 0^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

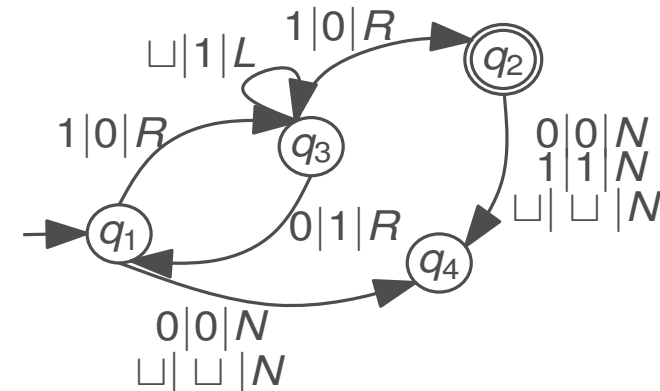
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

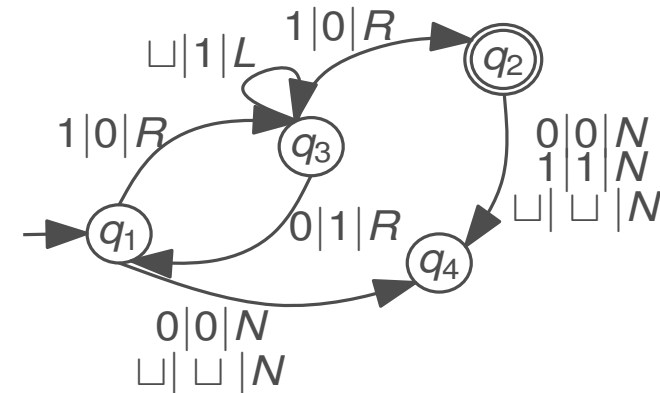
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

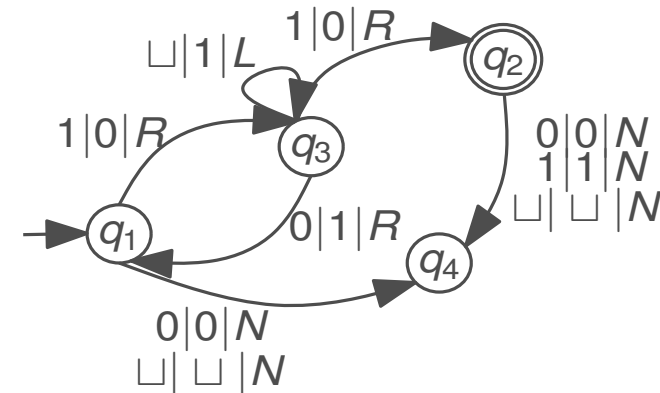
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

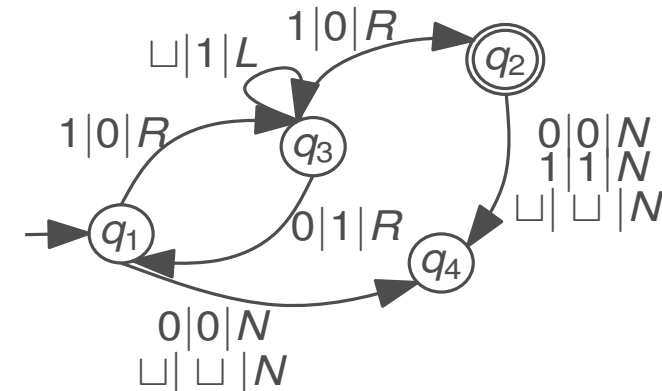
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	010001000010001000
4	$\delta(q_3, 0) = (q_1, 1, R)$	
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

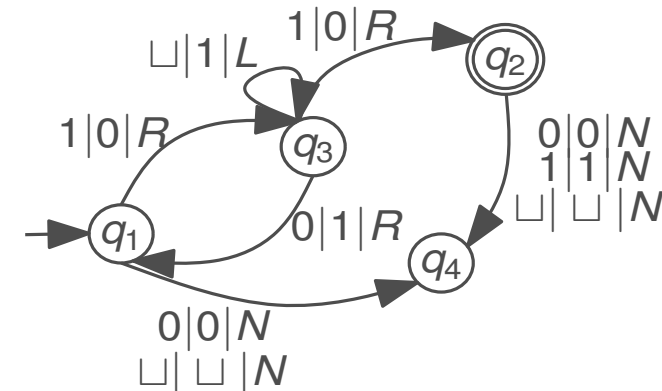
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	0100010000100001000
4	$\delta(q_3, 0) = (q_1, 1, R)$	0001010000101000
5	$\delta(q_3, 1) = (q_2, 0, R)$	
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

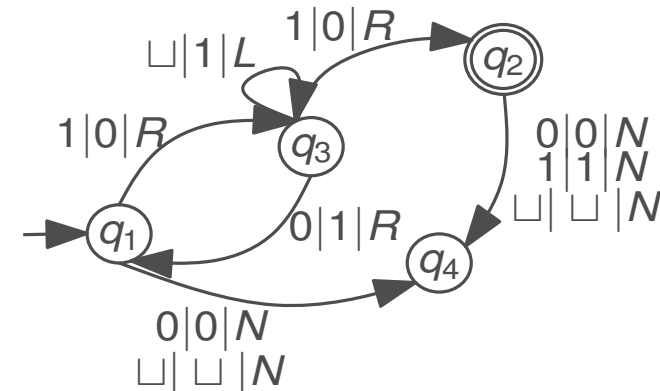
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad \left\{ 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m \right.$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	0100010000100001000
4	$\delta(q_3, 0) = (q_1, 1, R)$	0001010000101000
5	$\delta(q_3, 1) = (q_2, 0, R)$	00010010010100
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

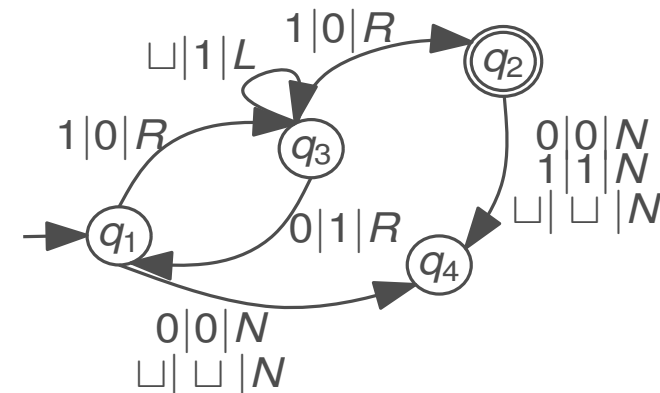
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \} 0^i 1 0^j 1 0^k 1 0^\ell 1 0^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	0100010000100001000
4	$\delta(q_3, 0) = (q_1, 1, R)$	0001010000101000
5	$\delta(q_3, 1) = (q_2, 0, R)$	00010010010100
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	0001000100010010

δ	0	1	\sqcup
q_1	$(q_4, 0, N)$	$(q_3, 0, R)$	(q_4, \sqcup, N)
q_3	$(q_1, 1, R)$	$(q_2, 0, R)$	$(q_3, 1, L)$



Universelle Turingmaschine

Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 10^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	0100010000100001000
4	$\delta(q_3, 0) = (q_1, 1, R)$	0001010000101000
5	$\delta(q_3, 1) = (q_2, 0, R)$	00010010010100
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	0001000100010010

$$\langle \mathcal{M} \rangle = 1110101000010100011101001000101001110100010000100010001000100010100001010001100010010010100110001000100010010111$$

Universelle Turingmaschine

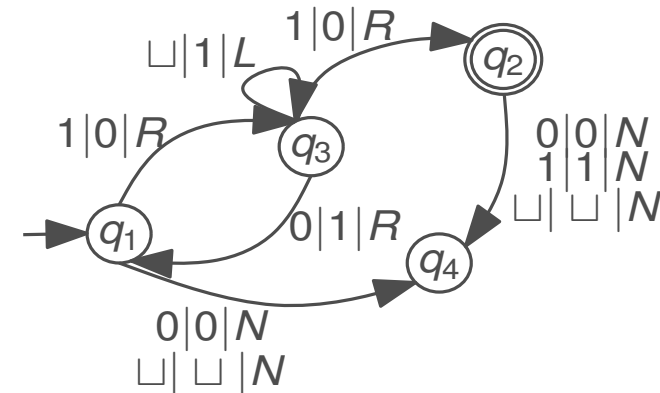
Gegeben ist folgende TM \mathcal{M} mit $Q = \{q_1, q_2, q_3, q_f\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, \sqcup\}$, $s = q_1$, $F = \{q_2\}$ und δ . Was ist die Gödelnummer $\langle \mathcal{M} \rangle$ der TM?

Kodierung:

$$\left. \begin{array}{l} X_1 \hat{=} 0, X_2 \hat{=} 1, X_3 \hat{=} \sqcup \\ D_1 \hat{=} L, D_2 \hat{=} R, D_3 \hat{=} N \end{array} \right\} \delta(q_i, X_j) = (q_k, X_\ell, D_m) \quad 0^i 10^j 1 0^k 10^\ell 10^m$$

#	Eintrag	Kodierung
1	$\delta(q_1, 0) = (q_4, 0, N)$	01010000101000
2	$\delta(q_1, 1) = (q_3, 0, R)$	0100100010100
3	$\delta(q_1, \sqcup) = (q_4, \sqcup, N)$	010001000010001000
4	$\delta(q_3, 0) = (q_1, 1, R)$	0001010000101000
5	$\delta(q_3, 1) = (q_2, 0, R)$	00010010010100
6	$\delta(q_3, \sqcup) = (q_3, 1, L)$	0001000100010010

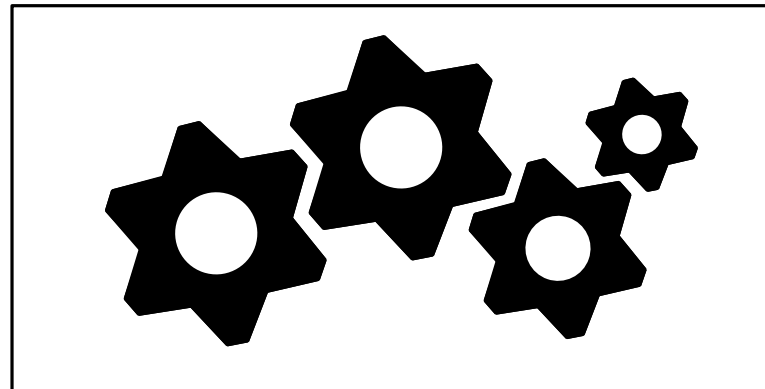
$\langle \mathcal{M} \rangle = 148365654112389252472285479602327$



Universelle Turingmaschine

Definition Eine Turingmaschine \mathcal{M}_0 heißt universell, falls für jede 1-Band-DTM \mathcal{M} und jedes $x \in \{0, 1\}^*$ gilt:

- \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ hält genau dann, wenn \mathcal{M} gestartet mit x hält.
- Falls \mathcal{M} gestartet mit x hält, berechnet \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ die gleiche Ausgabe wie \mathcal{M} gestartet mit x . Insbesondere akzeptiert \mathcal{M}_0 die Eingabe $\langle \mathcal{M} \rangle x$ genau dann, wenn \mathcal{M} die Eingabe x akzeptiert.



Universelle Turingmaschine
simuliert \mathcal{M}

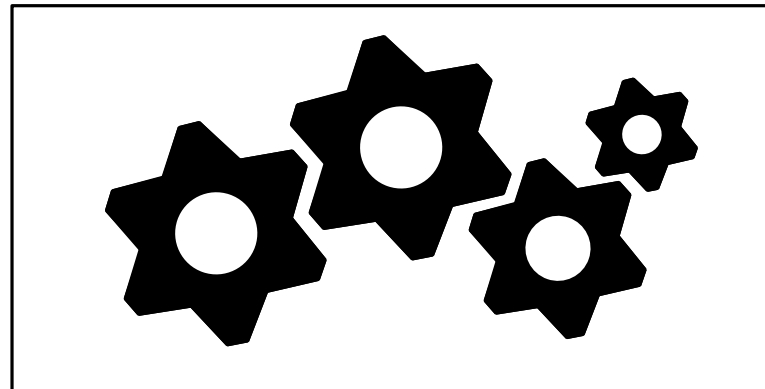
Universelle Turingmaschine

Definition Eine Turingmaschine \mathcal{M}_0 heißt universell, falls für jede 1-Band-DTM \mathcal{M} und jedes $x \in \{0, 1\}^*$ gilt:

- \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ hält genau dann, wenn \mathcal{M} gestartet mit x hält.
- Falls \mathcal{M} gestartet mit x hält, berechnet \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ die gleiche Ausgabe wie \mathcal{M} gestartet mit x . Insbesondere akzeptiert \mathcal{M}_0 die Eingabe $\langle \mathcal{M} \rangle x$ genau dann, wenn \mathcal{M} die Eingabe x akzeptiert.

$$x + y$$

Spezielle Turingmaschine \mathcal{M}
z.B.: Addition zweier Zahlen

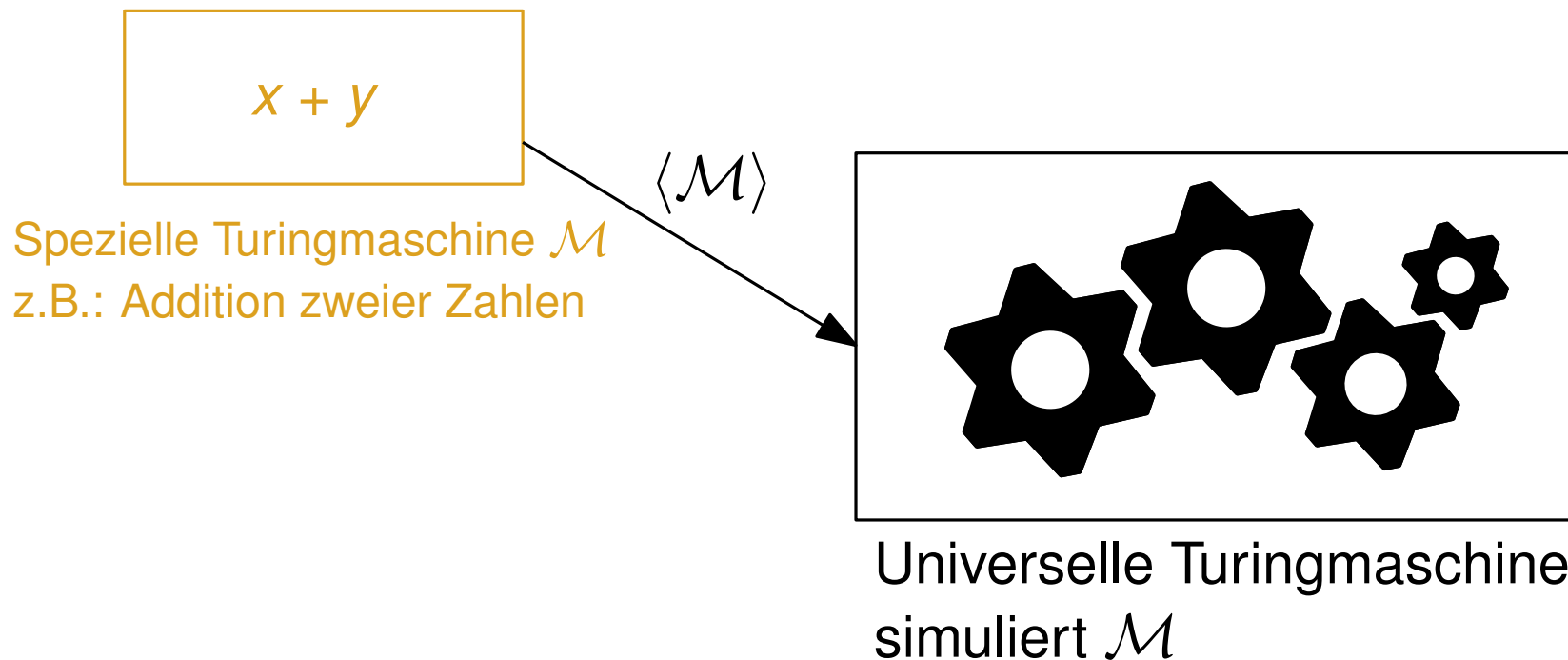


Universelle Turingmaschine
simuliert \mathcal{M}

Universelle Turingmaschine

Definition Eine Turingmaschine \mathcal{M}_0 heißt universell, falls für jede 1-Band-DTM \mathcal{M} und jedes $x \in \{0, 1\}^*$ gilt:

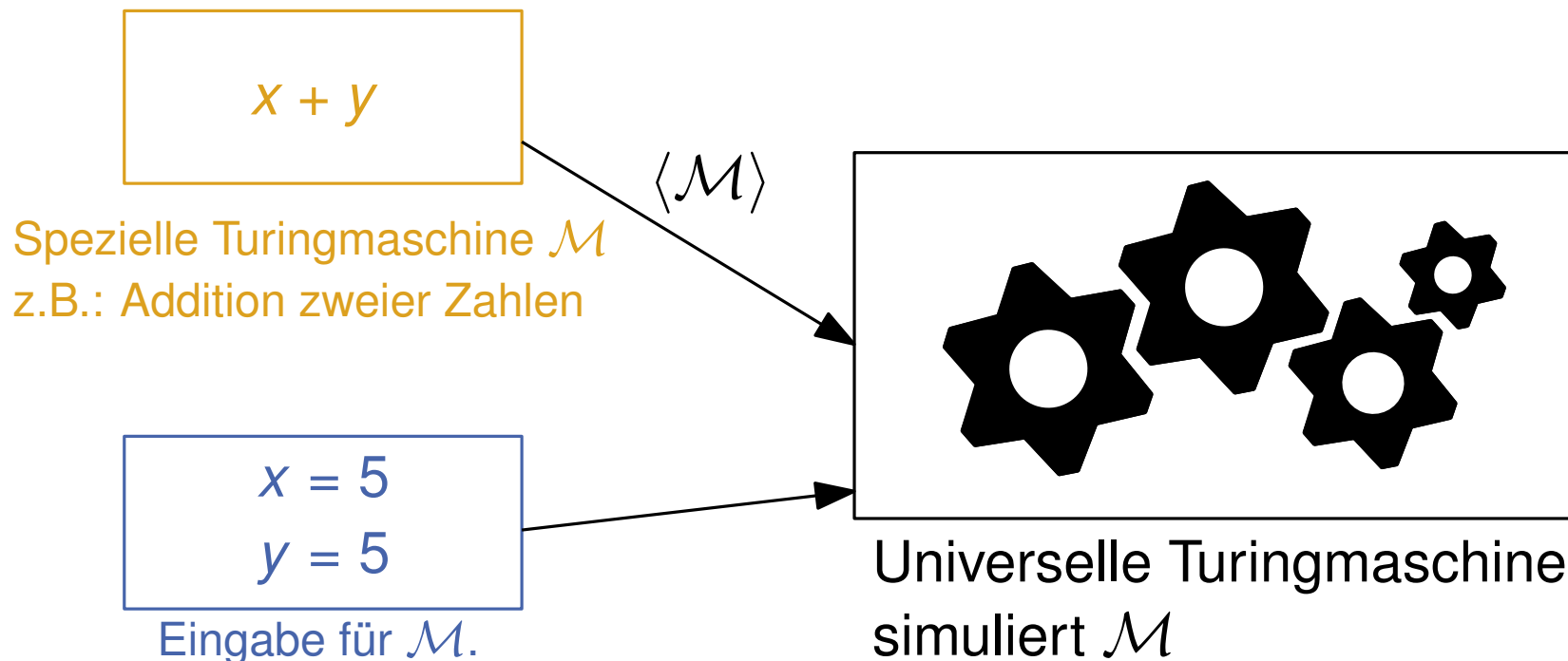
- \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ hält genau dann, wenn \mathcal{M} gestartet mit x hält.
- Falls \mathcal{M} gestartet mit x hält, berechnet \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ die gleiche Ausgabe wie \mathcal{M} gestartet mit x . Insbesondere akzeptiert \mathcal{M}_0 die Eingabe $\langle \mathcal{M} \rangle x$ genau dann, wenn \mathcal{M} die Eingabe x akzeptiert.



Universelle Turingmaschine

Definition Eine Turingmaschine \mathcal{M}_0 heißt universell, falls für jede 1-Band-DTM \mathcal{M} und jedes $x \in \{0, 1\}^*$ gilt:

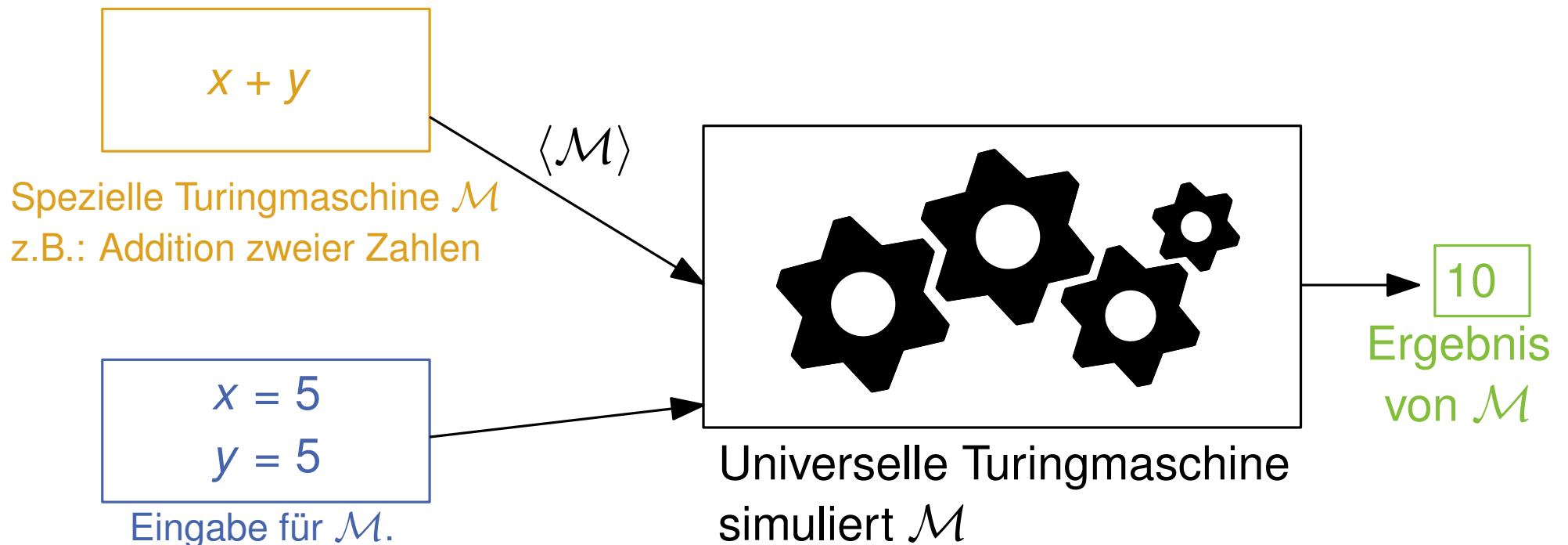
- \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ hält genau dann, wenn \mathcal{M} gestartet mit x hält.
- Falls \mathcal{M} gestartet mit x hält, berechnet \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ die gleiche Ausgabe wie \mathcal{M} gestartet mit x . Insbesondere akzeptiert \mathcal{M}_0 die Eingabe $\langle \mathcal{M} \rangle x$ genau dann, wenn \mathcal{M} die Eingabe x akzeptiert.



Universelle Turingmaschine

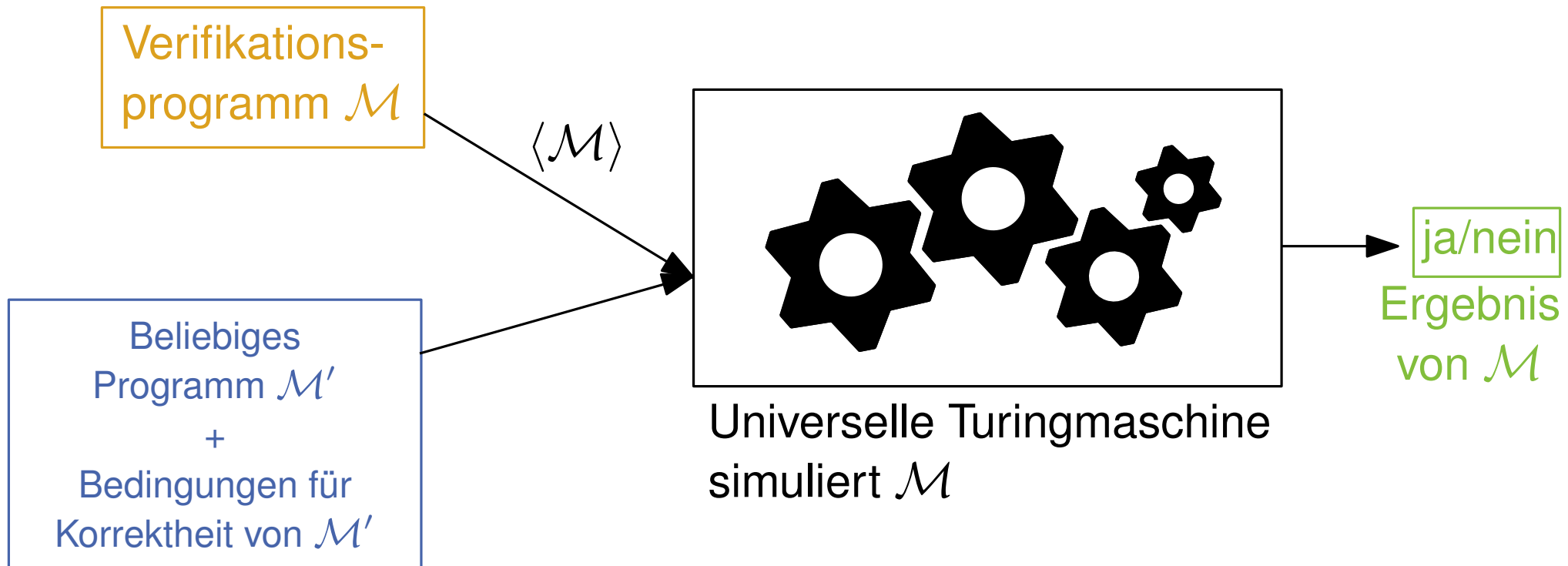
Definition Eine Turingmaschine \mathcal{M}_0 heißt universell, falls für jede 1-Band-DTM \mathcal{M} und jedes $x \in \{0, 1\}^*$ gilt:

- \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ hält genau dann, wenn \mathcal{M} gestartet mit x hält.
- Falls \mathcal{M} gestartet mit x hält, berechnet \mathcal{M}_0 gestartet mit $\langle \mathcal{M} \rangle x$ die gleiche Ausgabe wie \mathcal{M} gestartet mit x . Insbesondere akzeptiert \mathcal{M}_0 die Eingabe $\langle \mathcal{M} \rangle x$ genau dann, wenn \mathcal{M} die Eingabe x akzeptiert.



Universelle TM – Beispiel

Gibt es ein Programm \mathcal{M} , das für jedes beliebige Programm \mathcal{M}' dessen Korrektheit beweist?



Universelle TM – Beispiel

Gibt es ein Programm \mathcal{M} , das für jedes beliebige Programm \mathcal{M}' dessen Korrektheit beweist?

Verifikations-

Satz von Rice

Sei R die Menge der von Turingmaschinen berechenbaren Funktionen und S eine nicht-triviale Teilmenge von R ($\emptyset \neq S \neq R$). Dann ist die Sprache

$$L(S) := \{ \langle \mathcal{M} \rangle \mid \mathcal{M} \text{ berechnet eine Funktion aus } S \}$$

Pr nicht entscheidbar.

Bedingungen für
Korrektheit von \mathcal{M}'

ja/nein
Ergebnis
von \mathcal{M}

Aufgaben zu Entscheidbarkeit



Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

- Es gilt, dass $\mathcal{H} = \{wv \mid T_w \text{ hält für die Eingabe } v\}$ semi-entscheidbar ist

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

- Es gilt, dass $\mathcal{H} = \{wv \mid T_w \text{ hält für die Eingabe } v\}$ semi-entscheidbar ist
 - Konstruiere eine universelle TM \mathcal{M} , die T_w simuliert.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

- Es gilt, dass $\mathcal{H} = \{wv \mid T_w \text{ hält für die Eingabe } v\}$ semi-entscheidbar ist
 - Konstruiere eine universelle TM \mathcal{M} , die T_w simuliert.
 - \mathcal{M} akzeptiert, sobald T_w die Berechnung beendet hat

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

- Es gilt, dass $\mathcal{H} = \{wv \mid T_w \text{ hält für die Eingabe } v\}$ semi-entscheidbar ist
 - Konstruiere eine universelle TM \mathcal{M} , die T_w simuliert.
 - \mathcal{M} akzeptiert, sobald T_w die Berechnung beendet hat
 - T_w läuft endlos \Rightarrow auch \mathcal{M} akzeptiert nie (Semi-Entscheidbarkeit)

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

- Es gilt, dass $\mathcal{H} = \{wv \mid T_w \text{ hält für die Eingabe } v\}$ semi-entscheidbar ist
 - Konstruiere eine universelle TM \mathcal{M} , die T_w simuliert.
 - \mathcal{M} akzeptiert, sobald T_w die Berechnung beendet hat
 - T_w läuft endlos \Rightarrow auch \mathcal{M} akzeptiert nie (Semi-Entscheidbarkeit)

\Rightarrow Da \mathcal{H} nicht entscheidbar, aber semi-entscheidbar und \mathcal{H}^c semi-entscheidbar $\Rightarrow \mathcal{H}^c$ nicht semi-entscheidbar

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:


- a) Zeigen Sie, dass das Komplement \mathcal{H}^c des Halteproblems nicht semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- \mathcal{H} ist nicht entscheidbar.
- Für eine Sprache L . L und L^c sind semi-entscheidbar gdw. L ist entscheidbar.

Zeige: \mathcal{H}^c ist nicht semi-entscheidbar.

- Es gilt, dass $\mathcal{H} = \{wv \mid T_w \text{ hält für die Eingabe } v\}$ semi-entscheidbar ist
 - Konstruiere eine universelle TM \mathcal{M} , die T_w simuliert.
 - \mathcal{M} akzeptiert, sobald T_w die Berechnung beendet hat
 - T_w läuft endlos \Rightarrow auch \mathcal{M} akzeptiert nie (Semi-Entscheidbarkeit)

\Rightarrow Da \mathcal{H} nicht entscheidbar, aber semi-entscheidbar und \mathcal{H}^c semi-entscheidbar $\Rightarrow \mathcal{H}^c$ nicht semi-entscheidbar 

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

- b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$
- Die Sprache L_d ist nicht entscheidbar.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$
- Die Sprache L_d ist nicht entscheidbar.
- Die Sprache $L_d^c := \{0, 1\}^* \setminus L_d$ ist nicht entscheidbar.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$
- Die Sprache L_d ist nicht entscheidbar.
- Die Sprache $L_d^c := \{0, 1\}^* \setminus L_d$ ist nicht entscheidbar.

Idee: Verwende universelle TM T_U mit Eingabe $(\langle M_i \rangle, v)$.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$
- Die Sprache L_d ist nicht entscheidbar.
- Die Sprache $L_d^c := \{0, 1\}^* \setminus L_d$ ist nicht entscheidbar.

Idee: Verwende universelle TM T_U mit Eingabe $(\langle M_i \rangle, v)$.

Fall: M_i akzeptiert die Eingabe nach endlich vielen Schritten. Damit akzeptiert auch die universelle TM T_U die Eingabe.

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$
- Die Sprache L_d ist nicht entscheidbar.
- Die Sprache $L_d^c := \{0, 1\}^* \setminus L_d$ ist nicht entscheidbar.

Idee: Verwende universelle TM T_U mit Eingabe $(\langle M_i \rangle, v)$.

Fall: M_i akzeptiert die Eingabe nach endlich vielen Schritten. Damit akzeptiert auch die universelle TM T_U die Eingabe.

Fall: M_i akzeptiert nicht die Eingabe, so wird die Eingabe ebenfalls nicht von T_U akzeptiert (unabhängig, ob die Simulation stoppt oder nicht).

Aufgabe – Semi-Entscheidbarkeit

Zeigen Sie:

b) Zeigen Sie, dass das Komplement L_d^c der Diagonalsprache semi-entscheidbar ist.

Aussagen aus der Vorlesung:

- Diagonalsprache def. als $L_d = \{w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht}\}$
- Die Sprache L_d ist nicht entscheidbar.
- Die Sprache $L_d^c := \{0, 1\}^* \setminus L_d$ ist nicht entscheidbar.

Idee: Verwende universelle TM T_U mit Eingabe $(\langle M_i \rangle, v)$.

Fall: M_i akzeptiert die Eingabe nach endlich vielen Schritten. Damit akzeptiert auch die universelle TM T_U die Eingabe.

Fall: M_i akzeptiert nicht die Eingabe, so wird die Eingabe ebenfalls nicht von T_U akzeptiert (unabhängig, ob die Simulation stoppt oder nicht).



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Verfahren: Sei x die Eingabe.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Verfahren: Sei x die Eingabe.

- Wähle nicht-deterministisch ein nicht-leeres Präfix π von x .

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Verfahren: Sei x die Eingabe.

- Wähle nicht-deterministisch ein nicht-leeres Präfix π von x .
- Überprüfe mithilfe von \mathcal{M} , ob π in L liegt.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Verfahren: Sei x die Eingabe.

- Wähle nicht-deterministisch ein nicht-leeres Präfix π von x .
- Überprüfe mithilfe von \mathcal{M} , ob π in L liegt.

Fall: π liegt nicht in $L \rightarrow M$ akzeptiert x nicht.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Verfahren: Sei x die Eingabe.

- Wähle nicht-deterministisch ein nicht-leeres Präfix π von x .
- Überprüfe mithilfe von \mathcal{M} , ob π in L liegt.

Fall: π liegt nicht in $L \rightarrow M$ akzeptiert x nicht.

Fall: π liegt in L : M löscht π vom Band. Falls das Band nun leer ist, *akzeptiert* M die Eingabe x . Sonst wiederhole Verfahren.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- a) Die Menge der entscheidbaren Sprachen ist unter dem Kleene'schen Abschluss abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass L^* auch entscheidbar ist.

Idee:

- Sei L eine entscheidbare Sprache.
- Es gibt also eine Turing-Maschine \mathcal{M} , die L entscheidet: $L(\mathcal{M}) = L$.
- Konstruiere eine NTM \mathcal{M}' .

Verfahren: Sei x die Eingabe.

- Wähle nicht-deterministisch ein nicht-leeres Präfix π von x .
- Überprüfe mithilfe von \mathcal{M} , ob π in L liegt.

Fall: π liegt nicht in $L \rightarrow M$ akzeptiert x nicht.

Fall: π liegt in L : M löscht π vom Band. Falls das Band nun leer ist, *akzeptiert* M die Eingabe x . Sonst wiederhole Verfahren. ✓

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation min abgeschlossen, d.h. für jede entscheidbare Sprache L gilt, dass $\text{min}(L)$ auch entscheidbar ist. Die Operation min ist für eine entscheidbare Sprache L definiert als

$$\text{min}(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Hinweis: Ein Präfix von x heißt *echt*, wenn es nicht x ist.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.
3. T' generiert das nächste echte Präfix p von x .

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.
3. T' generiert das nächste echte Präfix p von x .
4. Es gibt kein weiteres Präfix mehr: T akzeptiert.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.
3. T' generiert das nächste echte Präfix p von x .
4. Es gibt kein weiteres Präfix mehr: T akzeptiert.
5. T_L entscheidet p .

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.
3. T' generiert das nächste echte Präfix p von x .
4. Es gibt kein weiteres Präfix mehr: T akzeptiert.
5. T_L entscheidet p .
6. Wenn $p \in L$, dann hält T und akzeptiert nicht.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

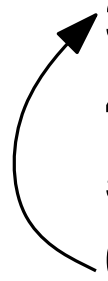
Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.
3. T' generiert das nächste echte Präfix p von x .
4. Es gibt kein weiteres Präfix mehr: T akzeptiert.
5. T_L entscheidet p .
6. Wenn $p \in L$, dann hält T und akzeptiert nicht.

sonst



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

b) Die Menge der entscheidbaren Sprachen ist bzgl. der Operation \min abgeschlossen mit

$$\min(L) := \{x \in L \mid \text{kein echtes Präfix von } x \text{ ist in } L\}$$

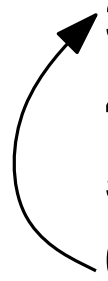
Idee:

- Die TM T_L entscheidet die Sprache $L \subseteq \Sigma^*$
- T' generiert alle echten Präfixe ihrer Eingabe ohne Wiederholung.

Arbeitsweise der TM T , die $\min(L)$ entscheidet mit der Eingabe x .

1. T_L entscheidet x .
2. Wenn T_L nicht akzeptiert, hält T und akzeptiert nicht.
3. T' generiert das nächste echte Präfix p von x .
4. Es gibt kein weiteres Präfix mehr: T akzeptiert.
5. T_L entscheidet p .
6. Wenn $p \in L$, dann hält T und akzeptiert nicht.

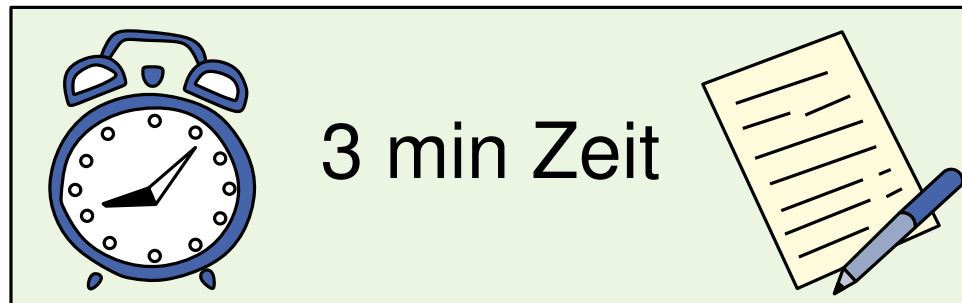
sonst



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit zwei Köpfen (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit zwei Köpfen (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Sobald \mathcal{M}_1 auf Band 1 akzeptiert \Rightarrow akzeptiere Eingabe

Sobald \mathcal{M}_2 auf Band 2 akzeptiert \Rightarrow akzeptiere Eingabe

Für genau die Eingaben in $L_1 \cup L_2$ tritt mindestens einer dieser Fälle ein! $\Rightarrow \mathcal{M}'$ akzeptiert $L_1 \cup L_2$, Vereinigung ist semi-entscheidbar

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit zwei Köpfen (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Sobald \mathcal{M}_1 auf Band 1 akzeptiert \Rightarrow akzeptiere Eingabe

Sobald \mathcal{M}_2 auf Band 2 akzeptiert \Rightarrow akzeptiere Eingabe

Für genau die Eingaben in $L_1 \cup L_2$ tritt mindestens einer dieser Fälle ein! $\Rightarrow \mathcal{M}'$ akzeptiert $L_1 \cup L_2$, Vereinigung ist semi-entscheidbar



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit zwei Köpfen (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit zwei Köpfen (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Sobald \mathcal{M}_1 auf Band 1 akzeptiert und \mathcal{M}_2 auf Band 2 akzeptiert \Rightarrow akzeptiere Eingabe

Für genau die Eingaben in $L_1 \cap L_2$ tritt dieser Fall ein! $\Rightarrow \mathcal{M}'$ akzeptiert $L_1 \cap L_2$, Schnitt ist semi-entscheidbar

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

(c) Die Menge der semi-entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Sei \mathcal{M}_1 TM, die L_1 (semi-entscheidbar) akzeptiert, also $L_1 = L(\mathcal{M}_1)$.

Sei \mathcal{M}_2 TM, die L_2 (semi-entscheidbar) akzeptiert, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit zwei Köpfen (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Sobald \mathcal{M}_1 auf Band 1 akzeptiert und \mathcal{M}_2 auf Band 2 akzeptiert \Rightarrow akzeptiere Eingabe

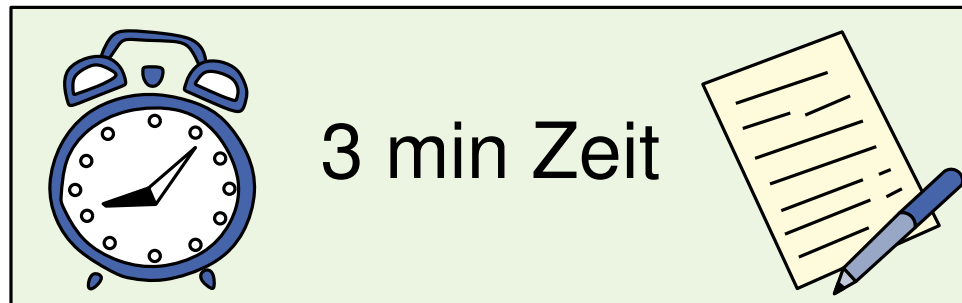
Für genau die Eingaben in $L_1 \cap L_2$ tritt dieser Fall ein! $\Rightarrow \mathcal{M}'$ akzeptiert $L_1 \cap L_2$, Schnitt ist semi-entscheidbar



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.

Idee:

- Verwende universelle Sprache $L_u := \{wv \mid v \in L(T_w)\}$.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.

Idee:

- Verwende universelle Sprache $L_u := \{wv \mid v \in L(T_w)\}$.

Aus der Vorlesung:

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.

Idee:

- Verwende universelle Sprache $L_U := \{wv \mid v \in L(T_w)\}$.

Aus der Vorlesung:

a) L_U ist semi-entscheidbar (Satz 3.15 im Skript, Seite 46).

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.

Idee:

- Verwende universelle Sprache $L_U := \{wv \mid v \in L(T_w)\}$.

Aus der Vorlesung:

- a) L_U ist semi-entscheidbar (Satz 3.15 im Skript, Seite 46).
b) L_U ist nicht entscheidbar (Satz 3.14 im Skript, Seite 45).

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.

Idee:

- Verwende universelle Sprache $L_U := \{wv \mid v \in L(T_w)\}$.

Aus der Vorlesung:

- a) L_U ist semi-entscheidbar (Satz 3.15 im Skript, Seite 46).
- b) L_U ist nicht entscheidbar (Satz 3.14 im Skript, Seite 45).
- c) Für eine Sprache L gilt L und L^c sind semi-entscheidbar gdw. L ist entscheidbar (Vorlesung).

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

d) Die Menge der semi-entscheidbaren Sprachen ist unter Komplementbildung nicht abgeschlossen.

Idee:

- Verwende universelle Sprache $L_U := \{wv \mid v \in L(T_w)\}$.

Aus der Vorlesung:

- a) L_U ist semi-entscheidbar (Satz 3.15 im Skript, Seite 46).
- b) L_U ist nicht entscheidbar (Satz 3.14 im Skript, Seite 45).
- c) Für eine Sprache L gilt L und L^c sind semi-entscheidbar gdw. L ist entscheidbar (Vorlesung).

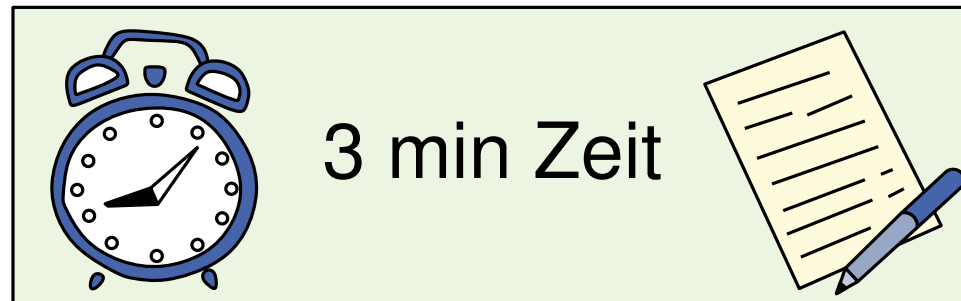
Annahme: L_U^c ist semi-entscheidbar. \Rightarrow Da L_U semi-entscheidbar ist, wäre damit L_U entscheidbar, im Widerspruch zu L_U ist nicht entscheidbar.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

Hinweis: Die Menge der entscheidbaren Sprachen ist unter Komplementbildung abgeschlossen



Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 entscheidet, also $L_1 = L(\mathcal{M}_1)$ und sei \mathcal{M}_2 TM, die L_2 entscheidet, also $L_2 = L(\mathcal{M}_2)$.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 entscheidet, also $L_1 = L(\mathcal{M}_1)$ und sei \mathcal{M}_2 TM, die L_2 entscheidet, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit einem Kopf (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 entscheidet, also $L_1 = L(\mathcal{M}_1)$ und sei \mathcal{M}_2 TM, die L_2 entscheidet, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit einem Kopf (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Falls \mathcal{M}_1 auf Band 1 akzeptiert \Rightarrow akzeptiere Eingabe

Sonst Wechsel auf Band 2. Falls \mathcal{M}_2 auf Band 2 akzeptiert \Rightarrow akzeptiere Eingabe

Sonst stopp in nicht-akzeptierendem Zustand.

Für genau die Eingaben in $L_1 \cup L_2$ tritt ein akzeptierender Fall ein!

$\Rightarrow \mathcal{M}'$ akzeptiert $L_1 \cup L_2$ und stoppt immer, Vereinigung ist entscheidbar

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

1. Vereinigung

Sei \mathcal{M}_1 TM, die L_1 entscheidet, also $L_1 = L(\mathcal{M}_1)$ und sei \mathcal{M}_2 TM, die L_2 entscheidet, also $L_2 = L(\mathcal{M}_2)$.

Benutze 2-Band-TM \mathcal{M}' mit einem Kopf (Skript, Seite 41/42):

- Simuliere \mathcal{M}_1 auf Band 1
- Simuliere \mathcal{M}_2 auf Band 2

Falls \mathcal{M}_1 auf Band 1 akzeptiert \Rightarrow akzeptiere Eingabe

Sonst Wechsel auf Band 2. Falls \mathcal{M}_2 auf Band 2 akzeptiert \Rightarrow akzeptiere Eingabe

Sonst stopp in nicht-akzeptierendem Zustand.

Für genau die Eingaben in $L_1 \cup L_2$ tritt ein akzeptierender Fall ein!

$\Rightarrow \mathcal{M}'$ akzeptiert $L_1 \cup L_2$ und stoppt immer, Vereinigung ist entscheidbar

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Verwende De Morgan Gesetz für Mengen $(\overline{A \cup B}) = (\overline{A} \cap \overline{B})$. Betrachte entscheidbare Sprachen L_1, L_2 als Mengen $A := L_1, B := L_2$.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Verwende De Morgan Gesetz für Mengen $(\overline{A \cup B}) = \overline{A \cap B}$. Betrachte entscheidbare Sprachen L_1, L_2 als Mengen $A := L_1, B := L_2$.

Dann gilt $\overline{A} = L_1^c$ und $\overline{B} = L_2^c$ und $(\overline{A \cup B}) = (L_1^c \cap L_2^c) = (L_1 \cup L_2)^c = \overline{A \cap B}$.

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Verwende De Morgan Gesetz für Mengen $(\overline{A \cup B}) = \overline{A \cap B}$. Betrachte entscheidbare Sprachen L_1, L_2 als Mengen $A := L_1, B := L_2$.

Dann gilt $\overline{A} = L_1^c$ und $\overline{B} = L_2^c$ und $(\overline{A \cup B}) = (L_1^c \cap L_2^c) = (L_1 \cup L_2)^c = \overline{A \cap B}$.

$\Rightarrow (L_1 \cup L_2)^c$ entscheidbar

$\Rightarrow ((L_1 \cup L_2)^c)^c = L_1 \cup L_2$ entscheidbar

Aufgabe – Abgeschlossenheit von entscheidbaren Sprachen

Zeigen Sie:

- e) Die Menge der entscheidbaren Sprachen ist unter Vereinigung und Schnitt abgeschlossen.

2. Schnitt

Verwende De Morgan Gesetz für Mengen $(\overline{A \cup B}) = \overline{A \cap B}$. Betrachte entscheidbare Sprachen L_1, L_2 als Mengen $A := L_1, B := L_2$.

Dann gilt $\overline{A} = L_1^c$ und $\overline{B} = L_2^c$ und $(\overline{A \cup B}) = (L_1^c \cap L_2^c) = (L_1 \cup L_2)^c = \overline{A \cap B}$.

$\Rightarrow (L_1 \cup L_2)^c$ entscheidbar

$\Rightarrow ((L_1 \cup L_2)^c)^c = L_1 \cup L_2$ entscheidbar



Aufgaben zu Komplexitätsklassen



Aufgabe – P und NP

Das Entscheidungsproblem Π , ob eine gegebene Zahl eine Potenz von 2 ist, ist durch die Problembeispiele $D_\Pi := \mathbb{N}$ und die Ja-Beispiele $J_\Pi := \{2^i \mid i \in \mathbb{N}\}$ gegeben. Seien s_b die Kodierungsschemata, die natürliche Zahlen auf ihre b -äre Repräsentation abbilden.

Betrachten Sie nun $L[\Pi, s_1]$ und $L[\Pi, s_2]$. Beschreiben Sie für jede der beiden Sprachen kurz die Arbeitsweise einer deterministischen TM, die sie entscheidet und geben Sie ihre Laufzeit asymptotisch an. Sind die Sprachen in P? Sind sie in NP?

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10\dots 0$.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
- Sonst gehe schrittweise nach rechts bis zum Ende der Eingabe.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
- Sonst gehe schrittweise nach rechts bis zum Ende der Eingabe.
- Überprüfe dabei, ob nach der führenden 1 noch eine 1 vorkommt.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
- Sonst gehe schrittweise nach rechts bis zum Ende der Eingabe.
- Überprüfe dabei, ob nach der führenden 1 noch eine 1 vorkommt.
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
- Sonst gehe schrittweise nach rechts bis zum Ende der Eingabe.
- Überprüfe dabei, ob nach der führenden 1 noch eine 1 vorkommt.
- Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
- Sonst akzeptiere die Eingabe.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
 - Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
 - Sonst gehe schrittweise nach rechts bis zum Ende der Eingabe.
 - Überprüfe dabei, ob nach der führenden 1 noch eine 1 vorkommt.
 - Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
 - Sonst akzeptiere die Eingabe.
- ⇒ Zeitkomplexität der TM ist linear in der Eingabegröße.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_2]$ zu Problem $\Pi = (D_\Pi := \mathbb{N}, J_{Pi} := \{2^i \mid i \in \mathbb{N}\})$.

Konvention: Es gibt keine führenden Nullen (außer die Eingabe ist 0).

Beobachtung: Die Zweierpotenzen haben in Binärdarstellung genau die Form $10 \dots 0$.

- Überprüfe, ob die Eingabe 0 ist (also ob an erster Stelle eine 1 steht).
 - Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
 - Sonst gehe schrittweise nach rechts bis zum Ende der Eingabe.
 - Überprüfe dabei, ob nach der führenden 1 noch eine 1 vorkommt.
 - Falls ja, stoppe die Berechnung und lehne die Eingabe ab.
 - Sonst akzeptiere die Eingabe.
- ⇒ Zeitkomplexität der TM ist linear in der Eingabegröße.
- ⇒ $L[\Pi, s_2]$ liegt in P.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.
- Bei jedem Durchlauf, merke ob gerade oder ungerade viele 1-en auf dem Band stehen.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.
- Bei jedem Durchlauf, merke ob gerade oder ungerade viele 1-en auf dem Band stehen.
- Bei jedem Durchlauf ersetze jede zweite 1 durch eine 0.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.
- Bei jedem Durchlauf, merke ob gerade oder ungerade viele 1-en auf dem Band stehen.
- Bei jedem Durchlauf ersetze jede zweite 1 durch eine 0.
- Wenn bei einem Durchlauf ungerade viele 1-en erkannt wurden, stoppe und lehne die Eingabe ab.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.
- Bei jedem Durchlauf, merke ob gerade oder ungerade viele 1-en auf dem Band stehen.
- Bei jedem Durchlauf ersetze jede zweite 1 durch eine 0.
- Wenn bei einem Durchlauf ungerade viele 1-en erkannt wurden, stoppe und lehne die Eingabe ab.
- Ansonsten, falls am Ende eine 1 stehen bleibt, akzeptiere die Eingabe.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.
- Bei jedem Durchlauf, merke ob gerade oder ungerade viele 1-en auf dem Band stehen.
- Bei jedem Durchlauf ersetze jede zweite 1 durch eine 0.
- Wenn bei einem Durchlauf ungerade viele 1-en erkannt wurden, stoppe und lehne die Eingabe ab.
- Ansonsten, falls am Ende eine 1 stehen bleibt, akzeptiere die Eingabe.

⇒ Zeitkomplexität der TM ist *quadratisch* in der Eingabegröße.

Aufgabe – P und NP

Wir betrachten $L[\Pi, s_1]$ zu dem Problem $\Pi = (D_\Pi := \mathbb{N}, J_\Pi := \{2^i \mid i \in \mathbb{N}\})$.

Eine TM die $L[\Pi, s_1]$ entscheidet kann wie folgt konstruiert werden:

- Durchlaufe immer wieder die Eingabe.
- Bei jedem Durchlauf, merke ob gerade oder ungerade viele 1-en auf dem Band stehen.
- Bei jedem Durchlauf ersetze jede zweite 1 durch eine 0.
- Wenn bei einem Durchlauf ungerade viele 1-en erkannt wurden, stoppe und lehne die Eingabe ab.
- Ansonsten, falls am Ende eine 1 stehen bleibt, akzeptiere die Eingabe.

⇒ Zeitkomplexität der TM ist *quadratisch* in der Eingabegröße.

⇒ $L[\Pi, s_1]$ liegt in P

Aufgabe – Komplexitätsklassen

- a) Definieren Sie PSPACE und EXPTIME, und beschreiben Sie diese Klassen mit eigenen Worten.
- b) Betrachten Sie die Klassen L, NLOG, P, NP, PSPACE, NPSPACE, EXPSPACE und EXP näher:
 - Stellen Sie die Beziehung zwischen diesen Klassen mithilfe eines geeigneten Diagrammtyps dar.
 - Geben Sie Probleme an, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?
 - Welche Folgen ergeben sich, falls $P = NP$? (Hinweis: Was bedeutet das in der Praxis, Kryptographie, . . .)

Aufgabe – Komplexitätsklassen

- a) Definieren Sie PSPACE und EXP, und beschreiben Sie diese Klassen mit eigenen Worten.

PSPACE

Klasse von Entscheidungsproblemen, die von einer DTM in polynomial viel Platz gelöst werden.

Aufgabe – Komplexitätsklassen

- a) Definieren Sie PSPACE und EXP, und beschreiben Sie diese Klassen mit eigenen Worten.

PSPACE

Klasse von Entscheidungsproblemen, die von einer TM in polynomial viel Platz gelöst werden.

Aufgabe – Komplexitätsklassen

a) Definieren Sie PSPACE und EXP, und beschreiben Sie diese Klassen mit eigenen Worten.

PSPACE

Klasse von Entscheidungsproblemen, die von einer TM in polynomial viel Platz gelöst werden.

EXP

Klasse von Entscheidungsproblemen, die von einer DTM in $\mathcal{O}(2^{p(n)})$ Zeit gelöst werden können.

Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.

Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.

L

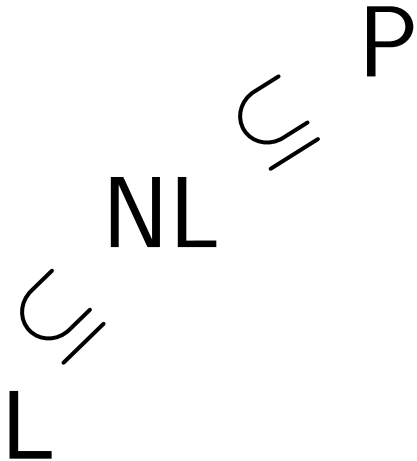
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.

$L \subset NL$

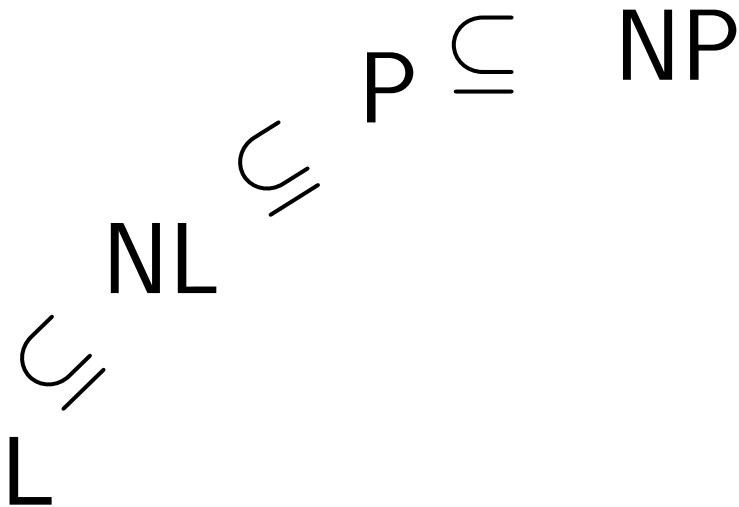
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.



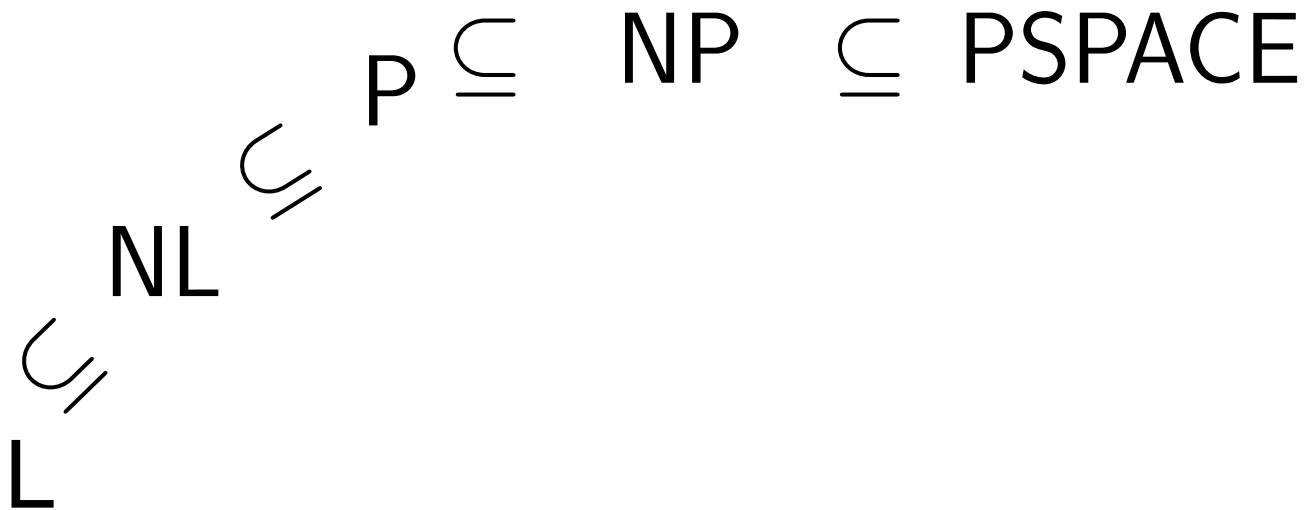
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.



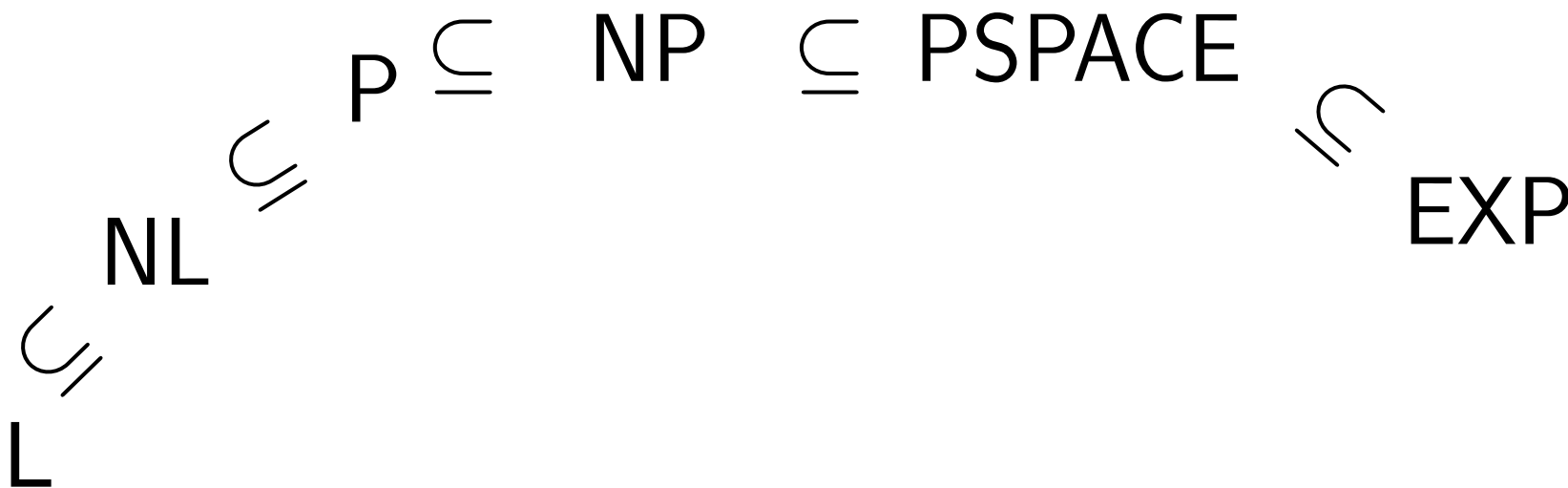
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.



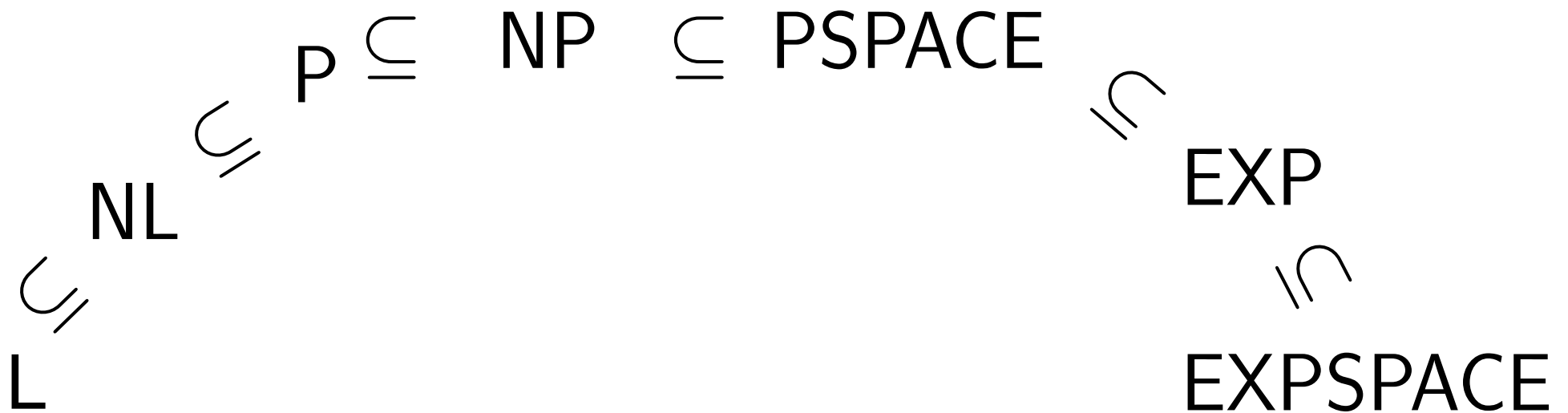
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.



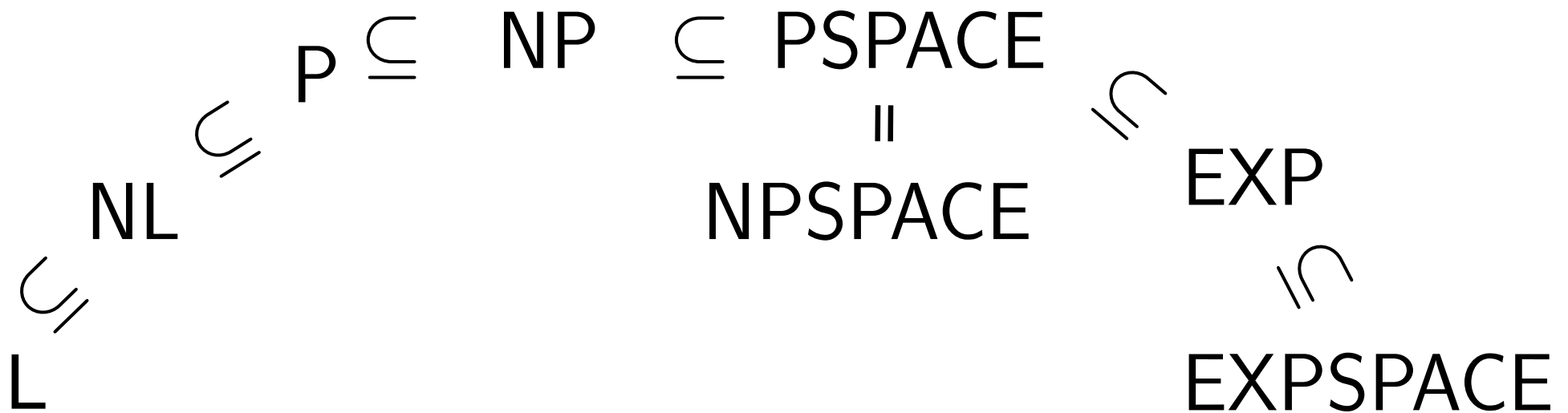
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.



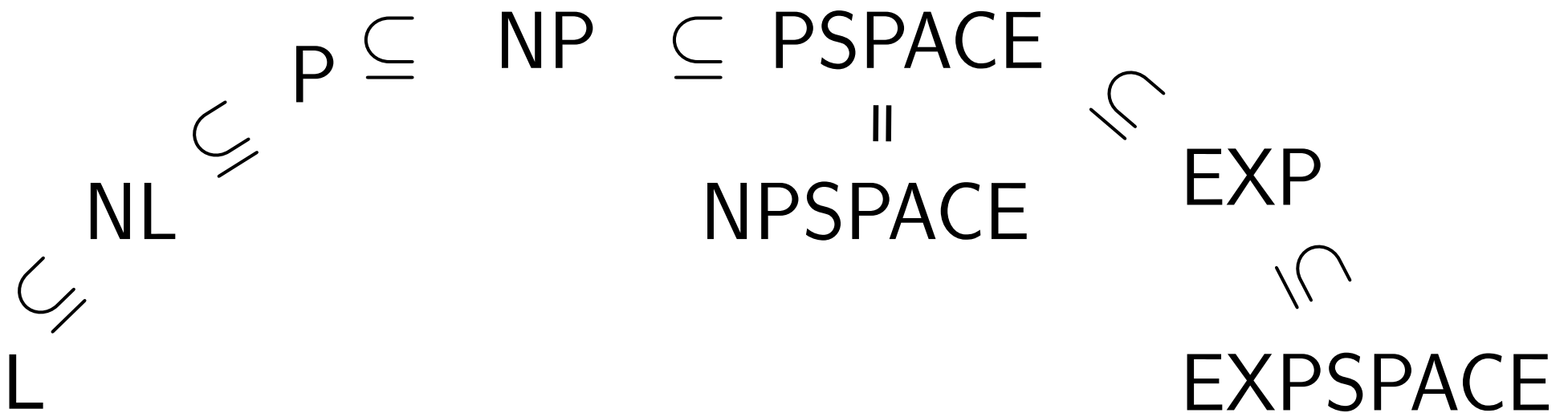
Aufgabe – Komplexitätsklassen

Beziehung zwischen den Klassen: L, NL, P, NP, PSPACE, NPSPACE, EXPSPACE, EXP.



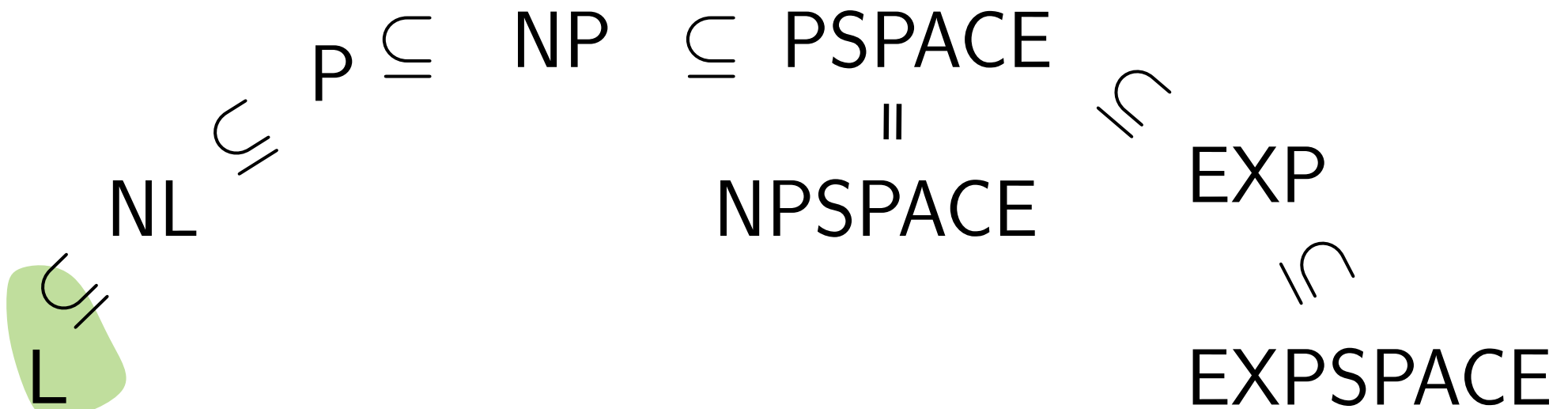
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



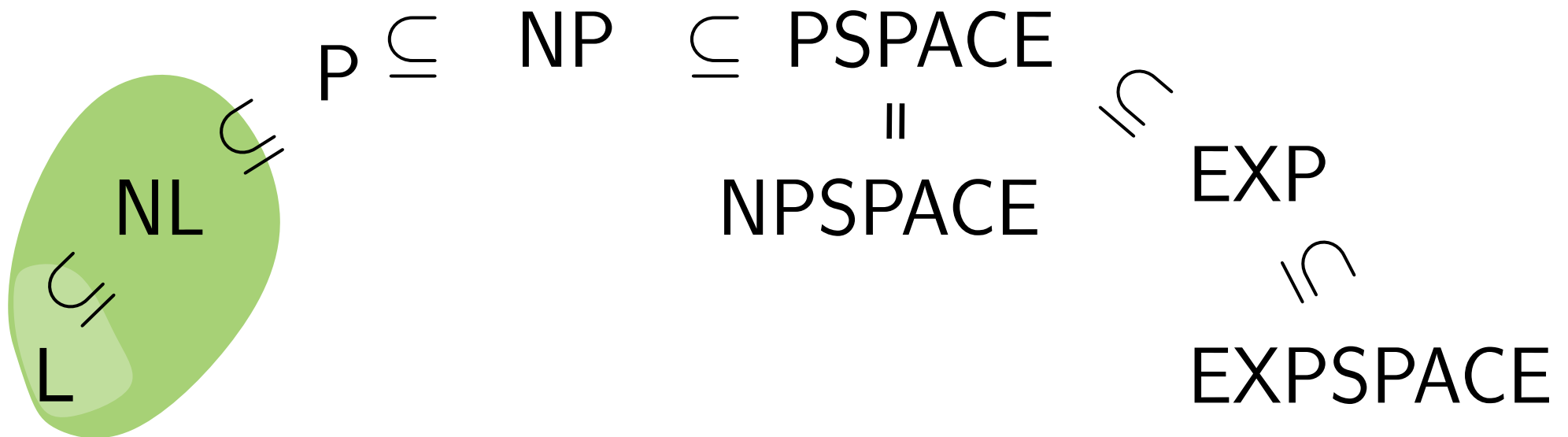
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



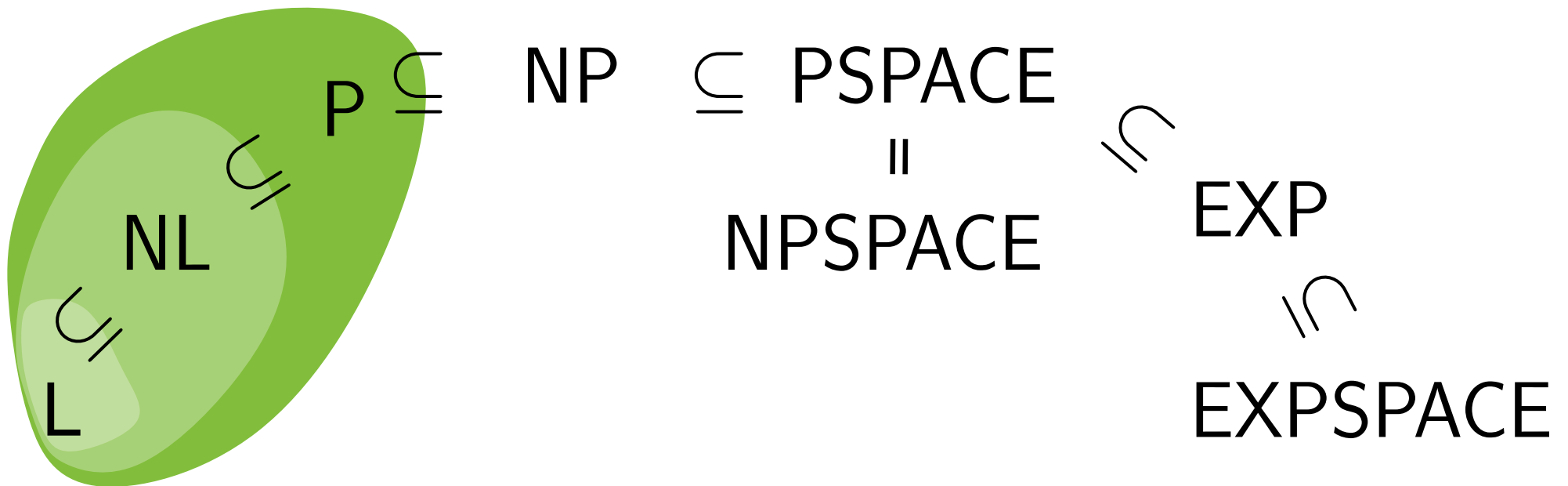
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



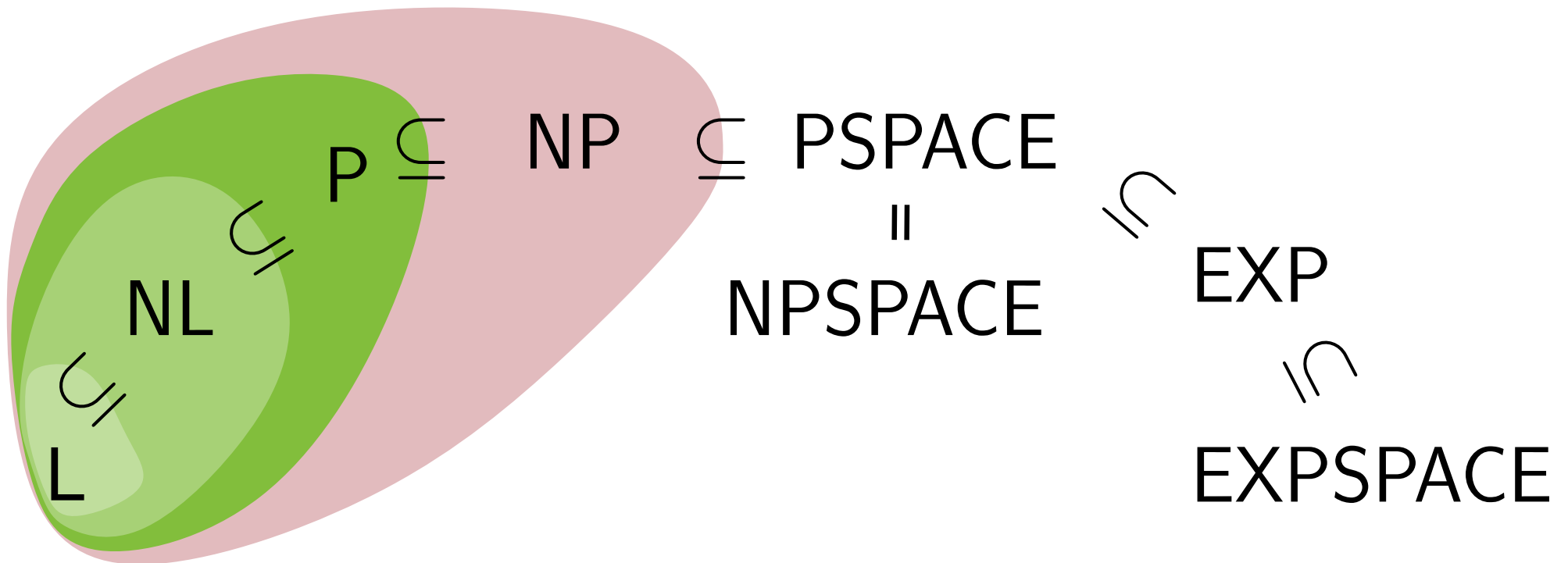
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



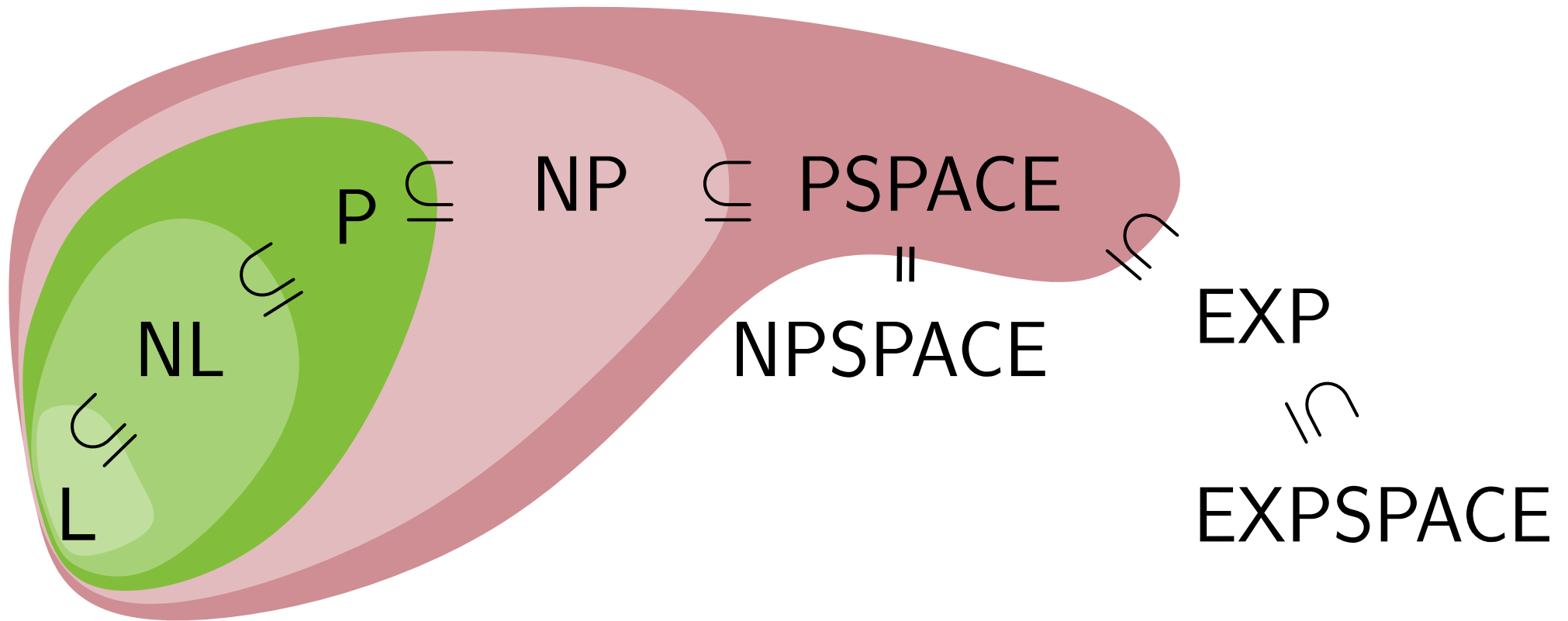
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



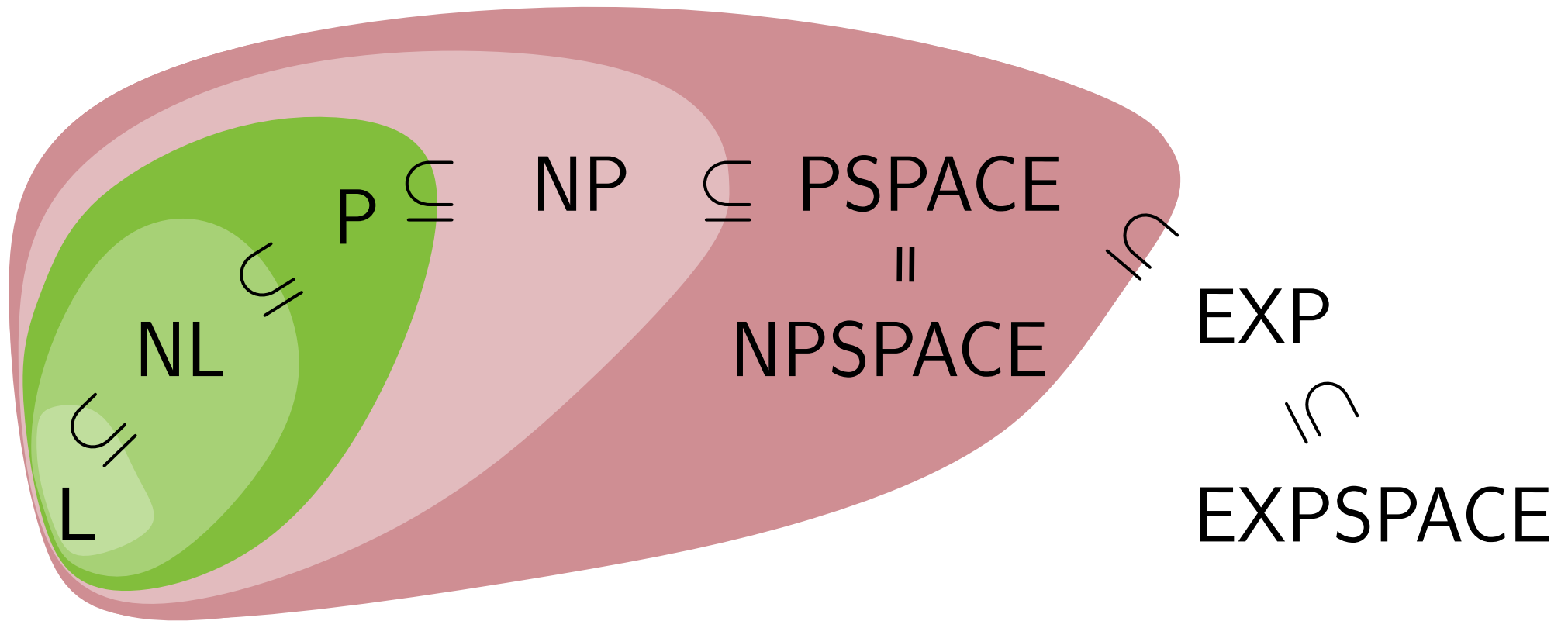
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



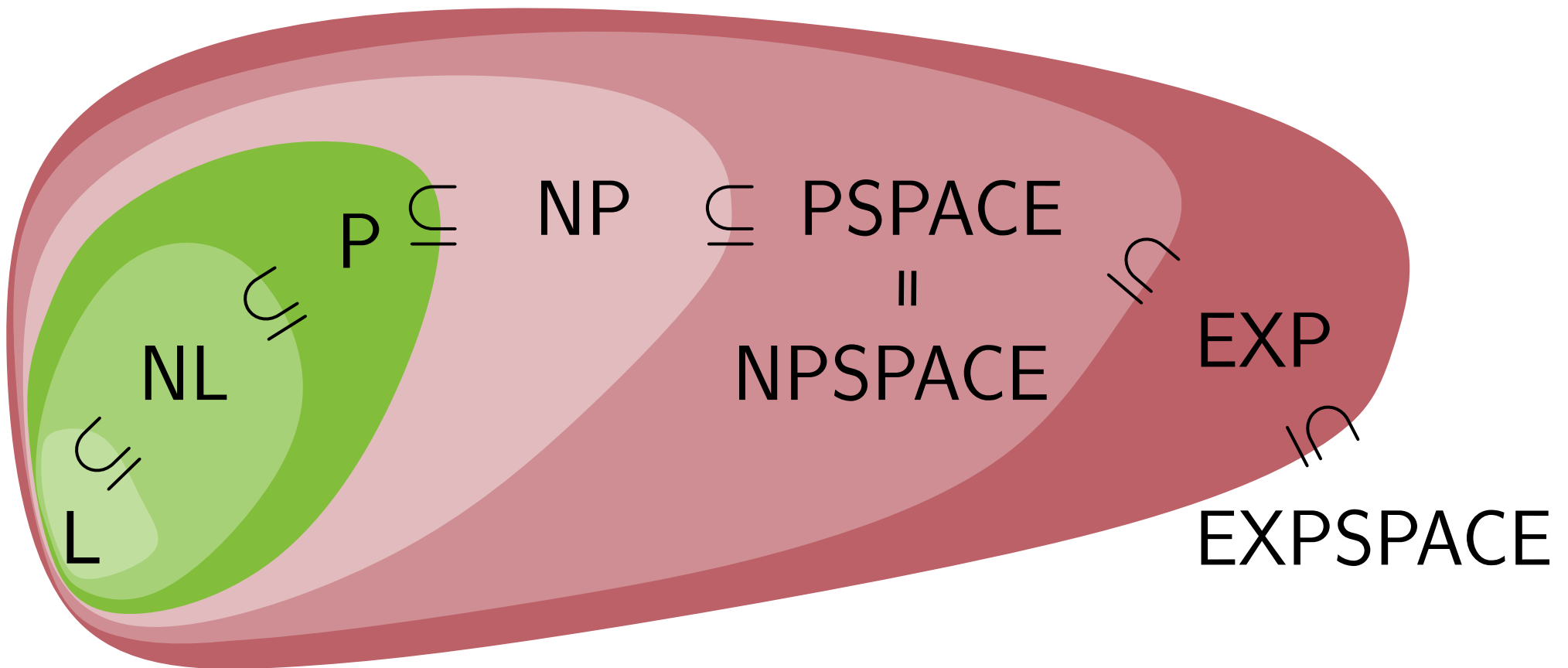
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



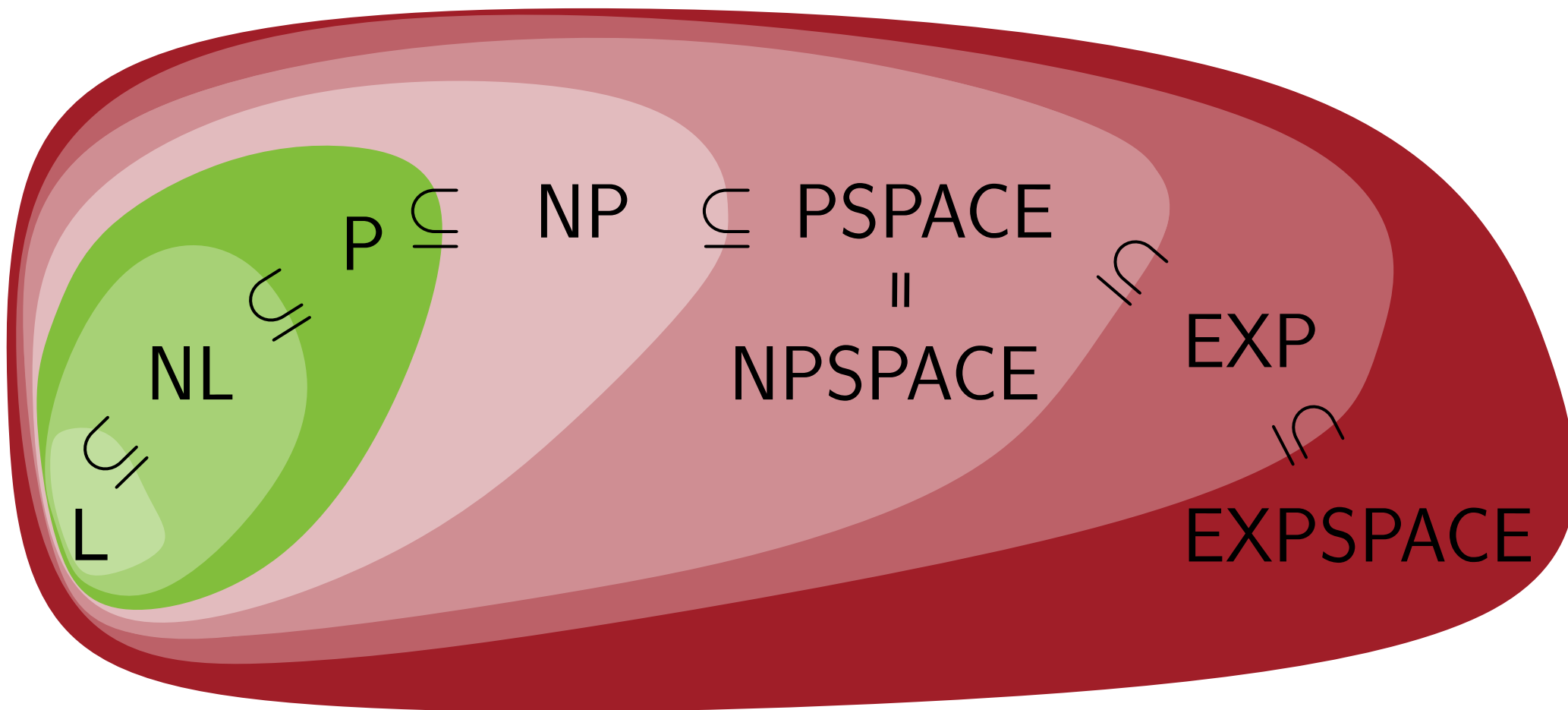
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



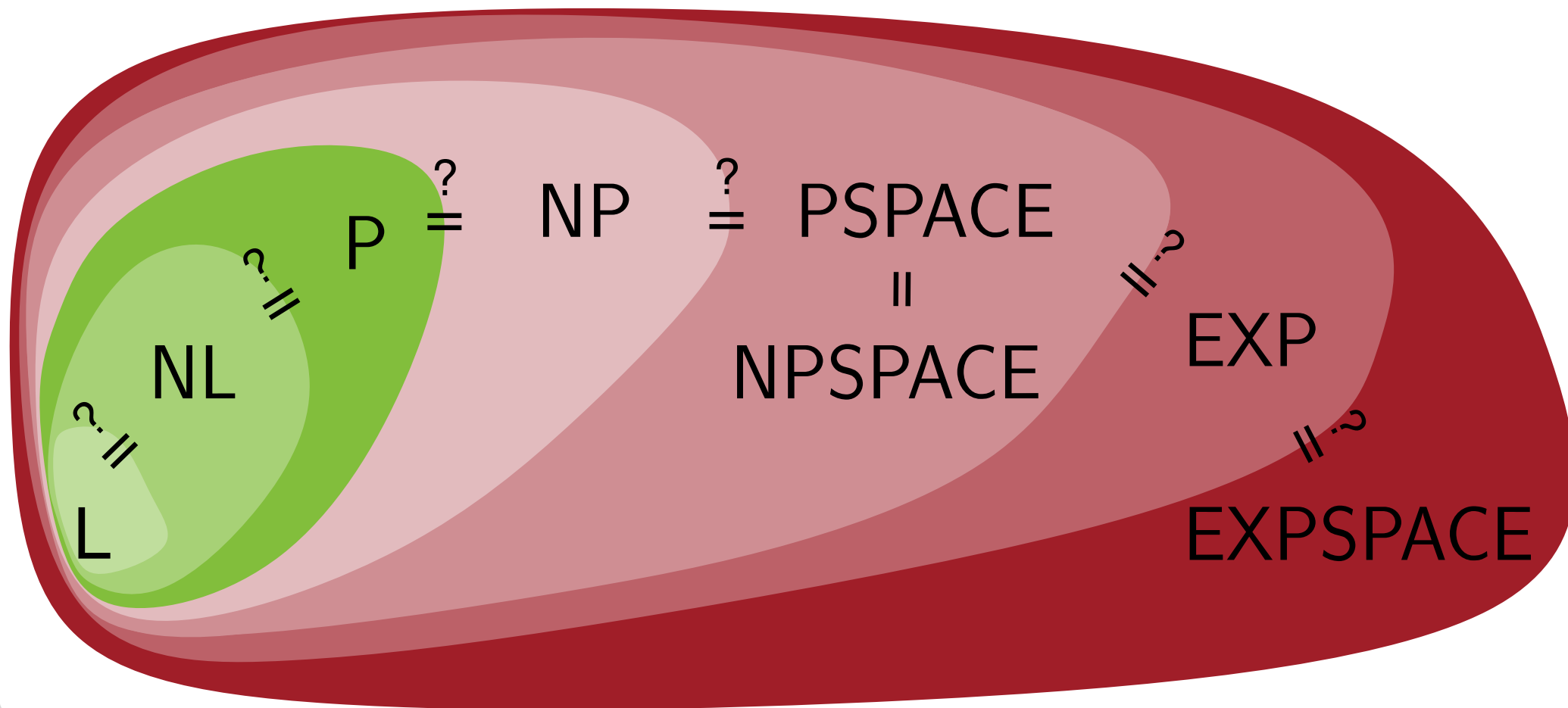
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



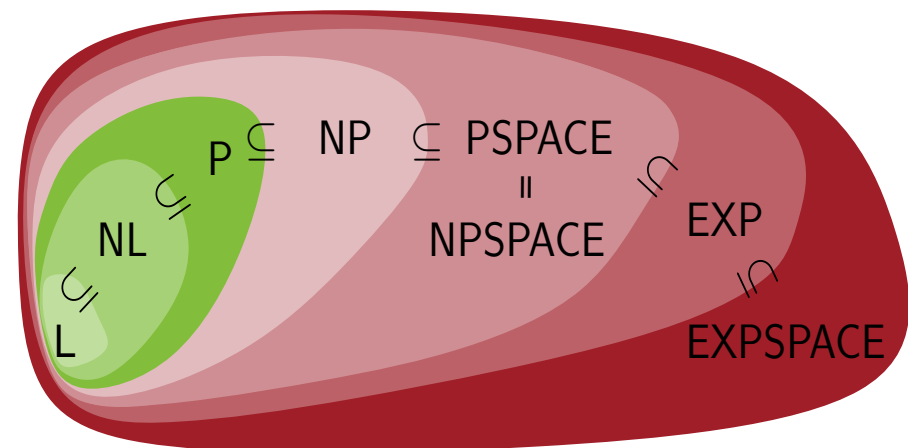
Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



Aufgabe – Komplexitätsklassen

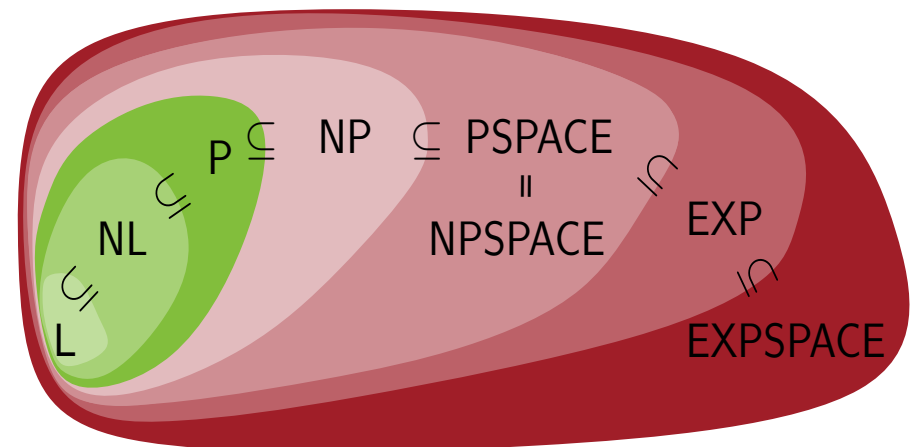
Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?



Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?

Problem des Handlungsreisenden

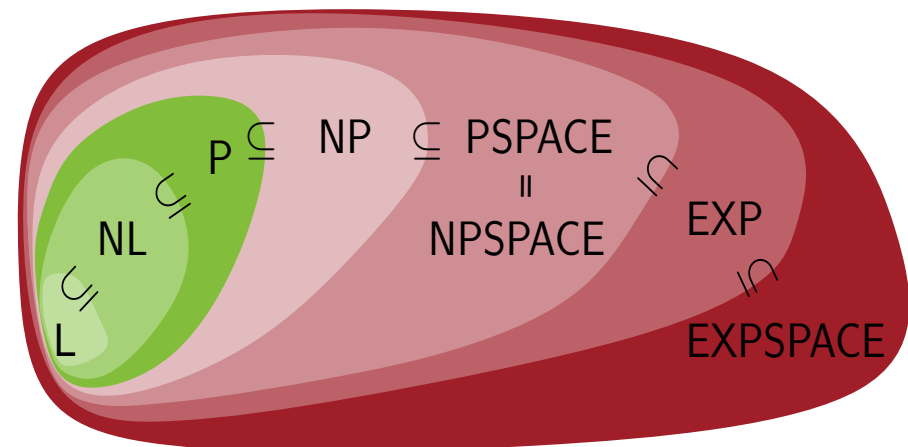


Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?

Problem des Handlungsreisenden

- Mögliches exaktes Lösungsverfahren: Alle Weglängen aller möglichen Rundreisen berechnen

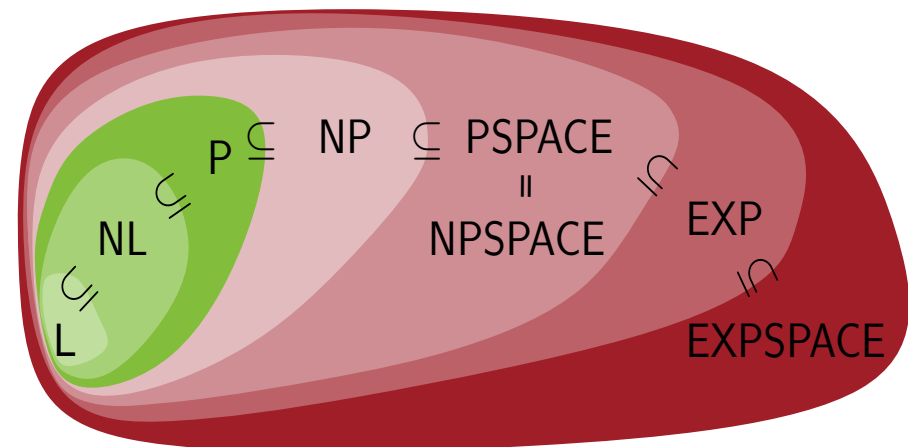


Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?

Problem des Handlungsreisenden

- Mögliches exaktes Lösungsverfahren: Alle Weglängen aller möglichen Rundreisen berechnen
- Schon bei kleiner Anzahl von Städten unpraktikabel

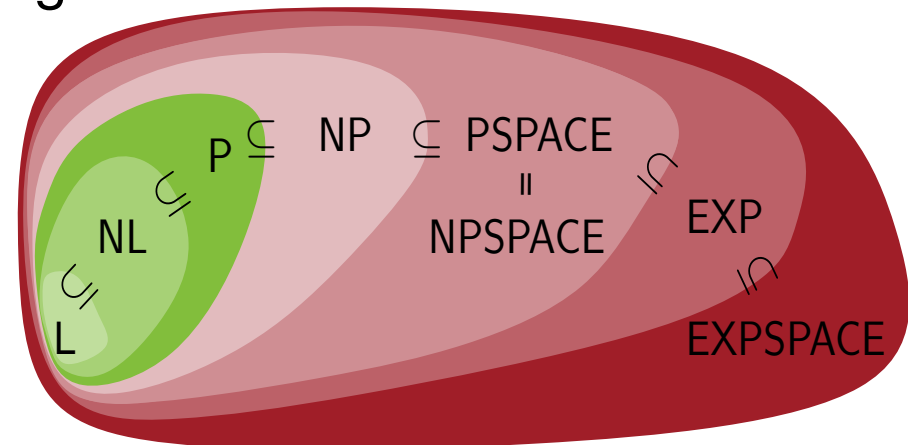


Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?

Problem des Handlungsreisenden

- Mögliches exaktes Lösungsverfahren: Alle Weglängen aller möglichen Rundreisen berechnen
- Schon bei kleiner Anzahl von Städten unpraktikabel
- Bei n Städten $\rightarrow \frac{(n-1)!}{2}$ verschieden mögliche Rundreisen

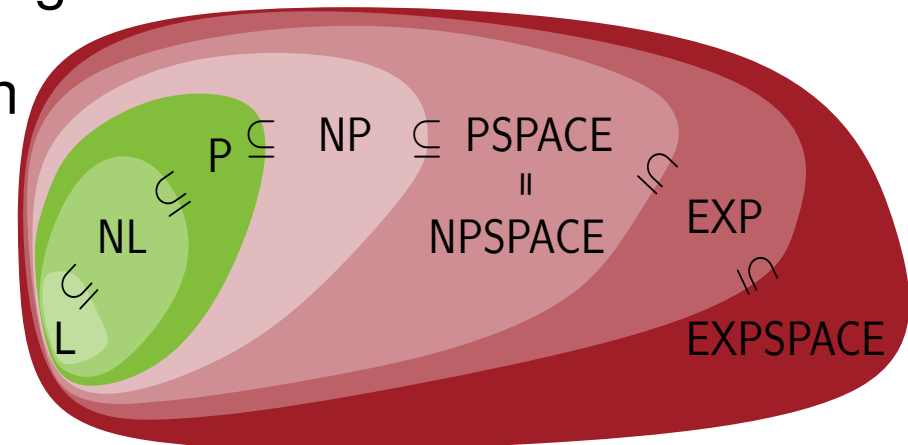


Aufgabe – Komplexitätsklassen

Probleme, die für heutige Computer bzgl. sinnvoll verwendeter Rechenzeit und Speicherplatz nicht lösbar sind. Zu welchen Komplexitätsklassen gehören diese Probleme?

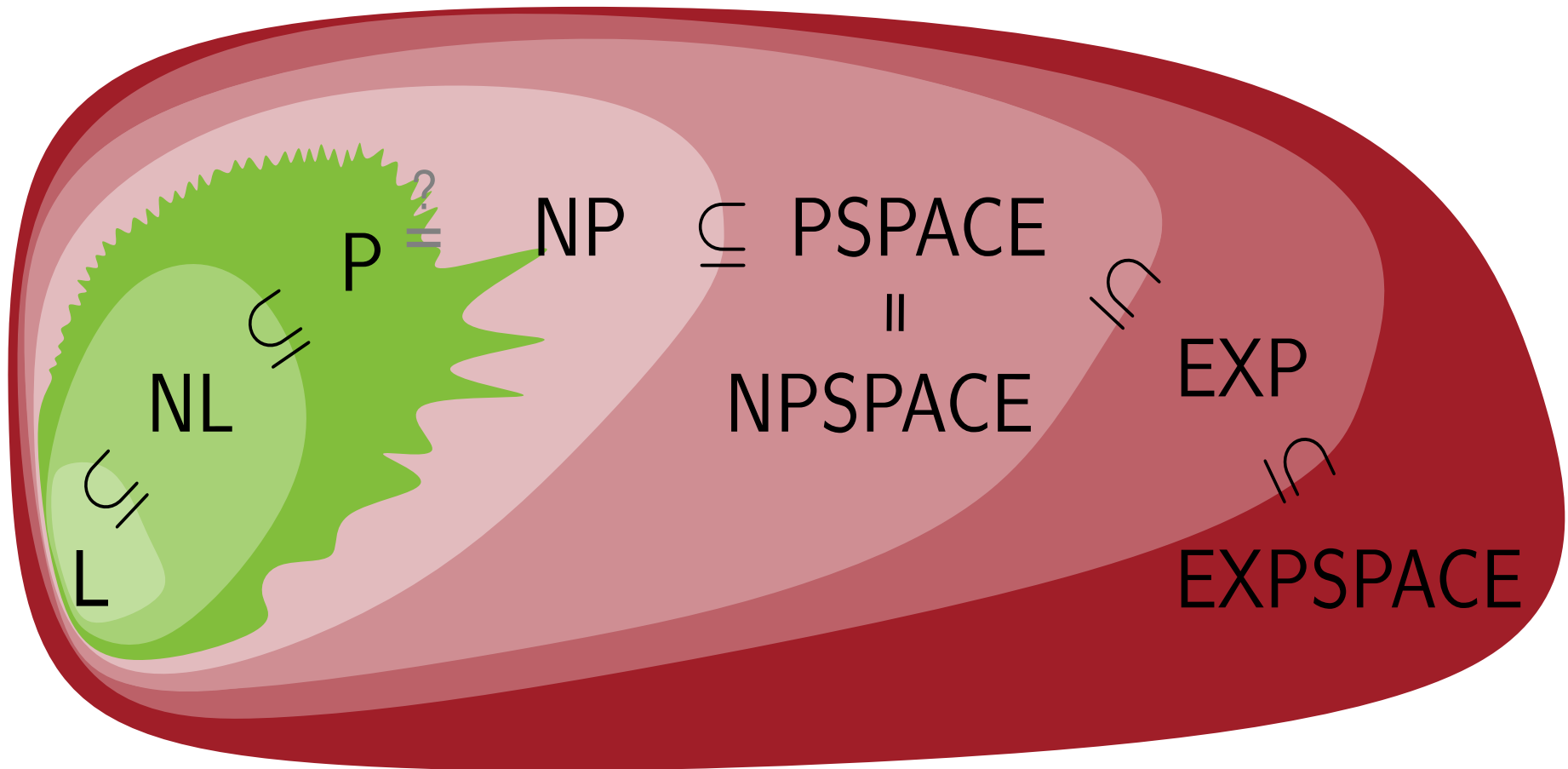
Problem des Handlungsreisenden

- Mögliches exaktes Lösungsverfahren: Alle Weglängen aller möglichen Rundreisen berechnen
- Schon bei kleiner Anzahl von Städten unpraktikabel
- Bei n Städten $\rightarrow \frac{(n-1)!}{2}$ verschieden mögliche Rundreisen
- Für 16 Städte \rightarrow 653 Mrd. Rundreisen



Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$



Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

- Eines der Millenium-Preis-Probleme

Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?

Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht

Aufgabe – Komplexitätsklassen

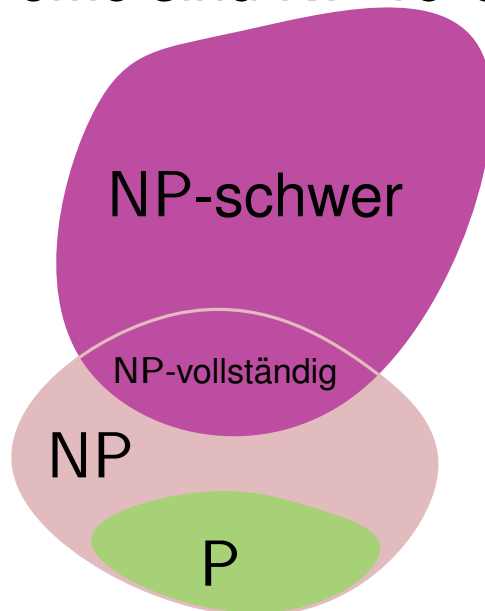
$$P \stackrel{?}{=} NP$$

- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht
- Viele Probleme sind NP-vollständig

Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

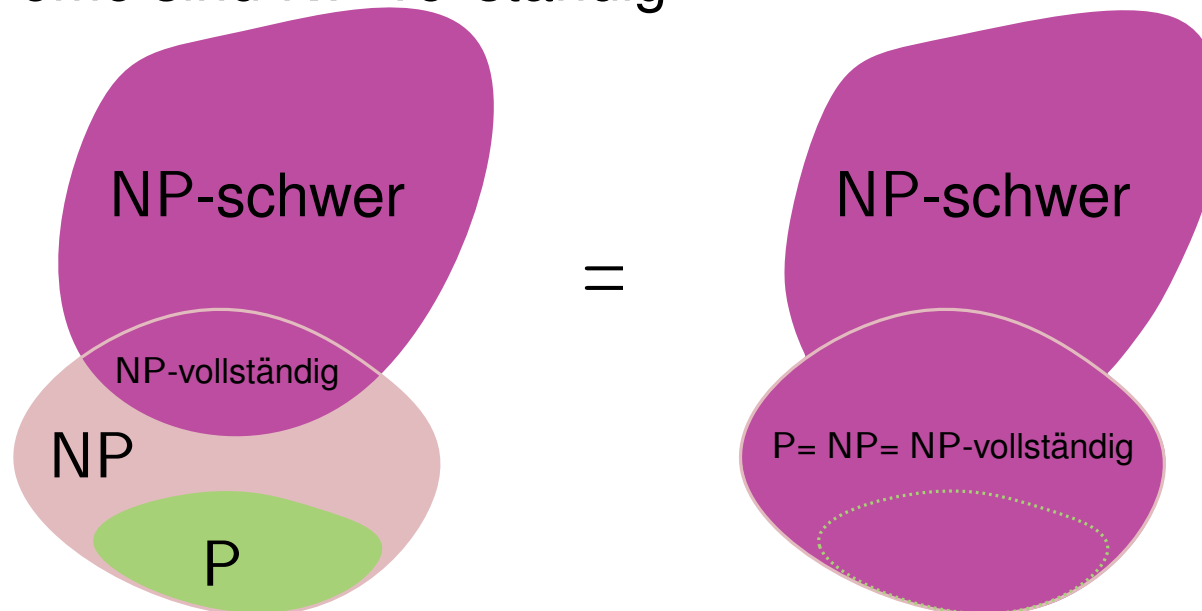
- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht
- Viele Probleme sind NP-vollständig



Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

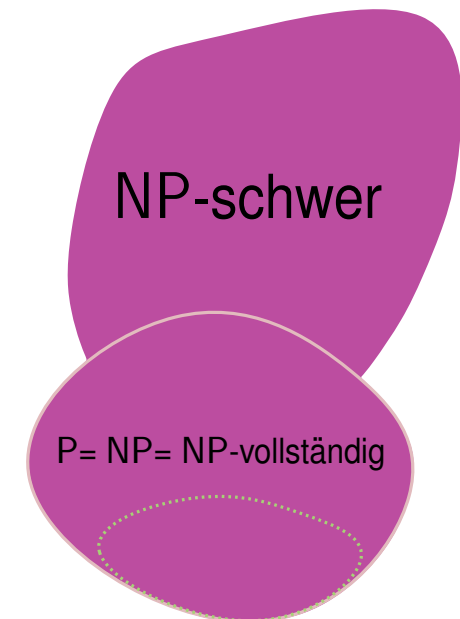
- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht
- Viele Probleme sind NP-vollständig



Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

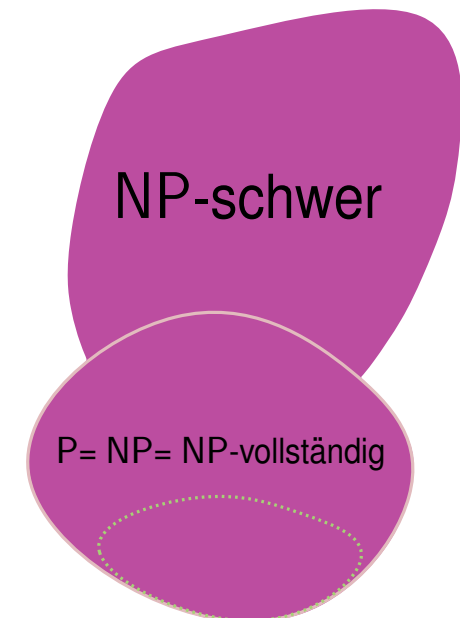
- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht
- Viele Probleme sind NP-vollständig
- **Beispiel: Kryptography**



Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

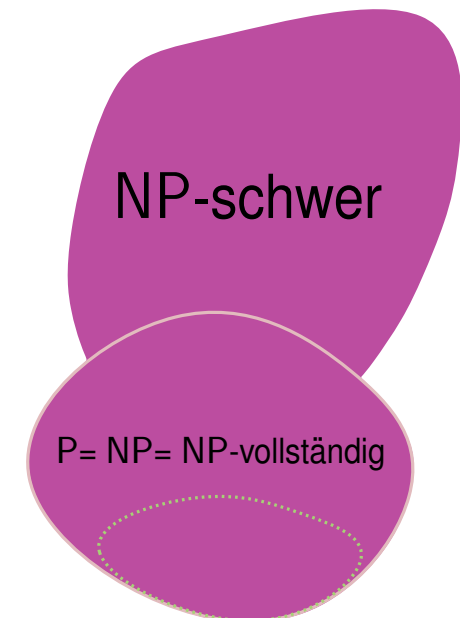
- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht
- Viele Probleme sind NP-vollständig
- **Beispiel: Kryptography**
 - Viele Verfahren vertrauen darauf, dass $P \neq NP$



Aufgabe – Komplexitätsklassen

$$P \stackrel{?}{=} NP$$

- Eines der Millenium-Preis-Probleme
- Können Probleme in NP genauso effizient gelöst werden, wie in P?
- Wenn der Beweis nicht konstruktiv ist, weiß man die Auswirkung nicht
- Viele Probleme sind NP-vollständig
- **Beispiel: Kryptography**
 - Viele Verfahren vertrauen darauf, dass $P \neq NP$
 - Konstruktive und effiziente Lösung eines NP-vollständigen Problems bricht die meisten Verfahren, wie Public-Key-Verfahren, AES, 3DES, ...



Aufgabe – 15 Puzzle

15-Puzzle

13	10	11	6
5	7	4	8
1	12	14	9
3	15	2	

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

- Formulieren Sie das Problem 15-Puzzle als Optimierungs-, Optimalwert- und Entscheidungsproblem
- Geben Sie ein Kodierungsschema an und bestimmen Sie die Kodierungslänge der Instanzen.

Aufgabe – 15 Puzzle

Optimierungsproblem:

13	10	11	6
5	7	4	8
1	12	14	9
3	15	2	

Optimalwertproblem:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Entscheidungsproblem:

Aufgabe – 15 Puzzle

Optimierungsproblem:

Gegeben: Anfangskonfiguration von 15-Puzzle

Gesucht: Folge von Zügen mit minimaler Länge, die die Anfangskonfiguration in die Zielkonfiguration überführt.

13	10	11	6
5	7	4	8
1	12	14	9
3	15	2	

Optimalwertproblem:

Entscheidungsproblem:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Aufgabe – 15 Puzzle

Optimierungsproblem:

Gegeben: Anfangskonfiguration von 15-Puzzle

Gesucht: Folge von Zügen mit minimaler Länge, die die Anfangskonfiguration in die Zielkonfiguration überführt.

13	10	11	6
5	7	4	8
1	12	14	9
3	15	2	

Optimalwertproblem:

Gegeben: Anfangskonfiguration von 15-Puzzle

Gesucht: Die Länge einer kürzesten Folge von Zügen, die die Anfangskonfiguration in die Zielkonfiguration überführt.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Entscheidungsproblem:

Aufgabe – 15 Puzzle

Optimierungsproblem:

Gegeben: Anfangskonfiguration von 15-Puzzle

Gesucht: Folge von Zügen mit minimaler Länge, die die Anfangskonfiguration in die Zielkonfiguration überführt.

13	10	11	6
5	7	4	8
1	12	14	9
3	15	2	

Optimalwertproblem:

Gegeben: Anfangskonfiguration von 15-Puzzle

Gesucht: Die Länge einer kürzesten Folge von Zügen, die die Anfangskonfiguration in die Zielkonfiguration überführt.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

Entscheidungsproblem:

Gegeben: Anfangskonfiguration von 15-Puzzle und ein Parameter k

Gesucht: Gibt es eine Folge von Zügen der Länge $\leq k$, die die Anfangskonfiguration in die Zielkonfiguration überführt.

Aufgabe – 15 Puzzle

Mögliche Kodierung:

- Eine Konfiguration als Folge v_1, v_2, \dots, v_{16} der 16 Kacheln (inklusive der leeren Kachel)
- Die Kachel mit Nummer i als Hexadezimalzahl i , und
- das leere Feld mit der Hexadezimalzahl 0.

Die Länge der Kodierung ist somit

$$\sum_{i=1}^{16} \langle v_i \rangle = \sum_{i=1}^{16} 1 = 16$$

13	10	11	6
5	7	4	8
1	12	14	9
3	15	2	

D A B 6 5 7 4 8 1 C E 9 3 F 2 0

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	

1 2 3 4 5 6 7 8 9 A B C D E F 0

Allgemeine Wiederholung

Polynomiale Transformation

Eine **polynomiale Transformation** einer Sprache $L_1 \subseteq \Sigma_1^*$ in eine Sprache $L_2 \subseteq \Sigma_2^*$ ist eine Funktion $f: \Sigma_1^* \rightarrow \Sigma_2^*$ mit den Eigenschaften:

- es existiert eine polynomiale deterministische Turing-Maschine, die f berechnet;
- für alle $x \in \Sigma_1^*$ gilt: $x \in L_1 \Leftrightarrow f(x) \in L_2$.

Wir schreiben dann $L_1 \propto L_2$ (L_1 ist polynomial transformierbar in L_2).

Eine Sprache L heißt **NP-vollständig**, falls gilt:

- $L \in \text{NP}$ und
- für alle $L' \in \text{NP}$ gilt $L' \propto L$.

Polynomiale Transformation

Eine **polynomiale Transformation** einer Sprache $L_1 \subseteq \Sigma_1^*$ in eine Sprache $L_2 \subseteq \Sigma_2^*$ ist eine Funktion $f: \Sigma_1^* \rightarrow \Sigma_2^*$ mit den Eigenschaften:

- es existiert eine polynomiale deterministische Turing-Maschine, die f berechnet;
- für alle $x \in \Sigma_1^*$ gilt: $x \in L_1 \Leftrightarrow f(x) \in L_2$.

Wir schreiben dann $L_1 \propto L_2$ (L_1 ist polynomial transformierbar in L_2).

Eine Sprache L heißt **NP-vollständig**, falls gilt:

- $L \in \text{NP}$ und
- für alle $L' \in \text{NP}$ gilt $L' \propto L$.

➔ Wenn ein NP-vollständiges Problem Π in P liegt, dann gilt $\text{NP}=\text{P}$.

Nichtdeterminismus

- Nicht praktischer Natur, da man nichtdet. Maschinen nicht wirklich bauen kann,
- theoretische Konstrukte sind hilfreich, da die Laufzeit von nichtdet. Maschinen deutlich geringer sind,
- liefern wichtige Hinweise auf effizientere Lösungen vieler praktischer Probleme (helfen mögliche Zunahme von Komplexität zu zeigen),
- det. Maschinen \Rightarrow Berechnung bei gegebener Eingabe klar definiert,
- bei nicht deterministischen TM bei nur einem gelesenen Zeichen mehrere Nachfolgezustände möglich,
- keine einzelne Berechnung, sondern viele verschiedene Möglichkeiten der Berechnung,
- min. eine Berechnung führt in einen akzeptierenden Zustand.

Orakel

- Kann man sich als Black Box vorstellen,
- kann von einer TM befragt werden und Probleme in einem Schritt lösen.
- Orakel dienen dazu die Hierarchien von Berechenbarkeit und Komplexität zu definieren und deren Eigenschaften zu studieren.
- Ein geeignetes Orakel \Rightarrow Berechenbarkeit verstärken oder Komplexität verringern.