2. Klausur zur Vorlesung
Theoretische Grundlagen der Informatik
Wintersemester 2023/2024

Hier Aufkleber mit Matrikelnummer anbringen

- Bringen Sie den Aufkleber mit Ihrer Matrikelnummer auf diesem Deckblatt an und beschriften Sie jedes Aufgabenblatt mit Ihrer Matrikelnummer.
- Schreiben Sie die Lösungen auf die Aufgabenblätter und Rückseiten. Am Ende der Klausur sind zusätzliche Leerseiten. Fordern Sie zusätzliches Papier bitte nur an, falls Sie den gesamten Platz aufgebraucht haben.
- Die Tackernadel darf nicht gelöst werden.
- Als Hilfsmittel ist ein beschriebenes A4-Papier erlaubt.
- Einlesezeit: 15 min
  Bearbeitungszeit: 2 h

	Mögliche Punkte						Erreichte Punkte					
	a	b	c	d	e	Σ	a	b	c	d	e	Σ
Aufg. 1	2	3	3	3	–	11					–	
Aufg. 2	2	2	2	3	–	9					–	
Aufg. 3	1	3	2	3	–	9					–	
Aufg. 4	1	7	2	–	–	10				–	–	
Aufg. 5	2	1	4	2	–	9					–	
Aufg. 6	3	1	1	5	2	12						
Σ						60						

Problem 1: Automaten 2+3+3+3= 11 Punkte

Für ein Wort $w = w_1 w_2 \dots w_n$ bezeichnen wir mit w_i das i-te Zeichen von w. Die Umkehrung $w_n w_{n-1} \dots w_1$ von w bezeichnen wir mit w^R. Sei L eine Sprache. Wir definieren $L^R := \{w^R \mid w \in L\}$ als die Sprache, die alle Umkehrungen von Wörtern in L enthält.

(a) Betrachten Sie die Sprache $L_4 := \{w \in \{0,1\}^* \colon |w| \geqslant 4, w_4 = 1\}$. Geben Sie einen NEA an, der L_4^R erkennt, also die Sprache aller Umkehrungen von Wörtern aus L_4.

(b) Sei L eine reguläre Sprache. Beschreiben Sie einen NEA, der die Sprache L^R erkennt.

Wir definieren nun die Familie von Sprachen

$$L_k := \{w \in \{0,1\}^* \colon |w| \geqslant k, w_k = 1\}$$

für $k \in \mathbb{N}^+$. Die Sprache L_k besteht also aus allen Wörtern, deren k-tes Zeichen eine 1 ist.

(c) Zeigen Sie für jedes $k \in \mathbb{N}^+$ die folgende Aussage. Für alle Wörter $w, w' \in \{0,1\}^k$ mit $w \neq w'$ gilt, dass sie nicht äquivalent bezüglich der Nerode-Relation von L_k^R sind.

(d) Zeigen Sie, dass es kein Polynom p gibt, sodass für alle regulären Sprachen die folgende Aussage gilt. Ist L eine reguläre Sprache, für die ein DEA mit n Zuständen existiert, so gibt es einen DEA, der L^R erkennt und höchstens $p(n)$ viele Zustände hat.

Hinweis: Betrachten Sie die Sprachen L_k.

Problem 2: Gödelnummersprachen $2 + 2 + 2 + 3 = 9$ Punkte

Für eine gegebene Sprache K sei $L_K := \{\langle M \rangle \mid L(M) = K\}$ die Sprache der Gödelnummern aller Turingmaschinen, die genau die Wörter aus K akzeptieren.

(a) Sei $K_1 = L(10^*1)$. Beschreiben Sie eine Turingmaschine M, sodass $\langle M \rangle \in L_{K_1}$.

(b) Sei K_2 regulär. Zeigen oder widerlegen Sie: Jede Turingmaschine M mit $\langle M \rangle \in L_{K_2}$ macht bei Eingabe w höchstens $O(|w|)$ Schritte.

Kreuzen Sie an: ☐ Ich zeige die Aussage. ☐ Ich widerlege die Aussage.

(c) Sei $K_3 \notin \{\varnothing, \Sigma^*\}$ regulär. Zeigen oder widerlegen Sie: Jede Turingmaschine M mit $\langle M \rangle \in L_{K_3}$ macht bei Eingabe w mindestens $\Omega(|w|)$ Schritte.

Kreuzen Sie an: ☐ Ich zeige die Aussage. ☐ Ich widerlege die Aussage.

Sei $H = \{\langle M\rangle\#w \mid M \text{ hält auf Eingabe } w\}$ die Sprache des Halteproblems. Nach dem Satz von Rice ist L_H unentscheidbar.

(d) Beschreiben Sie eine Turingmaschine T, sodass $\langle T\rangle \in L_H$. Zeigen Sie außerdem, dass die Sprache $L' = \{\langle T\rangle\}$ entscheidbar ist. Erklären Sie *kurz*, warum dies kein Widerspruch dazu ist, dass H und L_H unentscheidbar sind.

Problem 3: Grammatiken 1+3+2+3 = 9 Punkte

Für eine Sprache L und $k \in \mathbb{N}^+$ definieren wir

$$L_k \coloneqq \{w^k \mid w \in L\}.$$

Betrachten Sie die Sprache $L' \coloneqq \mathsf{ab}^*\mathsf{a}$.

(a) Geben Sie für jedes $k \in \mathbb{N}^+$ die Sprache L'_k an.

(b) Zeigen Sie für alle $k \in \mathbb{N}^+$ mit $k \geqslant 2$, dass L'_k nicht regulär ist.

Für eine Sprache L definieren wir

$$L_\infty \coloneqq \{w^k \mid w \in L, k \in \mathbb{N}^+\}$$

(c) Geben Sie eine geeignete Sprache S an, sodass $L'_2 = S \cap L'_\infty$ und zeigen Sie mithilfe von S, dass L'_∞ nicht regulär ist.

(d) Geben Sie eine Sprache L'' an, die unendlich viele Wörter enthält und für die L''_k regulär ist für alle $k \in \mathbb{N}^+$. Bestimmen Sie für alle $k \in \mathbb{N}^+$ die Sprache L''_k und beweisen Sie, dass L''_k regulär ist.

Hinweis: Es genügt die Funktionsweise eines DEA oder NEA zu beschreiben, der die Sprache erkennt, um Regularität zu zeigen.

Problem 4: NP-Vollständigkeit $1 + 7 + 2 = 10$ Punkte

Wir betrachten folgendes Entscheidungsproblem:

RECTANGLE-PACKING
Gegeben: Ein Rechteck R mit Höhe $h \in \mathbb{N}^+$ und Breite $b \in \mathbb{N}^+$, sowie eine Menge $M = \{R_1, \ldots, R_n\}$ weiterer Rechtecke mit Höhen $h_i \in \mathbb{N}^+$ und Breiten $b_i \in \mathbb{N}^+$.
Problem: Können alle Rechtecke aus M so in R platziert werden, dass sie sich weder überlappen noch aus R hinausragen?

Hierbei gelten die folgenden Regeln für alle RECTANGLE-PACKING Instanzen:

- R sowie alle Rechtecke in M sind achsenparallel.
- R hat seine linke untere Ecke an Position $(0, 0)$.
- Die Rechtecke aus M dürfen beliebig verschoben werden. Die Rechtecke dürfen nicht rotiert werden.
- Die Ecken von Rechtecken aus M dürfen nur an ganzzahligen Koordinaten platziert werden.

In dieser Aufgabe sollen Sie zeigen, dass RECTANGLE-PACKING NP-vollständig ist. Dazu können Sie folgendes NP-vollständiges Problem benutzen:

BIN-PACKING
Gegeben: Eine Menge $U = \{x_1, \ldots, x_n\}$ und eine Gewichtsfunktion $w\colon U \to \mathbb{N}^+$. Dazu zwei natürliche Zahlen $k, B \in \mathbb{N}^+$.
Problem: Gibt es eine Aufteilung von U in k disjunkte Mengen $U_1, \ldots, U_k$, sodass für jedes U_i das Gesamtgewicht $\sum_{x \in U_i} w(x)$ höchstens B ist?

(a) Zeigen Sie, dass RECTANGLE-PACKING in NP liegt.

Hinweis: Sie dürfen ohne Beweis benutzen, dass man in Polynomialzeit überprüfen kann, ob sich zwei achsenparallele Rechtecke überlappen.

(b) Zeigen Sie, dass RECTANGLE-PACKING NP-schwer ist.

(c) Zeigen Sie, dass RECTANGLE-PACKING auch dann NP-schwer ist, wenn die Rechtecke aus M um 90° rotiert werden dürfen.

Hinweis: Sie dürfen Ihren Beweis aus (b) anpassen, oder annehmen, dass RECTANGLE-PACKING *(ohne Rotation)* NP*-schwer ist.*

Problem 5: Approximation $2 + 1 + 4 + 2 = 9$ Punkte

In dieser Aufgabe betrachten wir das Knotenfärbungsproblem Vertex Color.

> Vertex Color
> **Gegeben:** Ungerichteter Graph $G = (V, E)$.
> **Problem:** Finde das kleinste $k \in \mathbb{N}^+$, sodass es für G eine k-Färbung $c\colon V \to \{1, \dots, k\}$ gibt.

Erinnerung: Eine k-Färbung ordnet jedem Knoten eine Farbe aus $\{1, \dots, k\}$ zu, sodass je zwei benachbarte Knoten unterschiedlich gefärbt sind.

Wir betrachten nun einen Algorithmus, der schrittweise Knoten eines Graphen färbt. Dabei nennen wir eine Farbe an einem Knoten *frei*, wenn noch kein Nachbar des Knotens in dieser Farbe gefärbt wurde.

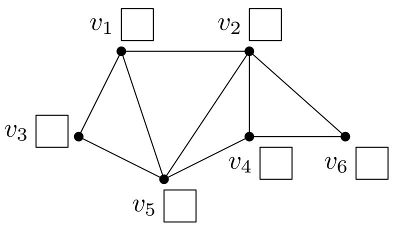
Betrachten Sie den unten beschriebenen Algorithmus $\mathcal{A}$. Er erhält als Eingabe ein Tupel (G, π) bestehend aus einem Graphen $G = (V, E)$ und einer Knotenordnung π.

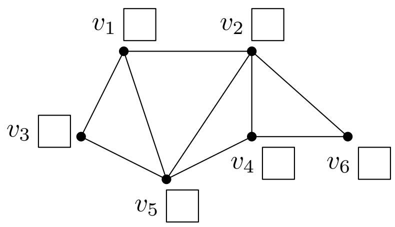
Algorithmus $\mathcal{A}$: Greedy Vertex Coloring

Input: (G, π)
Zu Beginn sind alle Knoten ungefärbt
Für *nächsten ungefärbten Knoten $v \in V$ in der durch π gegebenen Reihenfolge*
  Färbe v in der kleinsten freien Farbe
return *größte verwendete Farbe*

(a) Färben Sie die unten dargestellte Instanz, indem Sie Algorithmus $\mathcal{A}$ auf (G, π) anwenden, wobei π durch die Indizes gegeben ist ($\pi(v_i) = i$). Schreiben Sie für jeden Knoten v_i die Farbe $c(v_i) \in \mathbb{N}^+$ in die zugehörige Box. Ist diese Lösung optimal? Begründen Sie.

Falls Sie beide Kopien der Instanz bearbeiten, markieren Sie eindeutig die zu wertende Kopie.





Instanz für Teilaufgabe (a) (und eine zusätzliche Kopie)

(b) Liefert $\mathcal{A}$ für jede Eingabe (G, π) in Polynomialzeit eine k-Färbung (wobei k eine Zahl ist, die auch größer als die optimale Lösung von Vertex Color sein kann)? Begründen Sie.

(c) Zeigen Sie, dass es kein $r \in \mathbb{R}$ gibt, sodass $\mathcal{A}$ für alle Knotenordnungen π ein Approximationsalgorithmus mit relativer Güte r ist.

Hinweis: Sie können eine Familie von Bäumen $(T_i)_{i \in \mathbb{N}}$ *mit Knotenordnungen* π_i *konstruieren, für die* $\mathcal{A}$ *die Wurzel mit Farbe i färbt. Jeder Baum kann mit* 2 *Farben gefärbt werden.*

(d) Zeigen Sie, dass es für jeden Graphen G eine Knotenordnung π gibt, sodass $\mathcal{A}$ auf Eingabe (G, π) eine Lösung ausgibt, die genau so viele Farben wie eine optimale Lösung verwendet.

Problem 6: Entscheidbarkeit 3+1+1+5+2 = 12 Punkte

In dieser Aufgabe betrachten wir nur Gödelnummern von deterministischen Turingmaschinen mit Eingabealphabet $\Sigma := \{0, 1\}$.

Für Sprachen $L', L'' \subseteq \Sigma^*$ definieren wir die Sprache $L' \odot L''$ als

$$L' \odot L'' := \{w' \# w'' \mid w' \in L', w'' \in L''\}.$$

(a) Beweisen Sie die folgende Aussage. Sind L' und L'' semi-entscheidbare Sprachen, so ist die Sprache $L' \odot L''$ semi-entscheidbar.

Betrachten Sie die folgenden zwei Sprachen

$$A := \{\langle M \rangle \mid M \text{ hält auf höchstens einem Wort}\}$$
$$B := \{w \in \Sigma^* \mid \text{Es existiert TM } M\colon\ M \text{ hält auf } w\}.$$

Aus der Vorlesung wissen Sie, dass die Sprache des Halteproblems

$$H := \{\langle M \rangle \# w \mid M \text{ hält auf } w\}$$

unentscheidbar ist.

(b) Zeigen Sie, dass B entscheidbar ist.

(c) Zeigen Sie, dass für alle Gödelnummern $\langle M \rangle$ folgende Äquivalenz gilt:

$$\langle M \rangle \in A \iff \langle M \rangle \#0 \in A \odot B$$

(d) Zeigen Sie, dass $A \odot B$ nicht entscheidbar ist, indem Sie von der Sprache des Halteproblems reduzieren. Sie dürfen in dieser Aufgabe nicht den Satz von Rice verwenden.

(e) Zeigen Sie unter der Annahme, dass $A \odot B$ nicht semi-entscheidbar ist, dass A nicht semi-entscheidbar ist.