Universität Karlsruhe
Fakultät für Informatik
ILKD Prof. Dr. D. Wagner

Theoretische Informatik
WS 2003/04
14. April 2004

2. Klausur zur Vorlesung
Informatik III
Wintersemester 2003/2004

Lösung!

Beachten Sie:

- Bringen Sie den Aufkleber mit Ihrem Namen und Matrikelnummer auf diesem Deckblatt an und beschriften Sie jedes weitere Blatt mit Ihrem Namen und Ihrer Matrikelnummer.
- Schreiben Sie die Lösungen auf die Aufgabenblätter und Rückseiten. Zusätzliches Papier erhalten Sie bei Bedarf von der Aufsicht.
- Zum Bestehen der Klausur sind **20** der möglichen **61** Punkte hinreichend.
- Es sind keine Hilfsmittel zugelassen.

Aufgabe	Mögliche Punkte					Erreichte Punkte				
	a	b	c	d	Σ	a	b	c	d	Σ
1	2	2	4	4	12					
2	3	4	2	3	12					
3	3	3	3	3	12					
4	2	4	3	3	12					
5	13x1				13					
Σ					61					

Aufgabe 1: (2+2+4+4=12 Punkte)

(a) Gegeben seien die beiden folgenden Sprachen über dem Alphabet $\Sigma = \{a, b\}$:

$L_1 = \{$alle Wörter, die aa oder bb enthalten$\}$
$L_2 = \{$alle Wörter, in denen höchstens einmal aa und nie bb vorkommt$\}$

Geben Sie reguläre Ausdrücke für L_1 und L_2 an.

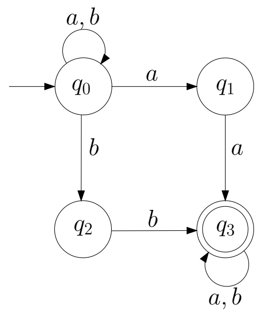
Lösung:

$$L_1 = (a \cup b)^*(aa \cup bb)(a \cup b)^*$$

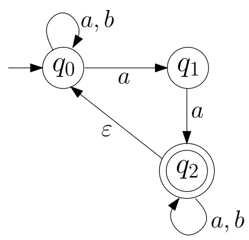
$$L_2 = (\varepsilon \cup b)(ab)^*(a \cup \varepsilon)(ab)^*(\varepsilon \cup a)$$

(b) Geben Sie den Übergangsgraphen eines endlichen Automaten an, der L_1 aus Teilaufgabe (a) erkennt.

Lösung:



(c) Gegeben sei folgender nichtdeterministischer endlicher Automat (NEA):



Konstruieren Sie mittels Potenzmengen-Konstruktion den äquivalenten deterministischen endlichen Automaten (DEA).

Lösung: Sei $A := (Q, \Sigma, \delta, s, F)$ der gegebene NEA. Potenzmengen-Konstruktion des zugehörigen DEAs $\widetilde{A} := (\widetilde{Q}, \Sigma, \widetilde{\delta}, \widetilde{s}, \widetilde{F})$:

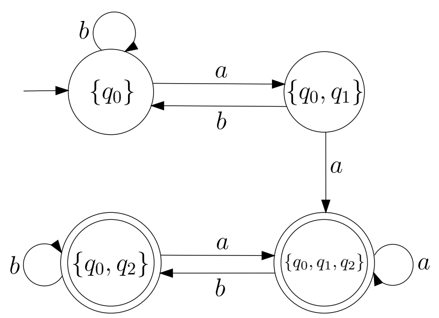
- $\widetilde{s} = E(s) = \{q_0\}$

- für $\widetilde{\delta}$ ergibt sich:

	a	b
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_0, q_1\}$	$\{q_0, q_1, q_2\}$	$\{q_0\}$
$\{q_0, q_1, q_2\}$	$\{q_0, q_1, q_2\}$	$\{q_0, q_2\}$
$\{q_0, q_2\}$	$\{q_0, q_1, q_2\}$	$\{q_0, q_2\}$

alle anderen Zustände aus 2^Q sind nicht erreichbar und werden gestrichen, somit

- $\widetilde{Q} = \{\{q_0\}, \{q_0, q_1\}, \{q_0, q_2\}, \{q_0, q_1, q_2\}\}$
- $\widetilde{F} = \{\widetilde{q} \in \widetilde{Q} | \widetilde{q} \cap F \neq \emptyset\} = \{\{q_0, q_2\}, \{q_0, q_1, q_2\}\}$.



(d) Sei $\Sigma = \{a, b\}$ und $L := \{w \in \Sigma^* | \; w$ enthält $ab\}$. Bestimmen Sie die Äquivalenzklassen der Nerode-Relation bezüglich L auf Σ^*.
Lösung: Sei R_L die Nerode-Relation bzgl. L und seien $x, y \in \Sigma^*$. Es gilt:

$$xR_Ly \iff (\forall z \in \Sigma^* : xz \in L \Leftrightarrow yz \in L)$$

Folgende Fälle können auftreten:

- x enthält ab.
  Dann gilt für alle $z \in \Sigma^* : xz \in L$
- x enthält ab nicht und x endet auf a.
  Dann gilt: $xz \in L \Leftrightarrow (z$ beginnt mit b oder z enthält $ab)$
- x enthält ab nicht und x endet nicht auf a.
  Dann gilt: $xz \in L \Leftrightarrow z$ enthält ab

Somit erhält man folgende Äquivalenzklassen:

$[ab] = \{x \in \Sigma^* | x$ enthält $ab\}$

$[a] = \{x \in \Sigma^* | x$ enthält ab nicht und endet auf $a\}$

$[\varepsilon] = [b] = \{x \in \Sigma^* | x$ enthält ab nicht und endet nicht auf $a\}$

Aufgabe 2: (3+4+2+3=12 Punkte)

(a) Betrachten Sie folgende Instanz des Post'schen Korrespondenzproblems über dem Alphabet $\{0, 1\}$:

$$K = ((100, 001), (100, 1), (10, 001), (10, 010))$$

(i) Geben Sie eine Indexfolge an, die K löst.

(ii) Modifizieren Sie K durch Entfernen *eines* Paares, so dass die Instanz nicht mehr lösbar ist. Begründen Sie, dass es für Ihre Modifikation keine Lösung gibt.

Lösung:

(i) $(2, 3, 4)$ erzeugt jeweils das Wort 1001010

(ii) Entferne das Paar $(100, 1)$. Jedes verbleibende Paar beginnt mit einem unterschiedlichen Symbol, also beginnen auch die zwei Wörter, die von einer beliebigen Indexfolge erzeugt werden, mit einem unterschiedlichen Symbol, und somit ist die modifizierte Instanz nicht mehr lösbar. Alternativ kann auch das Paar $(10, 010)$ entfernt werden. Dann endet jedes verbleibende Paar auf ein unterschiedliches Symbol. Damit erzeugt jede Indexfolge zwei verschiedene Wörter, denn sind die Wörter gleich lang, so ist das letzte Symbol unterschiedlich, und es gibt keine Lösung für die modifizierte Instanz.

(b) Sei Σ ein endliches Alphabet, $L_1 \subset \Sigma^*$ nicht entscheidbar und $\emptyset \neq L_2 \subset \Sigma^*$ entscheidbar. Sei ferner $L = \{w_1 X w_2 : w_1 \in L_1, w_2 \in L_2\}$ mit $X \notin \Sigma$.

Ist L entscheidbar? Begründen Sie Ihre Antwort!

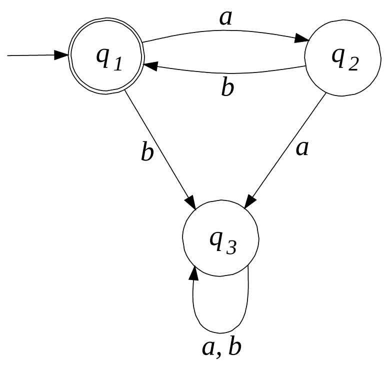
Lösung: L ist nicht entscheidbar. Angenommen, L sei entscheidbar. Dann existiert eine Turingmaschine $\mathcal{M}$, die auf alle Eingaben hält und genau die Sprache L akzeptiert. Wir konstruieren nun eine Turingmaschine $\mathcal{M}_1$, die auf allen Eingaben $w \in \Sigma^*$ hält und genau die Sprache L_1 akzeptiert. Sei dazu w_2 ein Wort aus L_2. Die Turingmaschine $\mathcal{M}_1$ geht an das Ende der Eingabe (also solange nach rechts, bis ein $\sqcup$ gelesen wird), schreibt ein X und danach das Wort w_2. Danach geht $\mathcal{M}_1$ wieder an den Anfang der Eingabe (also solange nach links, bis ein $\sqcup$ gelesen wird), und verhält sich dann genau wie die Turingmaschine $\mathcal{M}$. Sei w' eine Eingabe für $\mathcal{M}_1$. Ist $w' \in L_1$, so wird das Wort akzeptiert, denn $w' X w_2 \in L$. Falls aber $w' \notin L_1$, so ist auch $w' X w_2 \notin L$, also wird w' nicht akzeptiert, aber $\mathcal{M}_1$ hält, weil $\mathcal{M}$ auf allen Eingaben hält. Dies ist ein Widerspruch zur Annahme, denn L_1 ist nach Voraussetzung nicht entscheidbar.

(c) Gegeben sei eine Turingmaschine M mit folgender Eigenschaft: Wenn M ein Wort akzeptiert, dann geschieht das in weniger als 1000 Schritten. Kreuzen Sie für folgende Aussagen an, ob diese wahr oder falsch sind.

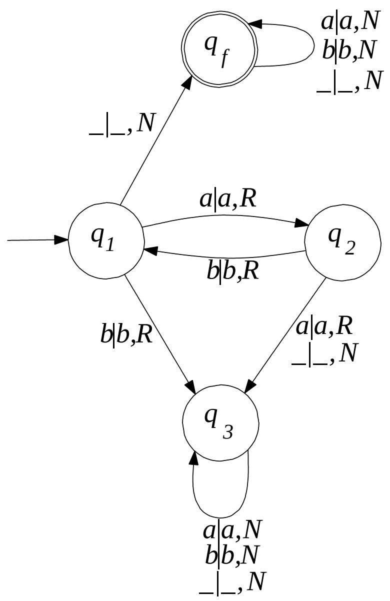
Hinweis: Für jede richtige Antwort gibt es einen Punkt, für jede falsche Antwort wird ein Punkt abgezogen. Es wird keine negative Gesamtpunktzahl für diese Teilaufgabe geben.

Aussage		Wahr	Falsch
$L(M)$ ist entscheidbar.	*Lösung:*	X	
$L(M)$ ist notwendigerweise endlich.	*Lösung:*		X

(d) Geben Sie graphisch eine deterministische Turingmaschine nach Definition aus der Vorlesung an, die auf allen Eingaben über dem Alphabet $\{a, b\}$ hält und dieselbe Sprache akzeptiert wie der folgende deterministische endliche Automat:



Lösung:



Aufgabe 3: (3+3+3+3 = 12 Punkte)

[Hinweis: Für in dieser Aufgabe vorkommende Probleme sind jeweils am Ende des Aufgabentextes Definitionen angegeben.]

(a) Zeigen Sie: LONGEST CYCLE ist $\mathcal{NP}$-vollständig.

Problem LONGEST CYCLE (LC)

Gegeben: Ungerichteter Graph $G = (V, E)$, Konstante K

Frage: Gibt es einen einfachen Kreis in G, der mindestens die Länge K hat?

Problem HAMILTONKREIS (HK)

Gegeben: Ungerichteter Graph $G = (V, E)$

Frage: Gibt es einen einfachen Kreis in G, der jeden Knoten genau einmal enthält?

[Ein einfacher Kreis der Länge ℓ ist eine Folge von ℓ verschiedenen Knoten, so dass jeweils eine Kante zwischen dem $(i+1)$-ten und dem i-ten, sowie zwischen dem ersten und letzten Knoten besteht.

In der Vorlesung wurde gezeigt, dass HAMILTONKREIS $\mathcal{NP}$-vollständig ist.]

Lösung:

1. LONGEST CYCLE $\in \mathcal{NP}$:
   Das Orakel der nichtdeterministischen TM gibt eine Folge von Knoten als Lösungsvorschlag vor. Dieser kann in polynomialer Zeit verifiziert werden, indem nachgeschaut wird, ob er aus mindestens K verschiedenen Knoten besteht und ob zwischen je zwei aufeinanderfolgenden Knoten sowie zwischen dem letzten und ersten Knoten eine Kante existiert.
2. HAMILTONKREIS $\propto$ LONGEST CYCLE:
   Eine Instanz von Hamiltonkreis mit Graph $G = (V, E)$ wird auf die Instanz $I = (G, |V|)$ von LONGEST CYCLE abgebildet. Eine einfacher Kreis der Länge $K = |V|$ ist nämlich genau ein Hamiltonkreis (da er damit alle Knoten enthält). Die Transformation funktioniert in linearer Zeit, da nur die Konstante $|V|$ zur Eingabe hinzugefügt werden muss.

(b) Formulieren Sie das Erfüllbarkeitsproblem SAT als 0/1-ILP. Geben Sie also an, wie für eine beliebige Instanz von SAT mit den Klauseln $K_1, \ldots, K_k$ über den Variablen $V_1, \ldots, V_l$ eine äquivalente Instanz von 0/1-ILP konstruiert werden kann.

Problem 0/1-INTEGER-PROGRAMMING (0/1-ILP)

Gegeben: Eine $(m \times n)$-Matrix A ganzer Zahlen und ein Spaltenvektor b von m ganzen Zahlen.

Frage: Gibt es einen Lösungsvektor x mit n Einträgen nur 0 oder 1 so, dass $Ax \geq b$?

Lösung: Die gesuchte Matrix A hat $k+2l$ Zeilen und $2l$ Spalten. Für jede Variable gibt es eine Spalte (V_i^-) für die negierte und eine (V_i^+) für die nicht negierte Form. Es gibt zunächst für jede Variable eine Zeile der Form $V_i^+ + V_i^- \geq 1$ und eine Zeile der Form $-V_i^+ - V_i^- \geq -1$, um zu gewährleisten, dass jede Variable entweder wahr oder falsch ist. Außerdem gibt es für jede Klausel $u_{i_1} \vee \cdots \vee u_{i_t}$ eine Zeile der Form $u_{i_1} + \cdots + u_{i_t} \geq 1$, wobei die u_{i_j} verneinte oder nicht verneinte Variablen sind.

(c) Zeigen Sie: Die Klasse $\mathcal{DTAPE}(n^2)$ ist unter Durchschnittsbildung abgeschlossen, d.h. mit $L_1, L_2 \in \mathcal{DTAPE}(n^2)$ ist auch $L_1 \cap L_2 \in \mathcal{DTAPE}(n^2)$.

Lösung: Seien $L_1, L_2 \in \mathcal{DTAPE}(n^2)$. Dann gibt es TMn M_1 bzw. M_2, die L_1 bzw. L_2 mit Platzbedarf n^2 bei Eingabelänge n akzeptieren. Wir konstruieren eine TM M, die zunächst M_1 auf der Eingabe simuliert und, falls M_1 akzeptiert, die Eingabe wieder herstellt und anschließend M_2 simuliert. Wenn auch M_2 akzeptiert, dann akzeptiert auch M. Somit akzeptiert M genau dann, wenn M_1 und M_2 akzeptieren. M akzeptiert also den Durchschnitt der beiden Sprachen. Der Speicherplatzbedarf von M ist nicht größer als der von M_1 und M_2. (M muss sich nur zusätzlich die ursprüngliche Eingabe merken. Dies kann etwa durch Erweiterung des Alphabets geschehen.)

(d) Sei Π ein Minimierungsproblem. Für jede gerade Zahl ℓ gebe es einen Algorithmus A_ℓ für Π mit folgenden Eigenschaften:

1. Die Eingabe für A_ℓ ist eine Instanz I von Π.
2. Die Ausgabe ist eine Zahl $A_\ell(I)$ mit

$$1 \leq \frac{A_\ell(I)}{\mathrm{OPT}(I)} \leq 1 + \frac{1/2}{1 + \ell/2} \quad ,$$

   wobei $\mathrm{OPT}(I)$ der Wert einer optimalen Lösung von I sei.
3. Die Laufzeit von A_ℓ sei in $\mathcal{O}(2^\ell + n)$.

Geben Sie an, welche der folgenden Aussagen aus der Existenz der A_ℓ und unter der Annahme $\mathcal{P} \neq \mathcal{NP}$ folgen. Begründen Sie ihre Antwort (jeweils zwei Sätze genügen).

(i) Es gibt einen Approximationsalgorithmus mit relativer Gütegarantie 2 für Π.

(ii) Es gibt ein PAS für Π.

(iii) Es gibt ein FPAS für Π.

Lösung:

(i) Ja. Für $\ell = 2$ ist A ein 1/4-approximativer Algorithmus. Die Laufzeit ist linear in der Eingabegröße, da ℓ eine Konstante ist.

(ii) Ja. Für jedes ε ist A_ℓ mit $\ell := 1/\varepsilon$ ein ε-approximativer Algorithmus. Die Laufzeit ist polynomial in der Eingabegröße.

(iii) Nein. Die Laufzeit der in Punkt (i) beschriebenen Algorithmenfamilie ist exponentiell in ℓ und also auch in $1/\varepsilon$.

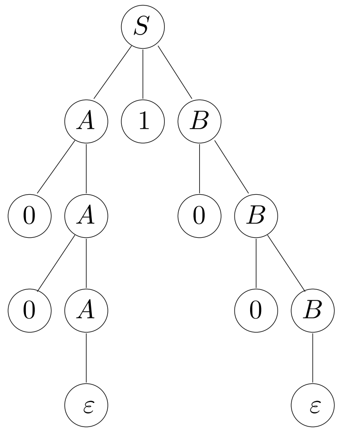
Aufgabe 4: (2+4+3+3=12 Punkte)

Sei eine Grammatik G über dem Alphabet $\{0, 1\}$, der Variablenmenge $\{S, A, B\}$, dem Startsymbol S und den folgenden Regeln gegeben:

$$S \to A1B, \quad A \to 0A \mid \varepsilon, \quad B \to 0B \mid 1B \mid \varepsilon.$$

(a) Geben Sie den Syntaxbaum für eine Ableitung des Wortes 00100 gemäß G an.

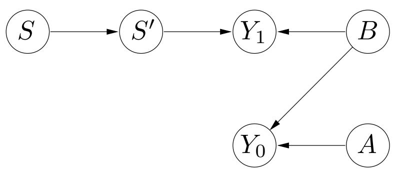
Lösung:



(b) Bringen Sie G durch eine systematische Konstruktion in Chomsky-Normalform (die einzelnen Schritte müssen dabei klar erkennbar sein!).

Lösung:

- Schritt 1 und 2 (Einführen von Variablen für Terminale und Splitten zu langer Regeln):
  $S \longrightarrow AS', \quad S' \longrightarrow Y_1B, \quad A \longrightarrow Y_0A \mid \varepsilon, \quad B \longrightarrow Y_0B \mid Y_1B \mid \varepsilon, \quad Y_0 \longrightarrow 0, \quad Y_1 \longrightarrow 1$
- Schritt 3 (Entfernen von ε-Regeln):
  Variablen, von denen ε abgeleitet werden kann: $V' = \{A, B\}$
  $S \longrightarrow AS' \mid S', \quad S' \longrightarrow Y_1B \mid Y_1, \quad A \longrightarrow Y_0A \mid Y_0, \quad B \longrightarrow Y_0B \mid Y_1B \mid Y_0 \mid Y_1,$
  $Y_0 \longrightarrow 0, \quad Y_1 \longrightarrow 1$
- Schritt 4 (Entfernen von Kettenregeln):
  Abhängigkeitsgraph:



$S \longrightarrow AS' \mid Y_1B \mid 1, \quad S' \longrightarrow Y_1B \mid 1, \quad A \longrightarrow Y_0A \mid 0, \quad B \longrightarrow Y_0B \mid Y_1B \mid 0 \mid 1,$
$Y_0 \longrightarrow 0, \quad Y_1 \longrightarrow 1$

(c) Zeigen Sie mit Hilfe des Lemmas von Ogden, dass die Sprache

$$L := \{a^i b^j a^k \mid j = \max\{i, k\}\}$$

nicht kontextfrei ist.
(Hinweis: Sie können dazu das Wort $z = a^n b^n c^n$ mit der Markierung b^n wählen.)

Lösung: [Das Lemma von Ogden besagt: Wenn eine Sprache L kontextfrei ist, dann gibt es eine natürliche Zahl n so, dass für alle Wörter z aus L der Länge mindestens n gilt: Wenn wir in z mindestens n Buchstaben markieren, so gibt es eine Zerlegung $z = uvwxy$ so, dass das Teilwort vx mindestens einen und das Teilwort vwx höchstens n markierte Buchstaben enthält und für alle natürlichen Zahlen i (einschließlich der Null) das gepumpte Wort uv^iwx^iy in L enthalten ist.]

Um nun die Nicht-Kontextfreiheit der Sprache L zu zeigen, wähle zu gegebenem n das Wort $z = a^n b^n a^n$ und markiere das Teilwort b^n. *Jede* Zerlegung $z = uvwxy$ mit den Eigenschaften markiert$(vx) \geq 1$ und markiert$(vwx) \leq n$ hat eine der folgenden Gestalten:

(i) v oder x enthält sowohl den Buchstaben a als auch b.

(ii) Entweder ist vwx ganz in b^n enthalten oder v (bzw. x) ist aus a^n und x (bzw. v) aus b^n.

Gib für jeden der Fälle ein gepumptes Wort uv^iwx^iy an, das *nicht* in der Sprache liegt:

(i) Wähle $i := 2$. Dann enthält das gepumpte Wort (mindestens) zwei b-Sequenzen.

(ii) Wähle $i := 0$. Dann gibt es im gepumpten Wort (mindestens) einen a-Block mit echt größerer Kardinalität als die Anzahl der bs.

Also ist L nicht kontextfrei.

(d) Widerlegen Sie durch ein Gegenbeispiel:

Unendliche Vereinigungen kontextfreier Sprachen sind kontextfrei. (Mit anderen Worten: Falls für alle $i \in I\!N_0$ die Sprachen L_i kontextfrei sind, dann ist auch $L := \bigcup_{i=0}^{\infty} L_i$ kontextfrei.)

Lösung: Die Sprachen $L_0 := \emptyset$, $L_1 := \{a^1b^1c^1\}$, $L_2 := \{a^2b^2c^2\}$, $\ldots$ sind regulär und somit also kontextfrei, die abzählbare Vereinigung dieser Sprachen $\bigcup_{i=0}^{\infty} L_i = \{a^nb^nc^n \mid n \in I\!N\}$ hingegen ist nicht kontextfrei (vgl. Vorlesung).

Aufgabe 5:

(13x1=13 Punkte)

Kreuzen Sie für folgende Aussagen an, ob diese wahr oder falsch sind.
Hinweis: Für jede richtige Antwort gibt es einen Punkt, für jede falsche Antwort wird ein Punkt abgezogen. Es wird keine negative Gesamtpunktzahl für diese Aufgabe geben.

Seien L_1, L_2 reguläre Sprachen.
Dann ist auch $L_1 \setminus L_2 = \{w \in L_1 | \ w \notin L_2\}$ regulär.

☐ Wahr ☐ Falsch

Lösung: ☒ Wahr ☐ Falsch

Die regulären Ausdrücke $a(b^* \cup c^*)$ und $a(b \cup c)^*$ sind äquivalent.

☐ Wahr ☐ Falsch

Lösung: ☐ Wahr ☒ Falsch

Das Komplement der universellen Sprache ist nicht semientscheidbar.

☐ Wahr ☐ Falsch

Lösung: ☒ Wahr ☐ Falsch

Seien L_1 und L_2 zwei semientscheidbare Sprachen. Dann ist auch
$L_1 \setminus L_2 = \{w \in L_1 | \ w \notin L_2\}$ semientscheidbar.

☐ Wahr ☐ Falsch

Lösung: ☐ Wahr ☒ Falsch

Sei $L \subset \{0,1\}^*$ nicht entscheidbar. Dann gilt: Der Index der Neroderelation zu L ist unendlich.

☐ Wahr ☐ Falsch

Lösung: ☒ Wahr ☐ Falsch

Aus 3SAT $\in \mathcal{P}$ folgt 2SAT $\in \mathcal{NPC}$.

☐ Wahr ☐ Falsch

Lösung: ☒ Wahr ☐ Falsch

Falls es einen Approximationsalgorithmus für CLIQUE mit absoluter Gütegarantie gibt, so gilt $P = NP$

☐ Wahr ☐ Falsch

Lösung: ☒ Wahr ☐ Falsch

Es gibt ein Entscheidungsproblem $\Pi \in \mathcal{NP}$, für das es keine polynomiale Transformation $\Pi \propto SAT$ gibt.

☐ Wahr ☐ Falsch

Lösung: ☐ Wahr ☒ Falsch

Sei k eine Konstante. Die Sprache $\mathrm{VC}_k := \{G|\ G = (V, E)$ ist ein Graph und hat eine Knotenüberdeckung $V' \subset V$ mit $|V| \leq k\}$ ist in $\mathcal{P}$.

☐ Wahr ☐ Falsch

Lösung:

☒ Wahr ☐ Falsch

Zu jeder entscheidbaren Sprache L existiert eine Chomsky-Typ-0-Grammatik, die L erzeugt.

☐ Wahr ☐ Falsch

Lösung:

☒ Wahr ☐ Falsch

Jede Sprache der Form $\{x_1^n x_2^n \ldots x_k^n \colon n \in \mathbb{N}\}$ ist kontextfrei. (Dabei sei $k \geq 1$ und die x_i jeweils Buchstaben aus einem endlichen Alphabet mit $x_i \neq x_j$ für $i \neq j$.)

☐ Wahr ☐ Falsch

Lösung:

☐ Wahr ☒ Falsch

Zu jedem nichtdeterministischen Kellerautomat gibt es einen deterministischen Kellerautomaten, der dieselbe Sprache akzeptiert.

☐ Wahr ☐ Falsch

Lösung:

☐ Wahr ☒ Falsch

Die Grammatik, die nur aus der Regel $S \to \varepsilon$ besteht, erzeugt dieselbe Sprache wie eine Grammatik ohne Regeln.

☐ Wahr ☐ Falsch

Lösung:

☐ Wahr ☒ Falsch