

Die Anzahl der Aufgaben, das Punkteschema, die Themenschwerpunkte, etc. können in der echten Klausur unterschiedlich sein!

Uni Osnabrück / Theoretische Informatik

Prof. Dr. M. Chimani

Informatik D: Einführung in die Theoretische Informatik Klausur — SoSe 2013 / PROBEKLAUSUR

Gruppe: Lollipop

Unbedingt ausfüllen

Matrikelnummer:

Nachname:

Vorname:

Studiengang:

Identifikator:

Identifikator?

Beliebiges Wort zur Identifikation im anonymen Notenaushang.

Grundregeln

- Die Bearbeitungszeit der Klausur beträgt **120 Minuten**.
- Sie schreiben diese Klausur **vorbehaltlich** der Erfüllung der **Zulassungsvoraussetzung**. Das heißt: Wir werden Ihre Zulassung vor Korrektur prüfen; die Tatsache, dass Sie die Klausur mitschreiben, bedeutet keine implizite Zulassung.
- Es sind **keine Unterlagen** und auch **keine** anderen **Hilfsmittel** erlaubt.
- Benutzen sie nur dokumentenechten (blauen oder zur Not schwarzen) **Kugelschreiber!** Bleistiftlösungen werden nicht gewertet!
- Es zählt die Antwort, die sich im dafür vorgesehenen Kästchen befindet! Soll eine andere Antwort gewertet werden, so ist diese **eindeutig** zu kennzeichnen!
- Jegliches Schummeln, und auch der Versuch desselben, führt zum Ausschluss von der Klausur und einer Bewertung mit **5,0**.

Wird vom Korrektor/Prüfer ausgefüllt

Aufgabe	1	2	3	4	5	6	7	8	9	10	Σ
Punkte (max)	6	10	16	12	20	14	18	12	18	10	136
Punkte (erreicht)											

Punkte	0..67	68..74	75..81	82..88	89..95	96..102	103..108	109..115	116..122	123..129	130..136
Note	5,0	4,0	3,7	3,3	3,0	2,7	2,3	2,0	1,7	1,3	1,0

Note:

Aufgabe 1: Sprachen vs. Automaten

(6 Punkte)

Geben Sie zu den Automaten die jeweils größten Sprachklasse an, die damit beschrieben werden kann.

Automaten	Sprachklasse
Nicht-deterministischer Endlicher Automat	
Turingmaschine	
Determ. Kellerautomat, Akz. d. Endzust.	

Aufgabe 2: Sprachen klassifizieren

(10 Punkte)

Zu welcher Sprachklasse gehören die folgenden Sprachen? Kreuzen Sie dabei **alle** korrekten Antworten an. (Falsche Kreuzchen führen zu Punkteabzug innerhalb der Teilaufgabe.)

	regulär	determ. kontextfrei	kontextfrei	kontextsensitiv	rek. aufzählbar
$\{ww^R \mid w \in \{a, b\}^*\}$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$\{a, b\}^* \cup \{cc, dd\}$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$\{a^n b^m \mid n \geq m \geq 0\}$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
$\{w \mid w \text{ ist ein korrektes Java-Programm}\}$	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Aufgabe 3: Sprachbeschreibungen erstellen

(16 Punkte)

(a) Regulärer Ausdruck

(4 Punkte)

Erstellen Sie eine reguläre Grammatik (gemäß den strikten Regeln), die zu folgendem regulären Ausdruck äquivalent ist:

$$(a|b)^+(c(a|\varepsilon)|d^*)$$

(b) Kellerautomaten

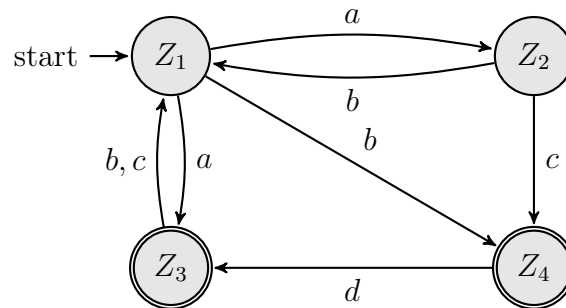
(12 Punkte)

Erstellen Sie einen Kellerautomaten, der genau die Sprache $\{(ab)^i c^{3i} \mid i \geq 0\}$ akzeptiert.

Aufgabe 4: Sprachbeschreibungen verwandeln

(12 Punkte)

Wandeln Sie—gemäß dem Vorgehen aus der Vorlesung!— den folgenden nicht-deterministischen endlichen Automaten in einen deterministischen um.



Aufgabe 5: Pumping Lemma

(20 Punkte)

(a) Quiz

(8 Punkte)

Markieren Sie die Richtigkeit der folgenden Aussagen bzgl. dem Pumping Lemma für kontextfreie Sprachen (PL-KF):

(Achtung: Pro Frage gibt es +2/0/-2 Punkte bei einer richtigen/keinen/falschen Antwort!)

Jede kontextfreie Sprache erfüllt das PL-KF: ja, nein.

Es gibt nicht-kontextfreie Sprache, die das PL-KF erfüllen: ja, nein.

Es gibt *deterministische* kontextfreie Sprache, die das PL-KF erfüllen: ja, nein.

Beim PL-KF wird ein Wort in 3, 4, 5, 7 Teilworte zerlegt.

(b) Anwendung

(12 Punkte)

Zeigen Sie, dass die Sprache $\{a^i b^i c^{2i} \mid i \in \mathbb{N}\}$ nicht kontextfrei ist. Vergessen Sie nicht, Begründungen für Ihrer Beweisschritte anzugeben.

Aufgabe 6: Rechnende Turingmaschine

(14 Punkte)

Geben Sie eine Turingmaschine an, die als Angabe eine binär kodierte Zahl $\alpha \in \mathbb{N}$ enthält und folgendes Code-Fragment berechnet:

```
if( $\alpha < 4$ ) return 0; else return  $\alpha$ ;
```

Aufgabe 7: Sprachen und Berechenbarkeit

(18 Punkte)

Markieren Sie die Richtigkeit der folgenden Aussagen.

(Achtung: Pro Frage gibt es +2/0/-2 Punkte bei einer richtigen/keinen/falschen Antwort!)

ja nein

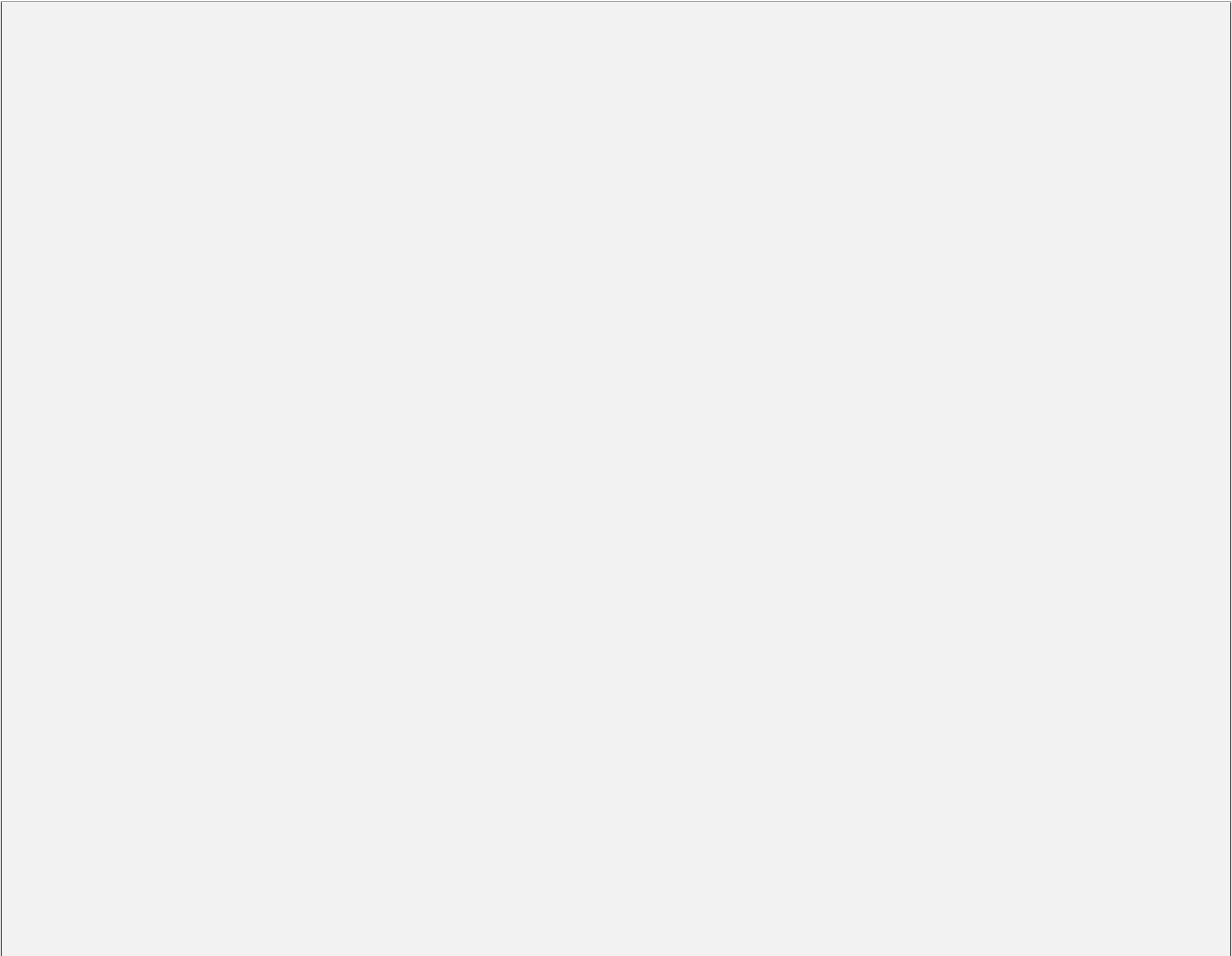
- | | | |
|--------------------------|--------------------------|---|
| <input type="checkbox"/> | <input type="checkbox"/> | Das Wortproblem kann auf kontextfreien Sprachen in polynomieller Zeit gelöst werden. |
| <input type="checkbox"/> | <input type="checkbox"/> | Die Vereinigung zweier deterministisch kontextfreier Sprachen ist wieder deterministisch kontextfrei. |
| <input type="checkbox"/> | <input type="checkbox"/> | Ein semi-entscheidbares Problem lässt sich immer als Wortproblem über eine entsprechende rekursiv aufzählbare Sprache auffassen. |
| <input type="checkbox"/> | <input type="checkbox"/> | Ein entscheidbares Problem lässt sich immer als Wortproblem über eine entsprechende rekursiv aufzählbare Sprache auffassen. |
| <input type="checkbox"/> | <input type="checkbox"/> | Das Wortproblem auf einer rekursiv aufzählbaren Sprache ist entscheidbar. |
| <input type="checkbox"/> | <input type="checkbox"/> | Die Frage nach einer periodischen Wang Kachelung ist entscheidbar. |
| <input type="checkbox"/> | <input type="checkbox"/> | Kolmogorov Komplexität, eingeschränkt auf eine Beschreibung mit einer Turing-vollständigen Programmiersprache, ist entscheidbar. |
| <input type="checkbox"/> | <input type="checkbox"/> | Wenn man die Anzahl der Schritte eines n -BusyBeavers kennt, ist die Anzahl der Schritte eines $(n - 1)$ -BusyBeaver berechenbar. |
| <input type="checkbox"/> | <input type="checkbox"/> | Es ist entscheidbar, ob zwei gegebene kontextfreie Grammatiken die selbe Sprache beschreiben. |

Aufgabe 8: GOTO

(12 Punkte)

Geben Sie ein GOTO Programm an, dass $x_3 := 2x_1 + x_2$ berechnet. Benutzen Sie dabei nur die auf die folgenden Anweisungen eingeschränkte GOTO-Sprache:

- $x_i := x_j \pm c$, für Konstanten $c \in \mathbb{N}$
- goto L_i
- if($x_i \neq 0$) goto L_i
- halt

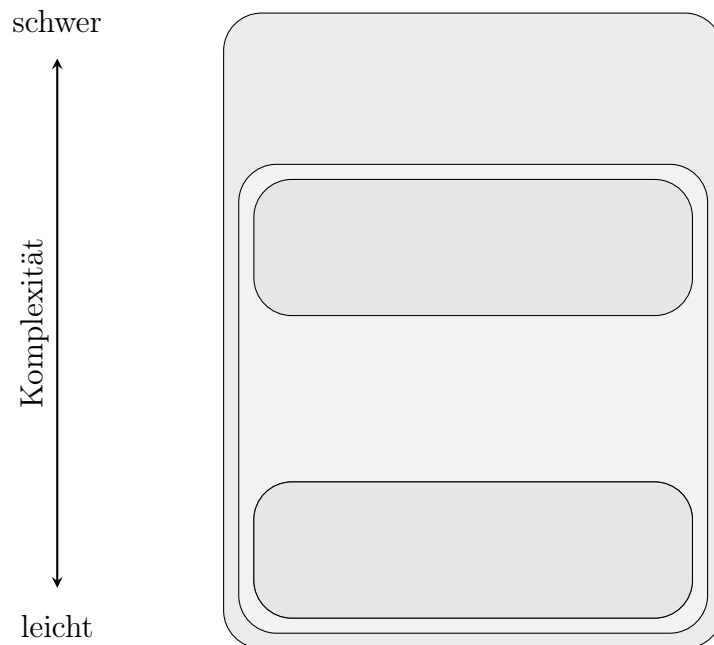


Aufgabe 9: Komplexitätsfragen

(18 Punkte)

Nehmen Sie innerhalb dieser Aufgabe an, dass $P \neq NP$.

Beschriften Sie das folgende Mengendiagramm mit „ P “, „ NP “, „ NP -vollständig“ und „ $PSPACE$ “. Markieren sie auch, wo sich NP -schwere Probleme befinden.



Nennen Sie jeweils zwei Probleme die in der gegebenen Komplexitätsklasse liegen:

P : ,

$NP \setminus P$: ,

Aufgabe 10: Reduktionen

(10 Punkte)

Seien \mathcal{Y}_i , $1 \leq i \leq 5$, NP -vollständige Entscheidungsprobleme. Wir möchten zeigen, dass ein Problem \mathcal{X} NP -vollständig ist, und können dabei die obigen Probleme benutzen.

Was sind die Beweisschritte, die wir durchführen müssen?

Angenommen, die Probleme \mathcal{Y}_i wären nur NP -schwer. Können wir den obigen Beweis genauso durchführen? ja, nein.

Viel Erfolg bei der Vorbereitung!