Universität Osnabrück / FB6 / Theoretische Informatik — Prof. Dr. M. Chimani

Informatik D: Einführung in die Theoretische Informatik

Klausur — SoSe 2018 — 9. Juli 2018

Haupttermin, Prüfungsnr. 1007049 — Gruppe: Erdbeere / Haselnuss

Unbedingt ausfüllen

Matrikelnummer

Studiengang/Abschluss

Fachsemester

Nachname

Vorname

Unterschrift

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	Σ
Punkte (max)	14	14	14	8	16	8	8	18	20	**120**
Punkte (erreicht)										

Punkte	0..59	60..66	67..73	74..77	78..82	83..87	88..91	92..96	97..100	101..107	108..120
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: Sprachgrundlagen **(14 Punkte)**

(a) Faktenwissen (6 Punkte)

Welche Form hat eine Grammatik, die eine kontextsensitive Sprache beschreibt?

Von welchem Maschinenmodell werden genau die kontextfreien Sprachen beschrieben?

Welche Sprachfamilie wird genau durch einen NDLBA beschrieben?

(b) Zuordnung von Sprachen (8 Punkte)

Zu welcher der *fünf* in der Vorlesung besprochenen Sprachfamilien innerhalb der Chomsky-Hierarchie gehören die folgenden Sprachen? Geben Sie dabei die *kleinste* Sprachfamilie an, die gerade mächtig genug ist, die entsprechende Sprache zu erkennen.

$S \to cA \mid aA, \quad A \to cA \mid \varepsilon$

$L_1 \cup L_2$, wobei L_1, L_2 kontextfreie Sprachen sind

$\{a^j c^i b^j \mid i, j \in \mathbb{N}, i \geq 2\}$

$\left(\{a,b\}^* \cup \{ac, \varepsilon\}^+\right) \cap \overline{\{a^i c^i \mid i \geq 1\}}$ über $\Sigma = \{a, b, c\}$

Aufgabe 2: Äquivalenz der Beschreibungen regulärer Sprachen (14 Punkte)

(a) Anwendung (4 Punkte)

Geben Sie einen NDEA an, der genau die Sprache über dem Alphabet $\{a, b, c\}$ akzeptiert, die von dem folgenden regulären Ausdruck erzeugt wird: $a(cb|ca)^+cc$

(b) Algorithmus (10 Punkte)

Beschreiben Sie das Verfahren aus der Vorlesung, das einen regulären Ausdruck in einen äquivalenten NDEA überführt.

Aufgabe 3: Pumping Lemma **(14 Punkte)**

(a) Definition (4 Punkte)

Wie lautet das Pumping Lemma für kontextfreie Sprachen?

Sei L eine kontextfreie Sprache. Dann...

... $z = uvwxy$ mit den Eigenschaften

(1) , (2) und (3) .

(b) Anwendung (10 Punkte)

Beweisen Sie, dass $L := \{\alpha 2^{|\alpha|/2} \alpha \mid \alpha \in \{0,1\}^*\}$ nicht kontextfrei ist.

Aufgabe 4: Rechnende Turingmaschine (8 Punkte)

Problem: ZahlKonvertierung
Gegeben: Die unäre Darstellung einer Zahl $\alpha \in \mathbb{N}$.
Gesucht: Die binäre Darstellung von α.

Konstruieren Sie eine Turingmaschine, die das Problem ZahlKonvertierung löst. Dazu dürfen Sie einen LBA $\mathcal{A}$, der zu einer gegebenen Binärzahl 1 addiert, als Black Box verwenden.

Aufgabe 5: Berechenbarkeit (16 Punkte)

GOTO-Programme sind Turing-vollständig, selbst wenn alle Vergleichsoperationen nur mit 0 vergleichen und alle Variablen initial auf 0 gesetzt sind. Sei GOTO_{17} die Teilmenge aller dieser GOTO-Programme mit der Einschränkung, dass die größte Konstante, die in einer Operation auftritt, 17 ist.

(a) Vollständigkeit (2 Punkte)

Warum lässt sich jedes GOTO-Programm durch ein GOTO_{17}-Programm simulieren?

(b) Halteproblem (8 Punkte)

Sei GOTO_{17}^{c} die Teilmenge aller GOTO_{17}-Programme, deren Variablen höchstens Wert c annehmen können. Warum ist das Halteproblem für GOTO_{17}^{c}-Programme entscheidbar?

(c) Fleißige GOTO-Programme (6 Punkte)

Sei $\text{GOTO}_{17}(n)$ die Teilmenge aller GOTO_{17}-Programme, die aus höchstens n Anweisungen bestehen.

Die Funktion $\vartheta(n)$ gibt den größten Wert an, der bei Aufruf eines terminierenden Programms aus $\text{GOTO}_{17}(n)$ einer der Variablen zugewiesen wird. Beweisen oder widerlegen Sie die folgende Aussage:

Die Funktion $\vartheta(n)$ ist berechenbar.

Aufgabe 6: Zusammenhänge in Berechenbarkeit **(8 Punkte)**

(a) Ankreuzen (5 Punkte)

Bewertung: Pro Aussage gibt es 1/0/-1 Punkte bei einer richtigen/keinen/falschen Antwort! Es gibt jedoch keine negativen Punkte für diese Teilaufgabe.

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.

Wahr	Falsch	
☐	☐	Die Frage nach der Existenz einer periodischen Lösung des Wang-Parketts ist unentscheidbar.
☐	☐	Ist $L \subseteq \{0,1\}^*$ semi-entscheidbar oder co-semi-entscheidbar, dann ist L entscheidbar.
☐	☐	Ist $L \subseteq \{0,1\}^*$ unentscheidbar, dann ist L nicht rekursiv aufzählbar.
☐	☐	Das Wortproblem von Turingmaschinen ist nicht co-semi-entscheidbar.
☐	☐	Die folgende Sprache ist unentscheidbar: $\{\mathbb{W}(M) \mid M \text{ ist eine rechnende TM und } M(x) = 4x^3 + 2x^2 + 5x + 1 \text{ für alle } x \in \mathbb{N}\}$.

(b) Beweis (3 Punkte)

Beweisen Sie eine der **oben als wahr angekreuzten** Aussagen Ihrer Wahl. Kennzeichnen Sie die gewählte Aussage.

Aufgabe 7: Zusammenhänge Komplexitätstheorie (8 Punkte)

(a) Ankreuzen (5 Punkte)

Bewertung: Pro Aussage gibt es 1/0/-1 Punkte bei einer richtigen/keinen/falschen Antwort! Es gibt jedoch keine negativen Punkte für diese Teilaufgabe.

Kreuzen Sie an, ob die folgenden Aussagen wahr oder falsch sind.
Hinweis: NPC ist die Klasse der NP-vollständigen Probleme.

Wahr	Falsch	
☐	☐	$\mathsf{P} = \mathsf{NP} \implies \mathsf{NP} = \mathsf{Co\text{-}NP}$.
☐	☐	$\mathsf{P} \neq \mathsf{NP} \implies \mathsf{NP} = \mathsf{P} \cup \mathsf{NPC}$.
☐	☐	Nehmen Sie an, dass $\mathsf{P} \neq \mathsf{NP}$. Es gibt Entscheidungsprobleme $\mathcal{X}, \mathcal{Y}$, sodass $\mathcal{X}$ polynomiell auf $\mathcal{Y}$ reduzierbar ist, aber $\mathcal{Y}$ nicht auf $\mathcal{X}$.
☐	☐	Wenn 3SAT polynomiell auf TAUTOLOGIE reduzierbar ist, dann ist $\mathsf{NP} = \mathsf{Co\text{-}NP}$.
☐	☐	Wenn $\mathsf{P} \neq \mathsf{NP}$ und das Wortproblem für $L \subseteq \{0,1\}^*$ in exponentieller Zeit entschieden werden kann, dann ist L NP-schwer.

(b) Beweis (3 Punkte)

Beweisen Sie eine der **oben als wahr angekreuzten** Aussagen Ihrer Wahl. Kennzeichnen Sie die gewählte Aussage.

Aufgabe 8: Randomisierte Algorithmen (18 Punkte)

(a) Definitionen (4 Punkte)

Wie lautet die Definition von ***RP***?

Ein Entscheidungsproblem liegt in ***RP***, wenn ein

{ ☐ Monte-Carlo ☐ Atlantic-City ☐ Las-Vegas }-Algorithmus existiert, der

— Ja-Instanzen mit Wahrscheinlichkeit ____ und

— Nein-Instanzen mit Wahrscheinlichkeit ____ korrekt erkennt.

(b) Algorithmus (8 Punkte)

In der Genetik gibt es vier verschiedenen Basen: Adenin (A), Cytosin (C), Guanin (G) und Thymin (T). Dabei bilden jeweils Adenin und Thymin sowie Cytosin und Guanin ein sogenanntes Basenpaar. Ein *DNA-Strang* ist ein String $S \in \{\mathrm{A}, \mathrm{C}, \mathrm{G}, \mathrm{T}\}^*$. Ein *Doppelstrang* ist ein Paar von gleich langen DNA-Strängen (S_1, S_2) mit der folgenden Eigenschaft: Für jede Base $S_1[i]$ muss $S_2[i]$ genau das zugehörige Basenpaar ergänzen. Zum Beispiel falls $S_1[3] = \mathrm{T}$ ist, so muss $S_2[3] = \mathrm{A}$ gelten.

Gegeben seien zwei DNA-Stränge S_1, S_2 auf verschiedenen Rechnern. Die beiden Rechner dürfen bei Ausführung des Algorithmus nur 1 Bit untereinander austauschen. Auf beiden Rechnern kann derselbe Pseudozufallsgenerator mit demselben Seed gestartet werden.

Geben Sie ein randomisiertes Vorgehen an, welches entscheidet, ob (S_1, S_2) ein Doppelstrang ist. Bei Eingabe einer Ja-Instanz soll immer korrekt entschieden werden; sonst soll eine konstante Fehlerwahrscheinlichkeit < 1 garantiert werden.

(c) Analyse (6 Punkte)

Welche Fehlerwahrscheinlichkeit hat Ihr Vorgehen bei Nein-Instanzen? Beweisen Sie.

Weiterer Platz für Aufgabe 7(b):

Aufgabe 9: *NP*-Vollständigkeit **(20 Punkte)**

(a) Definition (8 Punkte)

Definieren Sie *NP*-Vollständigkeit, ohne den Begriff „*NP*-schwer“ zu benutzen. Definieren Sie auch den Begriff der Reduktion. Sie können davon ausgehen, dass *NP* definiert ist.

(b) Reduktion (12 Punkte)

Ein Pharao beschäftigt Sklaven, von denen jeder mindestens ein Handwerk beherrscht. Sein oberster Berater wünscht sich eine Handwerkskammer, in der aus jedem Handwerksbereich mindestens ein Sklave vertreten ist.

HANDWERKSKAMMER **Gegeben:** Eine Menge von Sklaven $\mathcal{S}$; für jedes Handwerk $i = 1, ..., h$ eine Teilmenge der Sklaven H_i, sodass $\bigcup_{i=1}^{h} H_i = \mathcal{S}$; eine maximale Kammergröße $g \in \mathbb{N}$. **Gefragt:** Gibt es eine Handwerkskammer $K \subseteq \mathcal{S}$ mit höchstens g Sklaven, sodass aus jedem H_i mindestens ein Sklave in K vertreten ist?

Zeigen Sie, dass HANDWERKSKAMMER ***NP***-schwer ist. Definieren Sie dafür auch das bei Ihrer Reduktion verwendete Problem.

Notizen:

Notizen: