Lehrstuhl für Informatik 1
Prof. Dr. Gerhard Woeginger
Tim Hartmann, Daniel Neuen

WS 2017/18
20.03.2018

Klausur
Berechenbarkeit und Komplexität

Name: ....

Vorname: ....

Matrikelnummer: ....

Studiengang: ....

Hinweise:

- Die Bearbeitungszeit beträgt 120 Minuten.
- Bitte versehen Sie jedes Blatt mit Namen und Matrikelnummer.
- Bitte schreiben Sie deutlich. Unleserliches wird nicht korrigiert und als fehlerhaft gewertet.
- Streichen Sie Konzeptrechnungen, die nicht gewertet werden sollen, durch oder machen Sie sie anderweitig kenntlich. Bei mehreren Lösungsversuchen pro Aufgabe wird der schlechteste gewertet.
- Bitte verwenden Sie einen dokumentenechten Stift mit blauer oder schwarzer Tinte und verwenden Sie keinen Tintenkiller oder Ähnliches. Benutzen Sie ausschließlich das zur Verfügung gestellte Papier.
- Halten Sie bitte Ihren Studierendenausweis und einen Lichtbildausweis zur Kontrolle bereit.
- Bitte schalten Sie Ihre Mobiltelefone aus!

Ich versichere, die Klausur selbstständig bearbeitet zu haben, und mir ist bekannt, dass die Klausur bei einem Täuschungsversuch mit “nicht bestanden” bewertet wird.

....
(Unterschrift)

Aufgabe	1	2	3	4	Gesamt
Punkte	20	20	20	20	80
erreicht					

(Name/Matrikelnummer)

Aufgabe 1:

(a) Definieren Sie, wann eine Menge M **abzählbar** ist. **(3 Punkte)**

(b) Geben Sie (ohne weitere Begründung) eine Sprache L an, sodass weder die Sprache L selbst noch ihr Komplement $\overline{L}$ semi-entscheidbar sind. **(3 Punkte)**

(c) Formulieren Sie den Satz von Matijasevich. **(3 Punkte)**

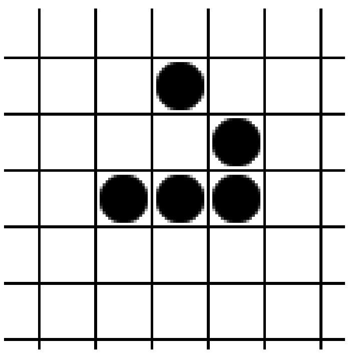
(d) Formulieren Sie die Entscheidungsvariante des Problems **CLIQUE**, ohne dabei das Wort "Clique" zu verwenden. **(3 Punkte)**

(e) Wann hat ein Algorithmus **pseudo-polynomielle** Laufzeit? **(3 Punkte)**

(f) Definieren Sie die Komplexitätsklasse **PSPACE**. **(3 Punkte)**

(g) Auf welche in der Vorlesung behandelten Berechnungsmodelle beziehen sich die folgenden drei Abbildungen? Ordnen Sie Ihre Antworten den Abbildungen zu. **(2 Punkte)**

$\boxed{\textbf{41: CLOAD 5}}$



Nachfolgerfunktion
s(n) = n+1

(Name/Matrikelnummer)

Aufgabe 2:

(a) Eine **BuK-Funktion** ist eine Funktion $f : \mathbb{N} \to \mathbb{N}$, die die folgende Bedingung erfüllt: **(4 Punkte)**

$$f(n) \in \{n, n+1\} \qquad \text{für alle } n \in \mathbb{N}.$$

Anmerkung: $\mathbb{N} = \{0, 1, 2, 3, \ldots\}$ *ist die Menge der nicht-negativen ganzen Zahlen.*

Wir betrachten zunächst eine konkrete BuK-Funktion $f^* : \mathbb{N} \to \mathbb{N}$, die wie folgt definiert ist:

$$f^*(2k) = 2k \;\text{ und }\; f^*(2k+1) = 2k+2 \quad \text{für alle ganzen Zahlen } k \geq 0.$$

Zeigen Sie, dass diese BuK-Funktion f^* **LOOP-berechenbar** ist.

Anmerkung: Für Ihre Lösung dürfen Sie alle in Vorlesung und Übung eingeführten Makros verwenden.

(b) Es seien $f, g : \mathbb{N} \to \mathbb{N}$ zwei BuK-Funktionen, sodass $f(n) \neq g(n)$ für alle $n \geq 1$ gilt. Zeigen Sie: Wenn f **primitiv rekursiv** ist, so ist auch g **primitiv rekursiv**. **(6 Punkte)**

(Name/Matrikelnummer)

(c) Beweisen Sie: Nicht jede BuK-Funktion ist **berechenbar**. **(10 Punkte)**

(Name/Matrikelnummer)

Zusätzliches Papier

(Name/Matrikelnummer)

Aufgabe 3:

(a) Wir betrachten n Prozesse $P_1, \dots, P_n$, die auf einer Maschine bearbeitet werden sollen. Dabei gibt es für jeden Prozess P_i für $i \in \{1, \dots, n\}$ eine Bearbeitungs-Zeit $t_i \in \mathbb{N}, t_i \geq 1$ und ein Zeit-Intervall, gegeben durch eine Startzeit $\ell_i \in \mathbb{N}$ und eine Zielzeit $r_i \in \mathbb{N}$, in dem P_i bearbeitet werden soll. Die Maschine kann zu jedem Zeitpunkt maximal einen Prozess bearbeiten. Ein Prozess P_i kann frühestens ab Zeitpunkt ℓ_i bearbeitet werden. Nach t_i Zeiteinheiten ist der Prozess bearbeitet, und die Maschine kann potentiell sofort beginnen einen neuen Prozess zu bearbeiten. Jeder Prozess P_i für $i \in \{1, \dots, n\}$ muss spätestens zum Zeitpunkt r_i bearbeitet sein. Die Bearbeitung eines Prozesses darf nicht unterbrochen werden. **(2 Punkte)**

Beispiel: Prozesse P_1 mit $t_1 = 1$, $\ell_1 = 1$ und $r_1 = 2$ sowie P_2 mit $t_2 = 3$, $\ell_2 = 0$ und $r_2 = 5$ könnten wie folgt bearbeitet werden: Zum Zeitpunkt 1 wird P_1 gestartet. Zum Zeitpunkt 2 ist Prozess P_1 fertig bearbeitet und die Maschine wieder frei. Somit kann zum Zeitpunkt 2 der Prozess P_2 gestartet werden, der zum Zeitpunkt 5 fertig ist.

Wir betrachten dazu das Entscheidungsproblem PROZESS-PLANUNG:

> Eingabe: Positive ganze Zahlen $t_1, \dots, t_n$ und nicht-negative ganze Zahlen $\ell_1, \dots, \ell_n$ und $r_1, \dots, r_n$
>
> Frage: Gibt es einen Bearbeitungsplan, der alle Prozesse rechtzeitig fertig stellt?

Betrachten Sie die folgende Instanz des PROZESS-PLANUNG Problems:

Prozess	P_1	P_2	P_3	P_4	P_5	P_6	P_7
t_i	1	2	2	3	3	4	4
ℓ_i	9	0	0	0	0	0	0
r_i	10	19	19	19	19	19	19

Ist diese Instanz eine JA-Instanz? Geben sie eine kurze Begründung an.

(b) Formulieren Sie die Zertifikat-Charakterisierung von NP. **(4 Punkte)**

(Name/Matrikelnummer)

(c) Zeigen Sie, dass das in Teil (a) definierte Problem PROZESS-PLANUNG die Zertifikat-Charakterisierung von NP erfüllt: **Beschreiben** Sie Ihr Zertifikat und **analysieren** Sie seine Länge. **Beschreiben** Sie das Verhalten Ihres Verifizierers und **analysieren** Sie seine Laufzeit. **(6 Punkte)**

(Name/Matrikelnummer)

(d) Beweisen Sie durch eine polynomielle Reduktion: PROZESS-PLANUNG ist **NP-schwer**. **(8 Punkte)**

(Name/Matrikelnummer)

Zusätzliches Papier

Aufgabe 4:

(a) Definieren Sie das **uniforme** Kostenmaß für Berechnungen auf der RAM. **(3 Punkte)**
Definieren Sie das **logarithmische** Kostenmaß für Berechnungen auf der RAM.

(b) Beantworten Sie für jede ganze Zahl $m \geq 0$: Wenn die folgende RAM R als Eingabe im Register $c(1)$ eine Zahl m erhält, welcher Wert steht dann bei Termination im Register $c(2)$? Beweisen Sie Ihre Antwort. **(6 Punkte)**

```
 1: CLOAD 5
 2: STORE 2
 3: LOAD 1
 4: IF c(0)>0 THEN GOTO 6
 5: END
 6: CSUB 1
 7: STORE 1
 8: LOAD 2
 9: MULT 2
10: STORE 2
11: GOTO 3
```

Erinnerung: Bei der RAM setzt zum Beispiel der Befehl ADD i *das Register* 0 *auf den Wert* $c(0) = c(0) + c(i)$. *Der Befehl* CADD i *hingegen setzt den Wert von Register* 0 *auf* $c(0) = c(0) + i$.

(Fortsetzung Teil (b))

(c) Analysieren Sie die Laufzeit der RAM R aus Aufgabenteil (b) im **uniformen Kostenmaß**. Nehmen Sie dazu an, dass die Eingabe im Register $c(1)$ eine Binärzahl mit n Bits ist. **(3 Punkte)**

(d) Widerlegen Sie die folgende Aussage: Wenn die RAM R aus Aufgabenteil (b) im **uniformen Kostenmaß** $t(n)$-zeitbeschränkt ist, dann existieren ein Polynom $q : \mathbb{N} \to \mathbb{N}$ und eine $q(n + t(n))$-zeitbeschränkte deterministische TM M, die R simuliert. **(8 Punkte)**

(Name/Matrikelnummer)

Zusätzliches Papier

(Name/Matrikelnummer)

Zusätzliches Papier