Technische Universität München
Fakultät für Informatik
Prof. Tobias Nipkow, Ph.D.
Sascha Böhme, Lars Noschinski

Sommersemester 2011
Lösungsblatt Endklausur
26. August 2011

# Einführung in die Theoretische Informatik

| Name | Vorname | Studiengang | Matrikelnummer |
|---|---|---|---|
| ................ | ................ | ☐ Diplom ☐ Inform.<br>☐ Bachelor ☐ BioInf.<br>☐ Lehramt ☐ Mathe. | ................ |

| Hörsaal | Reihe | Sitzplatz | Unterschrift |
|---|---|---|---|
| ................ | ................ | ................ | ................ |

## Allgemeine Hinweise

- Bitte füllen Sie obige Felder in Druckbuchstaben aus und unterschreiben Sie!
- Bitte schreiben Sie nicht mit Bleistift oder in roter/grüner Farbe!
- Die Arbeitszeit beträgt 120 Minuten.
- Alle Antworten sind in die geheftete Angabe auf den jeweiligen Seiten (bzw. Rückseiten) der betreffenden Aufgaben einzutragen. Auf dem Schmierblattbogen können Sie Nebenrechnungen machen. Der Schmierblattbogen muss ebenfalls abgegeben werden, wird aber in der Regel nicht bewertet.
- Es sind keine Hilfsmittel außer einem DIN-A4-Blatt zugelassen.

Hörsaal verlassen von ...... bis ...... / von ...... bis ......
Vorzeitig abgegeben um ......
Besondere Bemerkungen:

| | A1 | A2 | A3 | A4 | A5 | A6 | Σ | Korrektor |
|---|---|---|---|---|---|---|---|---|
| Erstkorrektur | | | | | | | | |
| Zweitkorrektur | | | | | | | | |

## Aufgabe 1 (8 Punkte)

Wahr oder falsch? Begründen Sie im Folgenden Ihre Antworten möglichst knapp!

1. Ist $L$ nicht entscheidbar, dann ist $L$ oder $\overline{L}$ nicht semi-entscheidbar.
2. Ist $f$ nicht LOOP-berechenbar, dann ist $f$ auch nicht total.
3. Jede totale Funktion $f$ wird von einem WHILE-Programm berechnet.
4. Sei $H_0$ das Halteproblem auf leerem Band. Dann existiert ein regulärer Ausdruck $\alpha$ mit $L(\alpha) = H_0$.
5. Die Funktion $f(n) = \begin{cases} 0, & \text{2034 landen Menschen auf dem Mars} \\ 1, & \text{sonst} \end{cases}$, $f : \mathbb{N} \to \{0, 1\}$, ist berechenbar.
6. Eine Instanz des PCP hat entweder gar keine oder unendlich viele Lösungen.
7. Sei $a(m, n)$ die Ackermann-Funktion. Dann ist die Funktion $f(n) = 1 \dot{-} a(n, n)$ primitiv rekursiv.
8. Gibt es eine Turingmaschine $M$, so dass $\varphi_M$ nicht berechenbar ist?

---

**Lösungsvorschlag**

1. Wahr, folgt direkt aus Satz 4.69.
2. Falsch, die Ackermann-Funktion ist nicht LOOP-berechenbar, aber total.
3. Falsch, denn nicht jede totale Funktion ist berechenbar (Gegenbeispiel $\chi_K$).
4. Falsch, denn dann wäre $H_0$ entscheidbar.
5. Wahr, denn $f$ ist konstant.
6. Wahr. Wenn $i_1 \cdots i_k$ eine Lösung ist, so ist auch $(i_1 \cdots i_k)^m$ für $m > 0$ eine Lösung.
7. Wahr, denn es gilt $a(n, n) > 0$ für alle $n \in \mathbb{N}$, also ist $f(n)$ eine konstante Funktion.
8. Falsch, denn eine Funktion ist genau dann berechenbar, wenn es eine Turingmaschine gibt, die diese Funktion berechnet.

## Aufgabe 2 (4 Punkte)

1. Zeigen Sie, dass die Abstandsfunktion $dist(m,n) = |m - n|$ primitiv rekursiv ist. Dabei bezeichnet $-$ die ganzzahlige Subtraktion.

2. Zeigen Sie, dass die Funktion $qsum(n) = \sum_{i=0}^{n} i^2$ primitiv rekursiv ist.

*Hinweis:* Sie dürfen zusätzlich zu den Basisfunktionen der primitiven Rekursion die folgenden Funktionen als primitiv rekursiv annehmen: Addition $(m+n)$, Multiplikation $(m*n)$, $div(m,n)$, $mod(m,n)$, $pred(n)$, modifizierte Subtraktion $(m \dot{-} n)$, $ifthen(n,a,b)$, Gleichheit $(m = n)$, $c(m,n)$, $p_1(n)$, $p_2(n)$ sowie $q(m) = max\{x \leq m \mid P(x)\}$ für ein PR-Prädikat $P$. Sie dürfen die erweiterte Komposition und das erweiterte rekursive Definitionsschema verwenden. LOOP- oder WHILE-Programme sind nicht erlaubt.

---

### Lösungsvorschlag

1. $dist(n,m) = (n \dot{-} m) + (m \dot{-} n)$

2. 

$$\begin{aligned} qsum(0) &= 0 \\ qsum(n+1) &= (n+1)*(n+1) + qsum(n) \end{aligned}$$

## Aufgabe 3 (8 Punkte)

Sei $\Sigma$ ein Alphabet. Für jedes Wort $w \in \Sigma^*$ bezeichnet $w^R$ das Wort, das man durch Rückwärtsschreiben von $w$ erhält.

1. Zeigen Sie: Die Sprache $L = \{w \mid \varphi_w(v) = v^R \text{ für alle } v \in \Sigma^*\}$ ist unentscheidbar.

2. Ist es entscheidbar, ob es für eine Turingmaschine $M = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F)$ eine nicht-leere Eingabe gibt, auf der mindestens einer der beiden folgenden Fälle eintritt?

   - $M$ verändert das Band (das heißt, $M$ führt einen Übergang $\delta(q, a) = (q', b, X)$ mit $a \neq b$ aus) oder
   - $M$ liest mindestens ein $\Box$.

   Beweisen Sie Ihre Behauptung.

3. Zeigen oder widerlegen Sie: Ist $A$ semi-entscheidbar und $B$ entscheidbar, so ist $A \setminus B$ semi-entscheidbar.

---

**Lösungsvorschlag**

1. Sei $F = \{\varphi \mid \varphi(v) = v^R \text{ für alle } v \in \Sigma^*\}$. Dann ist $F$ eine nicht-triviale Menge berechenbarer Funktionen und es gilt $L = \{w \mid \varphi_w \in F\}$. Nach dem Satz von Rice ist dann $L$ unentscheidbar.

2. Ja, dieses Problem ist entscheidbar. Dazu reicht es aus, das Verhalten von $M$ auf allen Eingaben der Länge 1 zu betrachten: Sei $a \in \Sigma$ und $u \in \Sigma^*$: Dann verhält sich $M$ auf $a$ und $au$ gleich, bis das erste Mal der Kopf bewegt wird. Wird aber der Kopf bewegt, so terminiert $M$ entweder direkt (dann unterscheidet sich das Verhalten auf $a$ und $au$ nicht) oder $M$ liest im nächsten Schritt auf $a$ ein Blank (und damit wissen wir, dass $M$ einen der beiden Fälle erreichen kann).

   Es bleibt noch zu zeigen, dass wir entscheiden können, ob $M$ auf einer Eingabe der Länge 1 einen der beiden Fälle erreicht. Aber zum Entscheiden müssen wir höchstens die ersten $|Q|$ Schritte betrachten: Wenn $M$ weder den Kopf bewegt (und damit ein Blank liest) noch das Band verändert, so verändert sich an der Konfiguration der TM nur der Zustand. Daher wiederholt sich nach spätestens $|Q|$ Schritten eine Konfiguration (und damit das Verhalten von $M$).

3. Da $A$ semi-entscheidbar und $B$ entscheidbar ist, sind die beiden folgenden Funktionen berechenbar.

   $$\chi'_A(x) = \begin{cases} 1 & \text{falls } x \in A \\ \perp & \text{sonst} \end{cases} \qquad \chi_B(x) = \begin{cases} 1 & \text{falls } x \in B \\ 0 & \text{sonst} \end{cases}$$

   Aus diesen Funktionen können wir jetzt $\chi'_{A\setminus B}$ konstruieren:

   $$\chi'_{A\setminus B}(x) = \begin{cases} 1 & \text{falls } \chi'_A(x) = 1 \text{ und } \chi_B(x) = 0 \\ \perp & \text{sonst} \end{cases}$$

   Die Berechenbarkeit von $\chi'_{A\setminus B}$ folgt direkt aus der Berechenbarkeit von $\chi'_A$ und $\chi_B$. Damit ist $A \setminus B$ semi-entscheidbar.

## Aufgabe 4 (6 Punkte)

Sei $NAND$ die Menge aller aussagenlogischen Formeln, die ausschließlich aus der Konstanten 1, logischen Variablen $x_i$ mit $i \in \mathbb{N}$ und der binären Operation $\uparrow$ als Operationszeichen aufgebaut sind, wobei natürlich auch Klammern zugelassen sind. Dabei sei die Operation $\uparrow$ so definiert, dass $x_i \uparrow x_j$ die gleiche Wahrheitstafel wie $\neg(x_i \wedge x_j)$ hat.

1. Finden Sie eine Formel $F$ in $NAND$, die äquivalent zu $\neg x_0$ ist und in der $x_0$ nur einmal vorkommt.

2. Wie betrachten das Problem $NSAT$:

   **Gegeben:** $F \in NAND$

   **Problem:** Ist $F$ erfüllbar, das heißt, gibt es eine Belegung der Variablen mit Konstanten 0 oder 1, so dass $F$ den Wert 1 annimmt?

   Zeigen Sie: $NSAT$ ist NP-vollständig.

   Sie dürfen dazu benutzen, dass das $SAT$-Problem NP-vollständig ist. Außerdem dürfen Sie das Resultat aus der ersten Teilaufgabe verwenden.

---

**Lösungsvorschlag**

1. $\neg x_0 = \neg(x_0 \wedge 1) = x_0 \uparrow 1$

2. - $NSAT \leq_p SAT$ ($NSAT$ ist in NP):

     Sei $f$ die Abbildung, die in jeder Formel $F$ aus $NAND$ jedes Vorkommen der von Teilformeln $a \uparrow b$ durch $\neg(a \wedge b)$ ersetzt. Dann ist $f$ total und in polynomieller Zeit berechenbar. Da $SAT$ in NP liegt und NP nach unten abgeschlossen ist, liegt also auch $NSAT$ in NP.

   - $SAT \leq_p NSAT$ ($NSAT$ ist NP-hart):

     Sei $f$ die Abbildung, die in jeder Formel $F$ aus $SAT$ mit Teilformeln $a$ und $b$ jedes Vorkommen

     – der Negation $\neg a$ durch $a \uparrow 1$,
     – der Konjunktion $a \wedge b$ durch $\neg\neg(a \wedge b) = \neg(a \uparrow b) = (a \uparrow b) \uparrow 1$ und
     – der Disjunktion $a \vee b$ durch $\neg(\neg a \wedge b) = \neg a \uparrow \neg b = (a \uparrow 1) \uparrow (b \uparrow 1)$

     ersetzt. Dann ist $f$ total und in polynomieller Zeit berechenbar. Da $SAT$ bereits NP-hart ist, ist somit auch $NSAT$ NP-hart.

     *Hinweis:* Die Darstellung von $\neg a$ durch $a \uparrow a$ liefert keine polynomielle Reduktion und damit auch keinen Beweis dafür, dass $NSAT$ NP-hart ist.

## Aufgabe 5 (6 Punkte)

Sei $\Sigma = \{a, b\}$.

1. Bestimmen Sie zwei unterschiedliche Lösungen mit je höchstens drei „Karten“ für die folgende Instanz des PCP:

$$P_1 = (\ \overbrace{(abba, baab)}^{1},\ \overbrace{(b, bab)}^{2},\ \overbrace{(aaba, aa)}^{3}\ )$$

2. Begründen Sie, dass die folgende Instanz des PCP keine Lösung hat:

$$P_2 = (\ \overbrace{(baa, aba)}^{1},\ \overbrace{(aa, ba)}^{2},\ \overbrace{(aab, ab)}^{3}\ )$$

3. Beweisen Sie: Gilt $|x_i| \geq |y_i|$ für alle Tupel $(x_i, y_i)$ einer Instanz $P$ des PCP, dann ist $P$ entscheidbar.

---

**Lösungsvorschlag**

1. Die einzig möglichen Lösungen mit höchstens drei Karten sind 3 2 und 3 1 2.

2. Keines der Tupel kann als erstes Element einer Lösung vorkommen, da sich dann entweder das erste oder das zweite Zeichen von $x$ und $y$ unterscheiden würden.

3. Ein Entscheidungsverfahren für $P$ beruht auf folgender Äquivalenz:

   $P$ hat eine Lösung genau dann, wenn es Tupel $(x_i, y_i)$ gibt mit $x_i = y_i$.

   Damit muß eine Turingmaschine $M$, die $P$ entscheidet, nur die Wörter in allen Tupeln auf Gleichheit testen. Gibt es zumindest ein Paar, das aus zwei gleichen Wörtern besteht, dann gibt $M$ 1 zurück, ansonsten 0. $M$ terminiert immer, da es nur endlich viele Tupel in $P$ gibt.

   Es bleibt zu beweisen, dass die obige Äquivalenz tatsächlich gilt.

   $\Longrightarrow$: Wenn $P$ eine Lösung $i_1 \cdots i_n$ besitzt, dann kann man folgende Fallunterscheidung machen:

   – $|x_{i_1}| = |y_{i_1}|$
     Da $i_1 \cdots i_n$ eine Lösung von $P$ ist, d.h., es gilt $x_{i_1} \cdots x_{i_n} = y_{i_1} \cdots y_{i_n}$, muss folglich auch $x_{i_1} = y_{i_1}$ gelten.

   – $|x_{i_1}| > |y_{i_1}|$
     Damit $i_1 \cdots i_n$ eine Lösung von $P$ ist, muss es aufgrund des Längenvergleiches ein Tupel $(x_k, y_k)$ in $P$ geben, für das $|x_k| < |y_k|$ gilt. Dies gilt aber für kein Tupel in $P$, und damit kann dieser Fall nicht eintreten.

   $\Longleftarrow$: Gibt es in $P$ ein Tupel $(x_i, y_i)$ mit $x_i = y_i$, dann ist dieses Tupel auch eine Lösung von $P$.

## Aufgabe 6 (8 Punkte)

Sei $\Sigma = \{0,1\}$ ein Alphabet und $f : \Sigma^* \to \Sigma^*$ die Funktion, die jede 0 aus einem Wort löscht (z.B. gilt $f(00100110) = 111$). Formal ist $f$ durch die folgenden Gleichungen definiert:

$$f(\epsilon) = \epsilon \qquad f(0w) = f(w) \qquad f(1w) = 1f(w)$$

Konstruieren Sie eine 1-Band-Turingmaschine $M$, die $f$ berechnet. Geben Sie $M$ als Tupel an und beschreiben Sie die Zustandsübergangsfunktion als Graph, Tabelle oder Liste von Gleichungen. Kommentieren Sie ihre Konstruktion durch eine informelle Beschreibung Ihrer Lösungsidee.

---

**Lösungsvorschlag**

Wir konstruieren eine DTM $M$, die wie folgt vorgeht: Das Band wird von links nach rechts durchlaufen. Eine 1 wird einfach übergangen, bei einer 0 wechselt die TM in einen anderen Zustand, in dem sie die nächste 1 sucht und diese an allen 0-en vorbei nach links schiebt. Dann wechselt die TM wieder in den Anfangszustand. Sind alle 0-en am Ende des Bandes, so werden diese gelöscht, der Kopf auf den Bandanfang bewegt und die Turingmaschine terminiert.

Formal definieren wir $M = (Q, \Sigma, \Sigma \cup \{\Box\}, \delta, q_0, \Box, \{q_f\})$ wobei $Q = \{q_0, \ldots, q_4, q_f\}$ und $\delta$ durch die folgende Tabelle gegeben ist:

| | 0 | 1 | $\Box$ | |
|---|---|---|---|---|
| $q_0$ | $(q_1, 0, R)$ | $(q_0, 1, R)$ | $(q_4, \Box, L)$ | Erste 0 suchen |
| $q_1$ | $(q_1, 0, R)$ | $(q_2, 0, L)$ | $(q_4, \Box, L)$ | 0-en überspringen |
| $q_2$ | $(q_2, 0, L)$ | $(q_3, 1, R)$ | $(q_3, \Box, R)$ | 1 nach links schieben |
| $q_3$ | $(q_0, 1, R)$ | | | Schieben abschliessen |
| $q_4$ | $(q_4, \Box, L)$ | $(q_4, 1, L)$ | $(q_f, \Box, R)$ | Band aufräumen |