Maike Buchin
Stef Sijben

27. Januar 2015

# Probeklausur

## Theoretische Informatik

Bearbeitungszeit: 2,5 Stunden

Name: ______________________________

Matrikelnummer: ______________________________

Studiengang: ______________________________

Geburtsdatum: ______________________________

**Hinweise:**

- Schreibe die Lösung jeder Aufgabe direkt auf das Blatt mit der Aufgabenstellung. Es dürfen Vorder- und Rückseite verwendet werden. Wenn der Platz nicht ausreicht, können die leeren letzten Seiten benutzt werden.
- Trenne die Blätter nicht auseinander und schreibe im eigenen Interesse *leserlich*. Was wir nicht verstehen, können wir auch nicht werten.
- Bitte keine Bleistifte verwenden.
- Die <u>einzigen</u> zugelassenen Hilfsmittel sind das Buch „Theoretische Informatik - kurzgefasst“ von Uwe Schöning, die Folien und Skripte zur Vorlesung, sowie ein deutsches Wörterbuch.

---

| Aufgabe | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Punkte | | | | | |

| Klausurpunkte | Note |
|---|---|
| | |

# I Automatentheorie und formale Sprachen

**Aufgabe 1** (20 Punkte)
Über dem Alphabet $\Sigma = \{0, 1\}$ sei folgender NFA gegeben:

| $\delta$ | $z_0$ | $z_1$ | $z_2$ | $z_3$ | $z_4$ |
|---|---|---|---|---|---|
| 0 | $\{z_0\}$ | $\{z_2\}$ | $\{z_1\}$ | $\{z_3\}$ | $\{z_3\}$ |
| 1 | $\{z_1\}$ | $\{z_0\}$ | $\{z_2\}$ | $\{z_4\}$ | $\{z_4\}$ |

Die Startzustände sind $\{z_0, z_3\}$ und die Endzustände sind $\{z_0, z_4\}$.

a) Zeichne den Zustandsgraphen des Automaten.

b) Konstruiere mittels des Verfahrens aus der Vorlesung einen DFA, der die gleiche Sprache akzeptiert. Zustände, die vom Startzustand aus nicht erreichbar sind, können weggelassen werden.

c) Minimiere den DFA aus Teilaufgabe b).

**Aufgabe 2** (20 Punkte)

Sei $G = (V, \Sigma, P, S)$ eine kontextfreie Grammatik mit $V = \{S, A, B, C\}$, $\Sigma = \{a, b\}$, Startvariable $S$ und folgenden Regeln $P$:

$$
\begin{aligned}
S &\to SA \mid BB \mid b \\
A &\to AC \mid a \\
B &\to CC \mid SB \mid a \\
C &\to AS \mid CB \mid b
\end{aligned}
$$

Prüfe mit dem CYK-Algorithmus ob das Wort $w = baaba$ durch $G$ erzeugt werden kann. Fülle dazu die folgende Tabelle aus.

| $i \to$ <br> $j \downarrow$ | $b$ | $a$ | $a$ | $b$ | $a$ |
|---|---|---|---|---|---|
| 1 | | | | | |
| 2 | | | | | |
| 3 | | | | | |
| 4 | | | | | |
| 5 | | | | | |

**Aufgabe 3** (20 Punkte)

Zeige mit Hilfe des Pumping-Lemmas, dass folgende Sprache $L$ über dem Alphabet $\Sigma = \{0, 1\}$ nicht regulär ist:

$$L = \{w\overline{w} \mid w \in \Sigma^*\}.$$

Hier bedeutet $\overline{w}$ das Wort, das man erhält wenn man alle nullen in $w$ durch einsen ersetzt und umgekehrt. Zum Beispiel:

$$\overline{\varepsilon} = \varepsilon$$
$$\overline{0} = 1$$
$$\overline{1001} = 0110$$

# II Berechenbarkeitstheorie

**Aufgabe 4** (20 Punkte)
Zur Bearbeitung dieser Aufgabe dürfen die folgenden Operatoren und Konstrukte verwendet werden:

- $x_i := x_j$
- $x_i := c$
- IF $x = 0$ THEN $A$ END
- $x_i := x_j + x_k$
- $x_i := x_j * x_k$
- $x_i := x_j$ DIV $x_k$
- $x_i := x_j$ MOD $x_k$

Weitere Konstrukte sind erlaubt, wenn man sie vorher mit Hilfe der bereits bekannten Konstrukte definiert. Außerdem dürfen auch andere Variablennamen als $x_0, \ldots, x_k$ verwendet werden. Es muss jedoch angegeben werden, welche Variablen die Ein- und Ausgabe enthalten sollen.

Gebe ein WHILE-Programm an, das folgende Funktion $f : \mathbb{N}_0 \to \mathbb{N}_0$ berechnet:

$$f(n) = \begin{cases} m, & \text{falls } m \text{ die kleinste Zahl mit } m \geq n \text{ ist,} \\ & \qquad \text{so dass } (m, m+2) \text{ ein Primzahl-Zwillingspaar ist.} \\ \text{undefiniert}, & \text{falls es so ein } m \text{ nicht gibt.} \end{cases}$$

$(m, m+2)$ ist ein Primzahl-Zwillingspaar, wenn $m$ und $m+2$ beiden Primzahlen sind. Also, die kleinsten Primzahl-Zwillingspaare sind $(3, 5)$, $(5, 7)$ und $(11, 13)$.
*Hinweis*: Beschreibe zuerst ein Konstrukt, das berechnet ob eine Zahl $i$ Prim ist.

Beschreibe die Arbeitsweise Deines Programms.

# III Komplexitätstheorie

**Aufgabe 5** (20 Punkte)

Betrachte folgendes Problem:

NOT-ALL-EQUAL 4SAT
**Eingabe:** Eine Boolesche Formel $F$ in konjunktiver Normalform (Klauselform) über den Variablen $x_1, \ldots, x_n$ mit höchstens 4 Literalen pro Klausel.
**Frage:** Gibt es eine Belegung der Variablen mit Booleschen Konstanten, so dass in jeder Klausel mindestens ein Literal wahr ist und mindestens ein Literal nicht wahr?

Zeige, dass das Problem „NOT-ALL-EQUAL 4SAT" NP-hart ist.