Technische Universität Berlin, Berlin, 13.10.2016

Name: .... **Matr.-Nr.: ....**

Wiederholung Schriftliche Prüfung: Berechenbarkeit und Komplexität

(Niedermeier/Chen/Froese/Sorge, Sommersemester 2016)

Einlesezeit:	15 Minuten
Bearbeitungszeit:	60 Minuten
Max. Punktezahl:	50 Punkte

1	**2**	**3**	**4**	**5**	**Σ**
(12)	(8)	(8)	(12)	(10)	(50)

Allgemeine Hinweise:

- Es sind keinerlei Hilfsmittel erlaubt.
- Benutzen Sie einen dokumentenechten Stift in der Farbe schwarz oder blau. Insbesondere also keinen Bleistift, sondern einen Kugelschreiber oder einen nicht löschbaren Füller.
- Beschriften Sie jedes Blatt mit Vor- und Nachnamen und Ihrer Matrikelnummer.
- **Falls es in der Aufgabenstellung nicht explizit ausgeschlossen wird, so sind alle Antworten zu begründen! Antworten ohne Begründung erhalten 0 Punkte.**

Viel Erfolg!

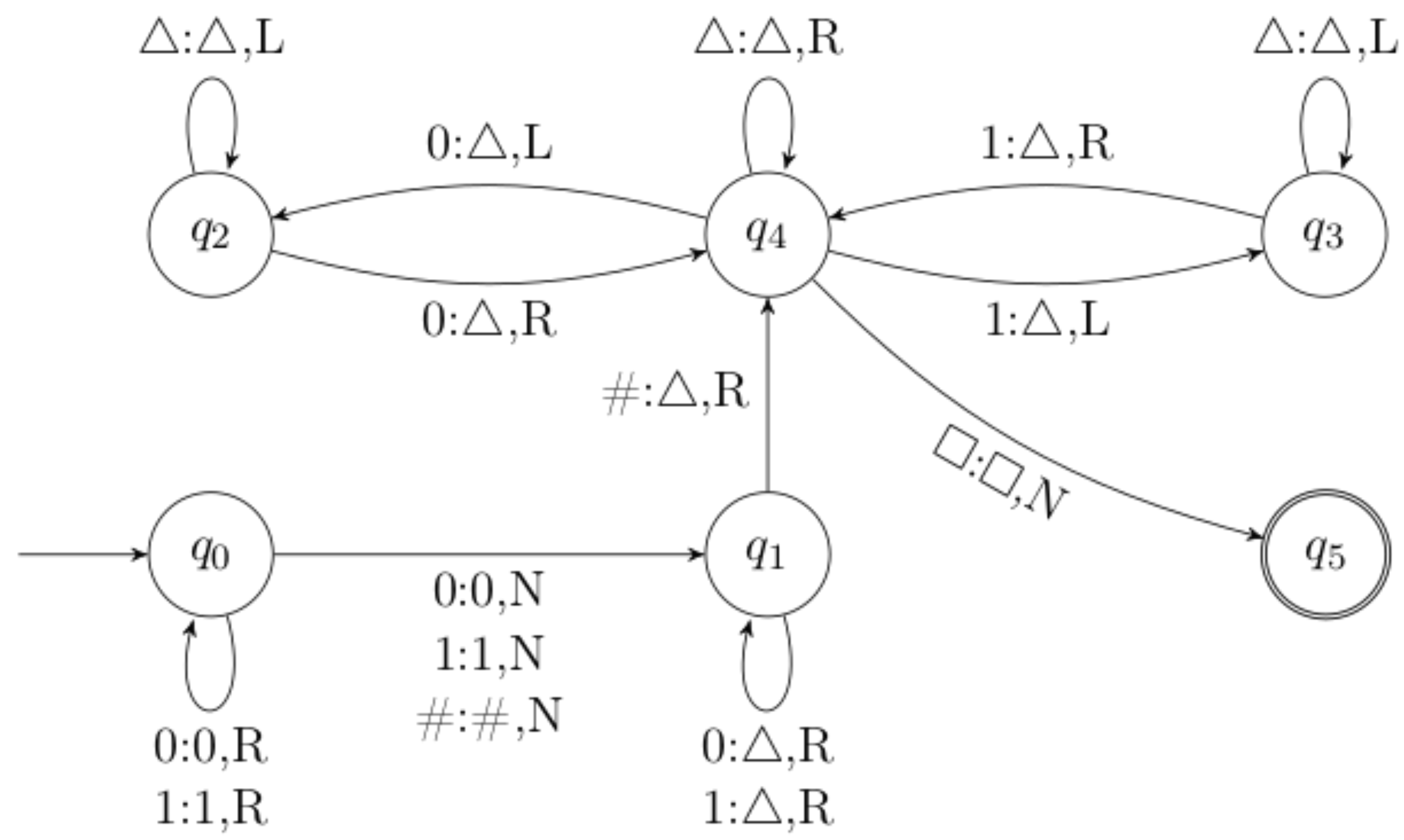
Name: .... **Matr.-Nr.: ....**

Aufgabe 1: **Eine Turing-Maschine** (7 + 5 Punkte)

Betrachten Sie die Turing-Maschine

$$M = (\{q_0, q_1, q_2, q_3, q_4, q_5\}, \{0, 1, \#\}, \{0, 1, \#, \triangle, \square\}, \delta, q_0, \square, \{q_5\}),$$

wobei δ wie folgt definiert ist:



Hinweis: Beispielsweise bedeutet der Pfeil mit Beschriftung „#:△,R" von q_1 zu q_4, dass M im Zustand q_1 beim Lesen von # in den Zustand q_4 übergeht, die # durch △ ersetzt, und ihren Leseschreibkopf nach rechts bewegt.

a) Die von der Turing-Maschine M akzeptierte Sprache $T(M)$ entspricht genau einer der folgenden Sprachen A, B, C. Welcher? Dabei genügt es, für jede nicht erkannte Sprache ein Gegenbeispiel und eine dazugehörige Begründung anzugeben. Hierbei ist $\mathrm{rev}\colon \{0,1\}^* \to \{0,1\}^*$ die Funktion, die jedem Wort $v \in \{0,1\}^*$ seine Umkehrung zuordnet. Beispielsweise ist $\mathrm{rev}(011) = 110$.

- $A = \{w\#v \mid (w, v \in \{0,1\}^*) \wedge \big(w = \mathrm{rev}(v)\big)\}$
- $B = \{w\#v \mid (w, v \in \{0,1\}^*) \wedge \big(\exists u \in \{0,1\}^*\colon w = u\,\mathrm{rev}(v)\big)\}$
- $C = \{w\#v \mid (w, v \in \{0,1\}^*) \wedge \big(\exists u_1, u_2 \in \{0,1\}^*\colon w = u_1\,\mathrm{rev}(v) u_2\big)\}$

b) Finden Sie ein Wort $w\#v$ mit $w \in \{0,1\}^2$ und $v \in \{0,1\}$ das von der Turing-Maschine M *nicht* akzeptiert wird und beweisen Sie, warum das so ist.

Name: .... **Matr.-Nr.:** ....

Aufgabe 2: **Rekursive Aufzählbarkeit** (4+4 Punkte)

Zur Erinnerung: Eine Sprache $L \subseteq \Sigma^*$ heißt rekursiv aufzählbar, falls $L = \emptyset$ oder falls es eine totale, berechenbare Funktion $f\colon \mathbb{N} \to \Sigma^*$ derart gibt, dass $L = \{f(0), f(1), f(2), \ldots\} = f(\mathbb{N})$. Wir sagen auch: „$f$ zählt L auf."

Geben Sie für die folgenden Sprachen jeweils eine Funktion f an, die die entsprechende Sprache aufzählt, und begründen Sie die Korrektheit (inklusive Berechenbarkeit) von f.

a) $L_1 = \{a^p b^p \mid p \in \mathbb{N} \wedge (p \text{ ist eine Primzahl})\}$.
Hinweis: Sie können davon ausgehen, dass die $i$te Primzahl berechenbar ist.

b) $L_2 = A \cup B$, wobei $A, B \subseteq \Sigma^*$ zwei Sprachen sind, sodass $A \setminus B$, $B \setminus A$ und $A \cap B$ rekursiv aufzählbar sind.

Name: .... **Matr.-Nr.:** ....

Aufgabe 3: **Reduzierbarkeit und Halteprobleme** (4+4 Punkte)

Gegeben seien zwei Sprachen $A \subseteq \Sigma^*$ und $B \subseteq \Gamma^*$. Begründen oder widerlegen Sie die folgenden Aussagen.

a) Wenn $A \leq B$, dann $(\Sigma^* \setminus A) \leq (\Gamma^* \setminus B)$.

b) Wenn $A \leq B$, dann $(\Gamma^* \setminus B) \leq (\Sigma^* \setminus A)$.

Hinweis: $A \leq B$ bedeutet, dass A reduzierbar auf B ist. Nutzen Sie die Eigenschaften von Reduktionen und den Fakt, dass zum Beispiel das spezielle Halteproblem K,

$$K = \{w \mid \text{die durch } w \text{ kodierte Turing-Maschine } M_w \text{ hält auf Eingabe } w\},$$

unentscheidbar ist.

Name: .... **Matr.-Nr.:** ....

Aufgabe 4: **NP-Vollständigkeit und Polynomzeitreduktion** (3 + 6 + 3 Punkte)

Eine *Bipartition* einer Menge V ist ein Tupel (A, B) mit $A, B \subseteq V$, $A \cap B = \emptyset$ und $A \cup B = V$. Sei $G = (V, E)$ ein Graph und (A, B) eine Bipartition seiner Knotenmenge V. Wir sagen, dass eine Kante $\{u, v\} \in E$ die Bipartition (A, B) *kreuzt*, wenn $u \in A$ und $v \in B$, oder $u \in B$ und $v \in A$. Beispielsweise wird die Bipartition $(\{b\}, \{a, c, d\})$ der Knotenmenge des in Teilaufgabe a) dargestellten Graphen von drei Kanten gekreuzt.

Betrachten Sie folgende Probleme:

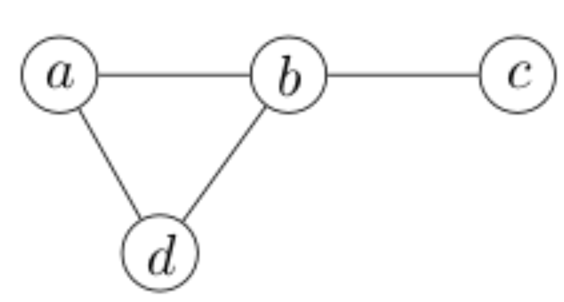
Maximum Cut
Eingabe: Ein ungerichteter Graph $G = (V, E)$ und eine natürliche Zahl $k \in \mathbb{N}$.
Frage: Existiert eine Bipartition der Knotenmenge V, die von *mindestens* k Kanten gekreuzt wird?

Minimum Bisection
Eingabe: Ein ungerichteter Graph $G = (V, E)$ und eine natürliche Zahl $\ell \in \mathbb{N}$.
Frage: Existiert eine Bipartition (A, B) der Knotenmenge V mit $|A| = |B|$, die von *höchstens* ℓ Kanten gekreuzt wird?

Betrachten Sie folgende Transformation f einer beliebigen Instanz $\langle G = (V, E), k\rangle$ von Maximum Cut in eine Instanz $f(\langle G, k\rangle) = \langle H, \ell\rangle$ von Minimum Bisection:

- Die Knotenmenge von H setzt aus der Knotenmenge V von G und einer neuen Knotenmenge V' zusammen, wobei $|V'| = |V|$.
- Die Kantenmenge von H enthält die Menge $E' = \{\{u, v\} \mid (u, v \in V) \wedge (\{u, v\} \notin E)\}$ aller Kanten, die *nicht* in G enthalten sind, sowie die Kantenmenge $E'' = \{\{u, v\} \mid (u \in V \cup V') \wedge (v \in V')\}$, die jeden neuen Knoten in V' mit allen anderen Knoten verbindet.
- Zusammengefasst ist $H \coloneqq (V \cup V', E' \cup E'')$.
- Setze $\ell \coloneqq |V|^2 - k$.

a) Geben Sie $f(\langle G, k\rangle) = \langle H, \ell\rangle$ an, wobei $k = 3$ und G wie folgt definiert ist:



Hinweis: Für den Graphen H in $f(\langle G, k\rangle)$ genügt eine bildliche Darstellung.

b) Beweisen Sie, dass obige Transformation f eine Polynomzeitreduktion von Maximum Cut auf Minimum Bisection ist.

c) Maximum Cut ist NP-schwer. Beweisen Sie, dass Minimum Bisection NP-vollständig ist.

Name: .... **Matr.-Nr.:** ....

Aufgabe 5: **Vermischtes zu P und NP** (4 + 3 + 3 Punkte)

Beweisen oder widerlegen Sie folgende Aussagen.

a) Eine Sprache $L \subseteq \Sigma^*$ ist in NP, wenn sie die folgende Eigenschaft erfüllt: Es existieren ein Polynom $p\colon \mathbb{N} \to \mathbb{N}$ und eine *deterministische* Turing-Maschine M mit $\mathsf{time}_M(n) \in n^{O(1)}$, sodass für jedes Wort $x \in \Sigma^*$ gilt

$$(x \in L) \Leftrightarrow \big(\exists u \in \Sigma^{p(|x|)} \colon \langle x, u \rangle \in T(M)\big).$$

Hierbei ist $T(M)$ die von der Turing-Maschine M akzeptierte Sprache.

b) Gegeben seien zwei Sprachen $A, B \subseteq \Sigma^*$ mit $A, B \notin \{\emptyset, \Sigma^*\}$. Falls $A, B \in \text{NP}$, dann ist A reduzierbar auf B.

c) Falls eine Sprache $L \subseteq \Sigma^*$ in P ist, dann ist auch $(\Sigma^* \setminus L)$ in P.