
Prüfungsteilnehmer

Prüfungstermin

Einzelprüfungsnummer

Kennzahl: _____

**Frühjahr
2008**

Kennwort: _____

66115

Arbeitsplatz-Nr.: _____

Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen
— Prüfungsaufgaben —

Fach: **Informatik (vertieft studiert)**

Einzelprüfung: **Theoretische Informatik, Algorithmen**

Anzahl der gestellten Themen (Aufgaben): 2

Anzahl der Druckseiten dieser Vorlage: 6

Bitte wenden!

Thema Nr. 1**Aufgabe 1:****Reguläre Sprachen**

Sei $\Sigma = \{a, b\}$ ein Alphabet. Wir betrachten die Sprache

$$L = \{w_1 \cdots w_n \in \Sigma^* \mid n \geq 2 \wedge \exists 1 \leq i < n : w_n = w_i \wedge \forall 1 \leq i \leq n : w_i \in \Sigma\}$$

als die Sprache, die alle Wörter enthält, deren letzter Buchstabe vorher bereits schon einmal im Wort vorkam.

- Geben Sie für das Alphabet $\Sigma = \{a, b\}$ einen nichtdeterministischen endlichen Automaten N_1 an, der L erkennt.
- Übersetzen Sie den Automaten N_1 in einen deterministischen endlichen Automaten, indem Sie die Potenzmengenkonstruktion durchführen.
- Geben Sie eine Chomsky-Typ-3-Grammatik G an, die L erzeugt.
- Zeigen Sie, dass G das Wort $aaabab$ erzeugt.

Aufgabe 2:**Kontextfreie Sprachen**

Zeigen Sie, dass die Sprache $L = \{a^n b^m \mid n > m\}$ genau eine Chomsky-Typ-2-Sprache ist!

Aufgabe 3:**Turing-Maschinen**

Sei $\Sigma = \{a, b, c\}$ ein endliches Alphabet. Wir betrachten die Sprache $L = \{w \in \Sigma^* \mid \forall s \in \Sigma \setminus \{a\} : |w|_s < |w|_a\}$, also die Menge der Wörter, in denen a echt häufiger als die anderen Buchstaben vorkommt.

- Geben Sie eine deterministische Turingmaschine an, die L für das Alphabet Σ entscheidet. Die Maschine soll dabei wie folgt verfahren:
 - Das Arbeitsalphabet Γ soll zusätzliche Zeichen \square und \diamond enthalten.
 - Das Wort steht initial als $\square w \square$ auf dem Band der Turingmaschine.
 - Die Turingmaschine sucht nach jeweils einem a, b und c und überschreibt das jeweils erste gefundene Zeichen mit \diamond .
 - Wird dabei kein a gefunden, ist das Wort zu verwerfen. Wird ein a gefunden, aber weder b noch c , wird das Wort akzeptiert. Beachten Sie, dass es dabei möglich ist, ein a und c , aber kein b , oder ein a und b , aber kein c zu finden.
 - Nach einem solchen Durchgang fährt die Turingmaschine nach links und beginnt von vorne.

Beschreiben Sie dabei die Bedeutung der Zustände $q \in Q$ Ihrer Turingmaschine informell.

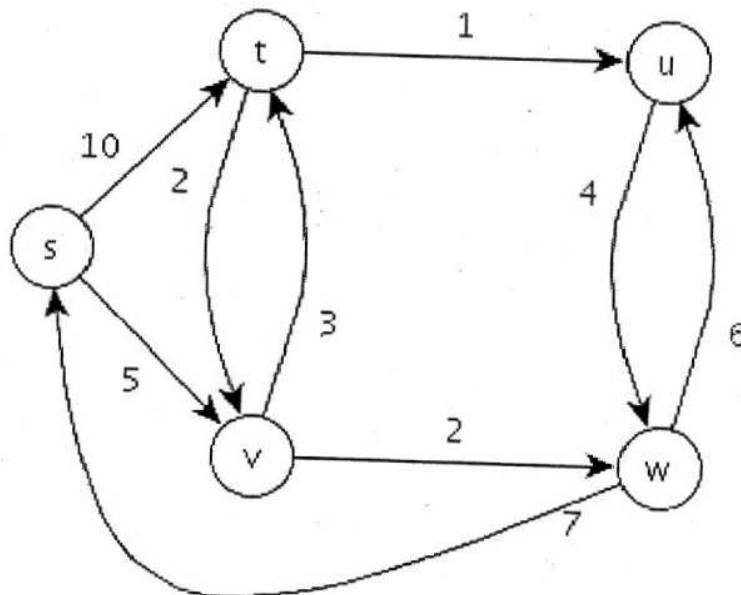
- b) Geben Sie den Ablauf Ihrer Turingmaschine über dem Wort *aba* an.
- c) Geben Sie Laufzeit- und Speicherplatzkomplexität Ihrer Turingmaschine M in O -Notation an. Kann die Laufzeitkomplexität durch die Verwendung einer Mehrband-Turingmaschine verbessert werden? Begründen Sie Ihre Antworten.

Thema Nr. 2

Hinweis: Die einzelnen Teilaufgaben bauen oftmals aufeinander auf, sind aber im Prinzip in beliebiger Reihenfolge lösbar. Sie dürfen hierbei die Angaben und Ergebnisse früherer Teilaufgaben uneingeschränkt zur Lösung späterer Teilaufgaben verwenden! Außerdem dürfen Sie Tatsachen aus dem Informatik-Duden ohne weitere Begründung als bekannt voraussetzen.

Aufgabe 1:Effiziente Algorithmen

- a) Führen Sie den Dijkstra-Algorithmus zur Bestimmung aller kürzesten Pfade vom Startknoten s am folgenden Graphen aus:



Dokumentieren Sie Ihre Schritte geeignet.

- b) Der Dijkstra-Algorithmus benutzt einen Heap R , in dem diejenigen Knoten verwaltet werden, deren Entfernung zu s noch nicht endgültig feststeht. Sei v_1, v_2, \dots, v_n eine Anordnung der Knoten in der Reihenfolge, in der sie aus R mittels $\text{deletemin}(R, v)$ herausgenommen werden. Weisen Sie nach, dass in dieser Anordnung die Knoten aufsteigend nach der Entfernung $d[v]$ von s sortiert sind.

Hinweis:

Zeigen Sie zuerst, dass für alle benachbarten Paare (v_i, v_{i+1}) von Knoten gilt: $v_i \leq v_{i+1}$.

Aufgabe 2:**Reguläre Sprachen**

- a) Konstruieren Sie einen deterministischen endlichen Automaten, der die Sprache akzeptiert, die durch den regulären Ausdruck $(a + b)^* \cdot a \cdot (a + b) \cdot (a + b) \cdot (a + b)$ beschrieben ist.

Hinweis:

Konstruieren Sie zuerst einen nichtdeterministischen endlichen Automaten, und wandeln Sie diesen anschließend in einen deterministischen endlichen Automaten um!

- b) Geben Sie einen nichtdeterministischen endlichen Automaten (NEA) mit maximal 4 Zuständen an, der die folgende reguläre Sprache akzeptiert:

$$L = \{ xy \mid x, y \in \{a, b\}^* \quad x \text{ endet mit } b \text{ und die Anzahl der Zeichen } a \text{ in } y \text{ ist durch } 3 \text{ teilbar} \}$$

Ein Wort w ist somit in L , falls das Zeichen b so in w vorkommt, dass die Zahl der Zeichen a , die nach diesem b stehen, durch 3 teilbar ist.

- c) Im Gegensatz zu deterministischen endlichen Automaten ist kein einfacher Minimierungsalgorithmus für nichtdeterministische endliche Automaten bekannt. Sie sollen hier dennoch nachweisen, dass jeder nichtdeterministische endliche Automat, der die in Teilaufgabe b) beschriebene Sprache L akzeptiert, mindestens 4 Zustände besitzt.

Betrachten Sie eine Zerlegung des Wortes $w = baaa$ in die folgenden Paare: $(x_1, y_1) = (\epsilon, baaa)$, $(x_2, y_2) = (ba, aa)$, $(x_3, y_3) = (baa, a)$, $(x_4, y_4) = (baaa, \epsilon)$

Sei A ein (unbekannter) endlicher Automat, der L akzeptiert. Da w vom Automaten A akzeptiert wird, gibt es für jedes Paar (x_i, y_i) einen Zustand q_i , so dass einer der Berechnungspfade den Automaten vom Startzustand aus auf Eingabe x_i in den Zustand q_i führt, und von q_i ausgehend auf der Eingabe y_i in einen der—möglicherweise mehreren—Endzustände. Führen Sie nun die Annahme zum Widerspruch, dass A drei Zustände besitzt und zugleich keine Wörter akzeptiert, die nicht in L sind.

Aufgabe 3:**Kontextfreie Sprachen**

Gegeben sei die Sprache $L = \{ ww \mid w \in \Sigma^* \}$ über dem Alphabet $\Sigma = \{a, b\}$

- a) Beweisen Sie mit Hilfe des Pumping Lemma für kontextfreie Sprachen, dass die Sprache nicht kontextfrei ist.

Hinweis:

Gehen Sie hierzu von einem Wort der Form $ww = a^n b^n a^n b^n$ aus, bei geeigneter Wahl von n .

Hinweis:

Das Pumping Lemma für kontextfreie Sprachen lautet wie folgt: Ist L kontextfrei, so existiert $n \geq 0$ sodass für alle $z \in L$ eine Zerlegung der Form $z = uvwxy$ existiert, derart, dass $|vx| \geq 1$ und $|vwx| \leq n$ und $uv^iwx^iy \in L$ für alle $i \geq 0$.

b) Begründen Sie, dass für jedes Wort $z \in \Sigma^*$, das nicht in L ist, einer der folgenden drei Fälle zutrifft:

- z hat ungerade Länge, oder
- z kann in der Form $z = uavbw$ geschrieben werden mit $|u| = m, |w| = n, |v| = m + n$ für geeignete $m, n \geq 0$, oder
- z kann in der Form $z = ubvaw$ geschrieben werden mit $|u| = m, |w| = n, |v| = m + n$ für geeignete $m, n \geq 0$.

Zum Beispiel ist $aabababbab \notin L$ und es passt der zweite Fall mit $u = a, v = baba, w = bab$, also $m = 1, n = 3$.

c) Begründen Sie nun, dass das Komplement $\Sigma^* \setminus L$ kontextfrei ist.