Grundlagen der theoretischen Informatik, SS 2001 — 1. Übungsblatt

Abgabe bis Montag, 30. April 2001

---

1. (0 Punkte) Sei Σ ein Alphabet mit k Elementen und $n \in \mathbb{N}$.

   (a) Wieviele Worte in Σ^* haben die Länge n?

   (b) Wieviele Palindrome in Σ^* haben die Länge n? (Ein Palindrom ist ein Wort, das von vorne und von hinten gelesen gleich ist.)

   (c) Wieviele Worte $w \in \{0,1\}^*$, die Binärdarstellung einer durch 5 teilbaren Zahl sind (führende Nullen sind erlaubt), haben die Länge n?

   (d) Wie ist es, wenn führende Nullen verboten sind?

   Jede Behauptung ist zu beweisen!

2. (0 Punkte) Beschreiben Sie die Sprache L, die durch den regulären Ausdruck

   $$0^*(0 + 10^*1)^*(\varepsilon + 0 + 00)$$

   gegeben ist. (Alle Worte in L haben eine sehr charakteristische Eigenschaft. Welche?) Beweisen Sie Ihre Behauptung! Gibt es einen einfacheren regulären Ausdruck für diese Sprache?

3. (6 Punkte)

   (a) (0 Punkte) Geben Sie einen regulären Ausdruck an, der genau die Worte w über Σ mit der folgenden Eigenschaft beschreibt: „w enthält keine zwei unmittelbar aufeinanderfolgenden gleichen Symbole," und zwar für $\Sigma_2 = \{a, b\}$, $\Sigma_3 = \{a, b, c\}$, und für $\Sigma_4 = \{a, b, c, d\}$. Beweisen Sie, dass Ihr Ausdruck wirklich diese Sprache beschreibt!

   (b) (6 Punkte) Lösen Sie dieselbe Aufgabe für die folgende Eigenschaft: „Jedes Symbol in w steht neben einem gleichen Symbol." (Zum Beispiel hat das Wort *aaabbcccaa* diese Eigenschaft.)

   (c) (Zusatzaufgabe, 2 Punkte) Wieviele Wörter der Länge n enthält die Sprache aus Aufgabe (b), für Σ_2 und Σ_3?

4. (6 Punkte)

   (a) Codieren Sie das Alphabet $\{a, b, \ldots, z\}$ durch Binärworte, so dass jedes codierte Wort wieder eindeutig decodierbar ist! (Eine ungeeignete Codierung wäre z.B. $a \mapsto 0, b \mapsto 01, c \mapsto 100$, etc. In diesem Fall könnte nämlich 0100 von *ac* oder von *baa* herstammen.) Zeigen Sie, dass Ihre Codierung die gewünschte Eigenschaft hat.

   (b) Das Alphabet $\Sigma = \{a, b\}$ sei durch $a \mapsto 0$ und $b \mapsto 010$ codiert. Zeigen Sie, dass diese Codierung eindeutig decodierbar ist!

5. (4 Punkte) Welche der folgenden Gleichungen gelten für alle regulären Ausdrücke?

   (a) $(AB)C = A(BC)$

   (b) $(A^*)^* = A^*$

   (c) $(A + B)^* = A^* + B^*$

   (d) $(A^*B^*)^* = (A + B)^*$

   Beweisen Sie Ihre Antworten.
   Zusatzfrage (2 Punkte): Wenn eine Beziehung nicht als Gleichung gilt, in welchen Fällen gilt dann eine Inklusionsbeziehung ($\subseteq$ oder $\supseteq$)?

6. (Zusatzfrage, 2 Punkte) Geben sie einen regulären Ausdruck an, der alle Wörter über $\Sigma = \{0, 1\}$ beschreibt, die 0110 nicht als Teilwort enthalten.

Grundlagen der theoretischen Informatik, SS 2001 — 2. Übungsblatt

Abgabe bis Montag, 7. Mai 2001

---

7. (0 Punkte) Entscheiden Sie, ob die folgenden regulären Ausdrücke jeweils dieselbe Sprache darstellen.

   (a) $\varepsilon^* = \varepsilon$ (b) $(01)^*0 = 0(10)^*$ (c) $\emptyset^* = \varepsilon + (0+\varepsilon)^*\emptyset$

   (d) $A^* = \varepsilon + A^*A$, wobei A ein beliebiger regulärer Ausdruck ist.

   (e) $(01^*)^* = (01^*0+0)^*$ (f) $(00)^* + (00000)^* + (000)^* = 00^*$

8. (6 Punkte) Eine *Kongruenzrelation* $\sim$ auf einer Menge Σ^* von Wörtern ist eine Äquivalenzrelation mit folgender zusätzlicher Eigenschaft:

$$a \sim b \land c \sim d \implies ac \sim bd.$$

   Welche der folgenden Relationen sind Kongruenzrelationen? (Beweisen Sie Ihre Antworten.)

   (a) (0 Punkte) $a \sim b \iff a$ und b haben die gleiche Anzahl von Nullen und die gleiche Anzahl von Einsen.

   (b) (0 Punkte) $a \sim b \iff a = \varepsilon$ oder $b = \varepsilon$ oder $a \neq \varepsilon \neq b$ und a und b haben denselben Anfangsbuchstaben.

   (c) (3 Punkte) $a \sim b \iff a = b = \varepsilon$ oder $a \neq \varepsilon \neq b$ und a und b haben denselben Anfangsbuchstaben.

   (d) (3 Punkte) $a \sim b \iff$ das Teilwort 01 kommt in a und b gleich häufig vor.

9. (5 Punkte) Für einen Zustand $q \in Q$ des DEA $M = (Q, \Sigma, \delta, q_0, F)$ sei die Sprache $L_q(M)$ so definiert:

$$L_q(M) = \{\, w \in \Sigma^* \mid \delta(q_0, w) = q \,\}$$

   Beweisen Sie: Wenn $w_1, w_2 \in L_q(M)$ sind, dann gilt für alle $x \in \Sigma^*$:

$$w_1x \in L(M) \iff w_2x \in L(M)$$

10. (5 Punkte) Konstruieren Sie, wenn möglich, DEA's, die folgende Sprachen über dem Alphabet $\{0,1\}$ akzeptieren.

   (a) (3 Punkte) $(0+1)^*010101(0+1)^*$

   (b) (2 Punkte) Alle Wörter, die das Teilwort 010101 *nicht* enthalten.

   (c) (0 Punkte) Alle Wörter, die das Teilwort 01, aber nicht das Teilwort 100 enthalten.

   (d) (0 Punkte) Alle Wörter, die gleich viele Nullen und Einsen enthalten.

   (e) (0 Punkte) Alle Wörter, die das Teilwort 010 *genau einmal* enthalten.

   (f) (Zusatzfrage, 3 Punkte) Beweisen Sie, dass jeder Automat, der Aufgabe (a) löst, mindestens sechs Zustände haben muss. [Hinweis: Sie können Aufgabe 9 verwenden.]

11. (0 Punkte) Konstruieren Sie, wenn möglich, DEA's, die folgende Sprachen akzeptieren.

   (a) Die durch 5 teilbaren Dezimalzahlen.

   (b) Die durch 9 teilbaren Dezimalzahlen.

   (c) Die durch 7 teilbaren Dezimalzahlen.

   Untersuchen Sie dabei jeweils beide Möglichkeiten der Reihenfolge, in der die Dezimalstellen der Zahl eingegeben werden können: mit der höchsten Stelle zuerst oder mit der Einerstelle zuerst.

Grundlagen der theoretischen Informatik, SS 2001 — 3. Übungsblatt

Abgabe bis Montag, 14. Mai 2001

12. (6 Punkte) Zeichnen Sie das Zustandsdiagramm des NEA $(\{a,b,c,d,e\},\{0,1\},\delta,\{a\},\{e\})$ mit $\delta = \{(a,0,a),(a,1,a),(a,0,b),(a,1,c),(b,0,b),(b,1,d),(c,0,d),(d,1,e),(e,0,e)\}$, und bestimmen Sie einen dazu äquivalenten DEA. Versuchen Sie, dabei mit möglichst wenigen Zuständen auszukommen.

13. (0 Punkte) Konstruieren Sie für den DEA $M = (\{a,b,c\},\{0,1\},\delta,a,\{b\})$ einen regulären Ausdruck, der die von M akzeptierte Sprache beschreibt. Die Übergangsfunktion δ ist in der nebenstehenden Tabelle angegeben.

δ	0	1
a	a	b
b	c	b
c	b	a

Wenden Sie den in der Vorlesung vorgestellten Algorithmus von Kleene auf die Zustände einmal in der Reihenfolge a, b, c und dann in der Reihenfolge c, a, b an. Vereinfachen Sie dabei die Zwischenergebnisse, so gut es geht. Versuchen Sie auch, „direkt“ durch Betrachten des Zustandsdiagramms und Nachdenken einen möglichst einfachen regulären Ausdruck zu finden.

14. (5 Punkte) Im Beweis für die Äquivalenz zwischen endlichen Automaten und regulären Ausdrücken wird für die von einem gegebenen deterministischen endlichen Automaten akzeptierte Sprache ein regulärer Ausdruck konstruiert. Funktioniert diese Konstruktion auch für *nichtdeterministische* endliche Automaten? Wie muss man die Konstruktion in diesem Falle anpassen, oder an welcher Stelle ist es wesentlich, dass der Automat deterministisch ist?

15. (0 Punkte) Betrachten Sie deterministische sowie nichtdeterministische endliche Automaten (mit oder ohne ε-Übergänge), die nur einen einzigen akzeptierenden Zustand haben dürfen. Sind diese Automaten gegenüber allgemeinen endlichen Automaten in ihrer Mächtigkeit eingeschränkt? Gibt es reguläre Sprachen, die von solchen Automaten nicht erkannt werden können?

16. (3 Punkte) Manchmal werden nichtdeterministische endliche Automaten so definiert, dass Sie nur einen einzigen Anfangszustand haben. Zeigen Sie, dass diese Automaten gegenüber NEA's mit beliebig vielen Anfangszuständen in ihrer Mächtigkeit nicht eingeschränkt sind und dieselbe Klasse von Sprachen akzeptieren.

    (Zusatzfrage, 2 Punkte) Geben Sie eine möglichst einfache und sparsame Methode an, wie man aus einem beliebigen NEA einen äquivalenten NEA mit nur einem Anfangszustand konstruiert.

17. (0 Punkte) Konstruieren Sie DEAs und NEAs für folgende Sprachen über $\{0,1\}$:

    (a) die Wörter, deren vierter Buchstabe eine 1 ist;

    (b) die Wörter, deren viertletzter Buchstabe eine 1 ist.

18. (0 Punkte) Gegeben seien zwei endliche Automaten M_1 und M_2, die die Sprachen L_1 beziehungsweise L_2 akzeptieren. Konstruieren Sie endliche Automaten, die die Sprachen (a) $L_1 \cup L_2$, (b) $L_1 \cdot L_2$, (c) $L_1 \cap L_2$ akzeptieren. Sie dürfen dabei wahlweise mit deterministischen oder mit nichtdeterministischen Automaten arbeiten, je nachdem, was Ihnen für die jeweilige Aufgabe günstiger erscheint.

Grundlagen der theoretischen Informatik, SS 2001 — 4. Übungsblatt

Abgabe bis Montag, 21. Mai 2001 (Abgabe in Dreiergruppen!)

---

19. (6 Punkte) Produkt von Automaten.
Wie kann man zu zwei gegebenen DEA M_1 und M_2 einen DEA konstruieren, der die Sprache $L(M_1) \cap L(M_2)$ akzeptiert? (Hinweis: Der neue Automat soll die Automaten M_1 und M_2 „gleichzeitig" simulieren; Welche Zustandsmenge muss er haben?)

20. (0 Punkte) Zeigen Sie: Wenn $L = \{ x \in \{0,1\}^* \mid x \text{ enthält gleich vielen Nullen und Einsen} \}$ regulär wäre, dann wäre auch die Sprache $L' = \{ 0^n 1^n \mid n \geq 0 \}$ regulär.

21. (4 Punkte) Konstruieren Sie einen NEA für die Sprache $\varepsilon + 0 + ((01^*0)^* + 10^*)^*$.

22. (0 Punkte) Konstruieren Sie einen NEA für die Sprache $(0+1)^*000(0+1+\varepsilon)^*$.

23. (6 Punkte) Bestimmen Sie den Minimalautomaten für den NEA aus Aufgabe 12.

24. (0 Punkte) Wie kann man für einen gegebenen DEA oder NEA M feststellen, ob $L(M) = \emptyset$ ist?

25. (0 Punkte) Beweisen Sie, dass die Klasse der regulären Sprachen abgeschlossen gegenüber *Umkehrung* ist: Wenn L regulär ist, dann ist auch L^R regulär. (L^R ist die Menge der von hinten gelesenen Wörter von L.)

Nächste Woche entfällt die Vorlesung, und es wird kein Übungsblatt ausgeteilt. Die Tutorien finden aber wie gewöhnlich statt.

Grundlagen der theoretischen Informatik, SS 2001 — 5. Übungsblatt

Abgabe bis Dienstag, 5. Juni 2001

26. (4 Punkte) Welche der folgenden Sprachen über $\Sigma = \{0, 1\}$ sind regulär? beweisen Sie Ihre Antworten.

    (a) Die Sprache $L_1 = \{ x \mid x = x^R \}$ der Palindrome
    (b) $L_2 = \{ 0^{i^2} \mid i \geq 1 \}$
    (c) $L_3 = \{ x \mid x$ enthält mindestens so viele Nullen wie Einsen $\}$
    (d) $L_4 = \{ x \mid$ jeder Präfix von x enthält mindestens so viele Nullen wie Einsen $\}$
    (e) (2 Punkte) $L_5 = \{ x \mid$ die Anzahl der Nullen von x unterscheidet sich um höchstens um 1 von der Anzahl der Einsen $\}$
    (f) (2 Punkte) $L_6 = \{ x \mid$ in jedem Präfix von x unterscheidet sich die Anzahl der Nullen höchstens um 1 von der Anzahl der Einsen $\}$

    Ein *Präfix* ist dabei ein Anfangsstück eines Wortes: x ist Präfix von y, wenn $y = xu$ für ein $u \in \Sigma^*$ ist.

27. (0 Punkte) Es seien Σ und Δ zwei Alphabete. Ein *Homomorphismus* $h\colon \Sigma^* \to \Delta^*$ ist durch eine beliebige Funktion $h\colon \Sigma \to \Delta^*$ gegeben, die folgendermaßen auf Σ^* erweitert wird:

    $$h(x_1 x_2 \ldots x_n) := h(x_1)h(x_2)\ldots h(x_n), \text{ für } n \geq 0,\ x_1, \ldots, x_n \in \Sigma$$

    Ein Homomorphismus ist durch folgende Eigenschaften charakterisiert:

    $$h(\varepsilon) = \varepsilon, \text{ und } h(x \cdot y) = h(x) \cdot h(y), \text{ für alle } x, y \in \Sigma^*$$

    Für eine *Sprache* $L \subseteq \Sigma^*$ bezeichnet $h(L)$ die Sprache

    $$h(L) := \{ h(x) \mid x \in L \}.$$

28. (4 Punkte) Welche der Sprachen $h_j(L_i)$ sind regulär, wenn L_i eine der Sprachen aus Aufgabe 26 ist und der Homomorphismus h_j folgendermaßen gegeben ist?

    (a) $h_1(0) = a$, $h_1(1) = b$, $\Delta = \{a, b\}$.
    (b) $h_2(0) = 1$, $h_2(1) = 1$.
    (c) $h_3(0) = 01$, $h_3(1) = \varepsilon$.

    Es gibt je 2 Punkte für die Sprachen $h_3(L_4)$ und $h_2(L_1)$.

29. (8 Punkte) Erweitern Sie das in der Vorlesung vorgestellte `lex`-Programm[1] zur Addition aller in der Eingabe vorkommenden Zahlen um folgende Funktionen:

    (a) Bei Eingabe des Zeichens `*` oder `-` wird die nächste Zahl nicht addiert, sondern multipliziert beziehungsweise subtrahiert. Bei Eingabe von `=` wird das bisherige Ergebnis ausgedruckt.
    (b) Es werden zusätzlich auch Kommentare im Stil von Haskell verstanden: Alles zwischen `--` und dem Ende der Zeile wird ignoriert.

    Die obigen Angaben lassen einige Fragen offen, zum Beispiel den Unterschied zwischen Subtraktion (`5 - 3`) und einem negativen Vorzeichen (`5 * -3`). Ergänzen Sie zunächst die obige Aufgabenspezifikation *in Worten*, sodass das Verhalten des Programmes eindeutig daraus hervorgeht. Ihr Programm sollte dann dieser Spezifikation entsprechen.

[1] `http://www.inf.fu-berlin.de/~rote/Lere/Grundlagen%20der%20theoretischen%20Informatik/flex/summe.flex`

Grundlagen der theoretischen Informatik, SS 2001 — 6. Übungsblatt

Abgabe bis Dienstag, 12. Juni 2001

30. (6 Punkte) Binäre Addition.
Konstruieren Sie eine Turingmaschine, die die Summe zweier Zahlen in Binärdarstellung berechnet. Bei Eingabe von $\mathrm{bin}(x)\#\mathrm{bin}(y)\$$ soll die Maschine mit der Ausgabe $\mathrm{bin}(x+y)\#\mathrm{bin}(y)\$$ und dem Kopf auf dem ersten Eingabesymbol halten. Dabei soll die Notation $\mathrm{bin}(x) \in \{0,1\}^*$ eine Darstellung von $x \in \mathbb{N}$ in Binärdarstellung darstellen. Der Automat soll am Ende in einen Zustand q_F übergehen und anhalten. Ein eventueller Bandinhalt rechts vom \$-Zeichen soll unverändert stehengelassen werden.

    Beschreiben Sie Ihren Algorithmus zunächst in Worten.

    Geben Sie die ersten 15 und die letzten 10 Konfigurationen Ihrer Turingmaschine bei Eingabe von `101#11$` an.

31. (0 Punkte) Multiplikation.
Konstruieren Sie eine Turingmaschine, die zwei „unäre“ Zahlen multipliziert. Bei Eingabe von $0^m 1^n$ soll die Ausgabe 0^{mn} berechnet werden.

32. (0 Punkte)

    (a) Entwerfen Sie eine Turingmaschine, die die Sprache $\{\, 0^n 1^n 0^n \mid n \geq 0 \,\}$ akzeptiert.

    (b) Entwerfen Sie eine Turingmaschine für dieselbe Sprache, die mit dem Bandalphabet $\Gamma = \{0, 1, B\}$ auskommt.
    (Hinweis: Sie können ein größeres Bandalphabet „simulieren“, indem Sie es „binär kodieren“. Dabei wird jeweils eine feste Anzahl von Feldern des Bandes zu einer logischen Einheit zusammengefasst.)

33. (5 Punkte) Definieren Sie formal die Nachfolgerrelation $xqy \vdash x'q'y'$ für Konfigurationen einer Turingmaschine. ($q, q' \in Q$, $x \in \{\varepsilon\} \cup (\Gamma - \{B\})\Gamma^*$, $y \in \{\varepsilon\} \cup \Gamma^*(\Gamma - \{B\})$.)

34. (0 Punkte) Speicherzugriff (Tabellenzugriff) über Adressen.

    (a) Lesen aus einer Tabelle.
    Bei Eingabe von
    $$x = \#x_1 | y_1 \#x_2 | y_2 \# \ldots \#x_n | y_n \$ x_0$$
    mit $n \geq 0$, $x_k, y_k \in \{0,1\}^*$ soll die Ausgabe $x|y$ berechnet werden. Dabei ist $y = y_i$ für den größten Index $i \leq n$ mit $x_i = x_0$, und $y = 0$, falls kein solches i existiert.

    (b) Löschen aus einer Tabelle.
    Bei den gleichen Angaben wie oben soll die Ausgabe
    $$x = \#x_1 | y_1 \# \ldots \#x_{i-1} | y_{i-1} \#x_{i+1} | y_{i+1} \# \ldots \#x_n | y_n \$ x_0$$
    berechnet werden. Wenn kein passender Index i existiert, soll die Eingabe unverändert gelassen werden.

35. (4 Punkte) Beweisen oder widerlegen Sie:

    (a) (4 Punkte) Das Komplement einer entscheidbaren Sprache ist entscheidbar.

    (b) (0 Punkte) Die Vereinigung zweier entscheidbarer Sprache ist entscheidbar.

    (c) (0 Punkte) Das Komplement einer unentscheidbaren Sprache ist unentscheidbar.

    (d) (0 Punkte) Die Vereinigung zweier unentscheidbarer Sprache ist unentscheidbar.

Grundlagen der theoretischen Informatik, SS 2001 — 7. Übungsblatt

Abgabe bis Montag, 18. Juni 2001

---

36. (6 Punkte) Beweisen Sie:
    (a) (6 Punkte) Wenn L_1 und L_2 entscheidbar sind, dann ist auch L_1L_2 entscheidbar.
    (b) Wenn L_1 und L_2 rekursiv aufzählbar sind, dann ist auch L_1L_2 rekursiv aufzählbar.

37. (0 Punkte) Eine Sprache L ist genau dann entscheidbar, wenn sowohl die Sprache L als auch ihr Komplement $\bar{L}$ rekursiv aufzählbar ist.

38. (0 Punkte)
    (a) Zeigen Sie, dass die Sprache $\bar{D}$ rekursiv aufzählbar ist.
    (b) Zeigen Sie, dass die Sprache $0D \cup 1\bar{D}$ nicht rekursiv aufzählbar ist.

    ($D = \{\, w_i \mid M_i \text{ akzeptiert } w_i \text{ nicht} \,\}$ ist die Diagonalsprache und $\bar{D}$ ihr Komplement.)

39. (2 Zusatzpunkte) Eine andere Konstruktion einer unentscheidbaren Sprache.

    $$U' := \{\, \langle M\rangle w \mid M \text{ akzeptiert das Wort } \langle M\rangle w \text{ nicht} \,\}$$

    (Der Beweis geht ähnlich wie bei der Diagonalsprache D.)

40. (6 Punkte) Welche der folgenden Sprachen sind rekursiv, welche sind rekursiv aufzählbar?
    (a) (3 Punkte) $L_1 = \{\, 0^k\langle M\rangle w \mid M \text{ akzeptiert } w \text{ nach höchstens } k \text{ Schritten.} \,\}$
    (b) $L_2 = \{\, 0^k\langle M\rangle w \mid M \text{ hat } w \text{ nach } k \text{ Schritten noch nicht akzeptiert.} \,\}$
    (c) $L_3 = \Sigma^* - (L_1 \cup L_2)$
    (d) (3 Punkte) $L_4 = \{\, 0^k\langle M\rangle w \mid M \text{ akzeptiert } w \text{ nach mehr als } k \text{ Schritten.} \,\}$

41. (4 Punkte) Beiweisen Sie, dass die Sprache $L = \{\, 0^i \mid i \geq 0, \text{ das Teilwort } 0^i123 \text{ kommt in der Dezimaldarstellung von } \pi \text{ unendlich oft vor} \,\}$ entscheidbar ist. (Sie müssen dazu nichts über π wissen, nur logisch überlegen können. Wie kann L aussehen?)

42. (0 Punkte) Welche der folgenden Aussagen sind wahr, und welche sind falsch? Welche Aussagen sind trivial, und welche sind absurd?
    (a) Jede Teilmenge einer rekursiven Sprache ist rekursiv.
    (b) Jede Teilmenge einer rekursiv aufzählbaren Sprache ist rekursiv aufzählbar.
    (c) Jede reguläre Sprache ist rekursiv aufzählbar.
    (d) Jede Teilmenge einer regulären Sprache ist regulär.

Weil noch etwas Platz ist, seien hier noch einmal die Kriterien für die Erlangung eines Übungsscheines in Erinnerung gerufen.[1]

1. 60 % der Gesamtpunktezahl der Übungen. (Die Übungszettel können in Dreiergruppen oder in Zweiergruppen abgegeben werden.)
2. Bei jedem Übungszettel bis auf einen müssen mindestens 30 % der Punkte erreicht werden.
3. 50 % der Punkte bei der Zwischenklausur und 50 % der Punkte bei der Abschlussklausur (oder bei der entsprechenden Nachklausur).
4. Aktive Teilnahme an den wöchentlichen Tutorien und *einmaliges Vorrechnen*

Der Schein wird benotet. Die Note ergibt sich aus den Klausuren.

[1] `http://www.inf.fu-berlin.de/~rote/Lere/Grundlagen%20der%20theoretischen%20Informatik/scheinkrit.html`

Grundlagen der theoretischen Informatik, SS 2001 — 8. Übungsblatt

Abgabe bis Montag, 25. Juni 2001

44. (5 Punkte) Beweisen Sie:
   (a) (0 Punkte) Wenn es eine Turingmaschine gibt, die die Wörter einer Sprache in beliebiger Reihenfolge, möglicherweise mit Wiederholungen, auf ein Ausgabeband schreibt, dann ist diese Sprache rekursiv aufzählbar.
   (b) (5 Punkte) Eine Sprache ist genau dann rekursiv, wenn es eine Turingmaschine gibt, die die Wörter der Sprache der Länge nach sortiert auf ein Ausgabeband schreibt (Wörter gleicher Länge können in beliebiger Reihenfolge, möglicherweise mit Wiederholungen, ausgegeben werden.)

45. (0 Punkte) Zeigen Sie:
   (a) Die rekursiven Sprachen sind unter Differenz abgeschlossen. (Das heißt: Mit L_1 und L_2 ist auch $L_1 - L_2$ immer rekursiv.)
   (b) Die rekursiv aufzählbaren Sprachen sind unter Differenz nicht abgeschlossen.

46. (5 Punkte) Bestimmen Sie die Sprache, die von der Grammatik mit dem Startsymbol S, dem Terminalalphabet $\{a\}$ und den folgenden Regeln erzeugt wird. Begründen Sie Ihre Antwort, indem Sie den Ableitungsprozess in Worten beschreiben.

$$\begin{array}{ll} S \to ACaB & aD \to Da \\ Ca \to aaC & AD \to AC \\ CB \to DB & aE \to Ea \\ CB \to E & AE \to \varepsilon \end{array}$$

47. (0 Punkte) Bestimmen Sie kontextsensitive Grammatiken für folgende Sprachen.
   (a) $L_1 = \{\, ww \mid w \in \{0,1\}^+ \,\}$
   (b) $L_2 = \{\, 0^{i^2} \mid i \geq 1 \,\}$
   (c) $L_3 = \{\, 0^i \mid i \geq 2,\ i \text{ ist keine Primzahl} \}$
   (d) $L_4 = \{\, 0^i 1^i 2^i \mid i \geq 1 \,\}$

48. (6 Punkte) Geben Sie eine *eindeutige* kontextfreie Grammatik für arithmetische Ausdrücke über dem Alphabet $\{+, -, *, \div, \uparrow, (,), Z\}$ an. Die Grammatik soll die Ausdrücke gemäß den üblichen mathematischen Regeln analysieren, wobei der Operator $\uparrow$ für die Exponentiation höchste Priorität hat und rechtsassoziativ ist. Es soll also zum Beispiel $Z \uparrow Z \uparrow Z$ wie $Z \uparrow (Z \uparrow Z)$ interpretiert werden, und $Z - Z \div Z * Z - Z * Z$ wie $(Z - ((Z \div Z) * Z)) - (Z * Z)$.

49. (0 Punkte) Betrachten Sie die kontextfreie Grammatik G mit den Regeln

$$\begin{array}{l} S \to ScS \mid A \\ A \to aAb \mid \varepsilon \end{array}$$

   (a) Beschreiben Sie die erzeugte Sprache, und zeigen Sie, dass G mehrdeutig ist.
   (b) Geben Sie eine eindeutige kontextfreie Grammatik an, die dieselbe Sprache erzeugt.

50. (0 Punkte) Zeigen Sie, dass die Grammatiken
   (a) $S \to SS \mid (S) \mid \varepsilon$
   (b) $S \to (S)S \mid \varepsilon$

   die gleiche Sprache erzeugen und dass (a) mehrdeutig und (b) eindeutig ist.

51. (0 Punkte) Bringen Sie die Grammatik aus Aufgabe 49 und Ihre Antwort aus Aufgabe 49b in Chomsky-Normalform.

Grundlagen der theoretischen Informatik, SS 2001 — 9. Übungsblatt

Abgabe bis Montag, 2. Juli 2001

---

52. (a) (0 Punkte) Geben Sie eine kontextfreie Grammatik für arithmetische Ausdrücke über dem Alphabet $\{+, *, (,), Z\}$ an. Dabei sollen nur Ausdrücke *ohne überflüssige Klammern* erzeugt werden, d. h., solche Audrücke, die nach den üblichen mathematischen Vorrangregeln einen anderen Wert bekommen, wenn man Klammern weglässt. Zum Beispiel sind die Klammern in $Z * (Z + Z)$ notwendig, weil man den Ausdruck ohne Klammern $Z * Z + Z$ als $(Z * Z) + Z$ lesen würden. Aus diesem Grund sind auch die Klammern im Ausdruck $(Z * Z) + Z$ überflüssig (redundant).
    (b) (0 Punkte) Lösen Sie die Aufgabe mit dem erweiterten Alphabet $\{+, -, *, \div, (,), Z\}$.

53. (0 Punkte) Beweisen Sie, dass man zu jeder kontextsensitiven Grammatik eine äquivalente Grammatik finden kann, deren Regeln von der Gestalt $S \to \varepsilon$ oder von folgender Gestalt sind.
$$\alpha_1 A \alpha_2 \to \alpha_1 \beta \alpha_2, \text{ mit } A \in V,\ \alpha_1, \alpha_2, \beta \in (T \cup V - \{S\})^*,\ \beta \neq \varepsilon$$

54. (6 Punkte) Welche der folgenden Sprachen sind kontextfrei?
    (a) (4 Punkte) $L_1 = \{\, 0^i 1^{i^2} \mid i \geq 1 \,\}$
    (b) (2 Punkte) $L_2 = \{\, 0^i 1^j \mid i \geq j \,\}$
    (c) Die Sprache der Doppelwörter $L_3 = \{\, ww \mid w \in \{0,1\}^* \,\}$
    (d) $L_4 = \{\, 0^i 1^j 0^k 1^l \mid i + j = k + l \,\}$, $\qquad L_5 = \{\, 0^i 1^j 0^k 1^l \mid i + k = j + l \,\}$
    (e) $L_6 = \{\, w \in \{0,1\}^* \mid w$ enthält doppelt so viele Nullen wie Einsen $\}$

55. (5 Punkte) Beweisen Sie, dass die Grammatik mit den Regeln $S \to aS \mid aSbS \mid \varepsilon$ genau die Wörter über $\{a, b\}$ erzeugt, in denen jeder Präfix mindestens so viele a's wie b's enthält.

56. (0 Punkte) Die Grammatik
$$\begin{aligned} &S \to \textbf{if } (B)\ S \mid \textbf{if } (B)\ S \textbf{ else } S \mid \text{print}(0); \mid \text{print}(1); \mid \text{print}(2);\\ &B \to \text{true} \mid \text{false}\end{aligned}$$
ist mehrdeutig, wie das Beispiel

if (true) **if** (false) **if** (false) print(0); **else if** (true) print(0); **else** print(1); **else** print(2);

zeigt. Die Mehrdeutigkeit kann durch die gängige Regel aufgelöst werden, dass eine **else**-Klausel immer dem *letzten* **if** zugeordnet wird, dem noch keine **else**-Klausel zugeordnet wurde. Nach dieser Regel würde der obige Ausdruck 2 drucken, wenn man ihn als Programm auffasst. (Welche Werte könnte das Programm sonst noch drucken, wenn man alle Mehrdeutigkeiten berücksichtigt?) Geben Sie eine eindeutige Grammatik an, die dieser Regel entspricht.

57. (5 Punkte) Bleibt eine eindeutige Grammatik bei der Transformation in Chomsky-Normalform eindeutig? Beweisen Sie Ihre Antwort.

58. (0 Punkte) G sei eine Grammatik in Chomsky-Normalform, die eine endliche Sprache erzeugt. Finden Sie eine Schranke für das längste Wort in $L(G)$.

59. (0 Punkte) Wie lang kann die Ableitung eines Wortes w bei einer Grammatik in Chomsky-Normalform höchstens sein?

Grundlagen der theoretischen Informatik, SS 2001 — 10. Übungsblatt

Letztes Pflichtübungsblatt — Abgabe bis Montag, 9. Juli 2001

60. (0 Punkte) Schreiben Sie eine kontextfreie Grammatik für die regulären Ausdrücke über dem Alphabet $\{0,1\}$. Sie können annehmen, dass die Ausdrücke voll geklammert sind, entsprechend der ursprünglichen Definition regulärer Ausdrücke, zum Beispiel

$$(((0+(1)^*))\cdot((0+(0\cdot 1))+\varepsilon))$$

61. (0 Punkte) Kontextfreie Sprachen sind unter Umkehrung (Spiegelung) abgeschlossen.

62. (4 Punkte) Entfernen Sie die überflüssigen Variablen und Regeln aus dieser Grammatik:

$$\begin{array}{lll} S \to AB \mid CA & B \to BC \mid AB \mid BDE & D \to ADEba \mid ACA \\ A \to a & C \to aB \mid c & E \to DD \mid BA \end{array}$$

63. (7 Punkte) Überprüfen Sie mit dem CYK-Algorithmus, ob das Wort $w = cdccd$ von der folgenden Grammatik erzeugt wird, und geben Sie gegebenfalls eine Ableitung für w an.

$$\begin{array}{lll} S \to AB \mid CD \mid AS & B \to SB \mid d & D \to CD \mid d \\ A \to AS \mid DC & C \to CA \mid c & \end{array}$$

64. (0 Punkte) Übersetzen Sie den folgenden vereinfachten Ausschnitt aus der Syntax der Programmiersprache Java aus der EBNF (extended Backus-Naur-Form) in entsprechende kontextfreie Regeln:

TryBlock	::=	**try** Block Catches | **try** Block [Catches] **finally** Block
Catches	::=	CatchClause { CatchClause }
CatchClause	::=	**catch (** Exception Identifier **)** Block
Exception	::=	Identifier

Die Terminalsymbole sind hier fett gedruckt. Sie können die Variablen durch passende einbuchstabige Namen abkürzen. (Wozu dient eigentlich die Variable „Exception", die man doch genausogut mit der Regel „Exception ::= Identifier" eliminieren könnte?)

65. (0 Punkte) Kann man für eine gegebene kontextfreie Grammatik G und eine gegebene reguläre Sprache L_1 entscheiden, ob $L(G) = L_1$ ist?

66. (5 Punkte) Betrachten Sie die deterministischen Zweiweg-Kellerautomaten mit dem Eingabealphabet $\Sigma = \{a\}$, dem Kelleralphabet $\Gamma = \{0, 1, Z_0\}$ und der Übergangsfunktion $\delta\colon Q \times \Gamma \times (\Sigma \cup \{\lozenge, \$\}) \to Q \times \Gamma^* \times \{L, N, R\}$.

$\delta(q_0, \gamma, a) = (q_0, 0\gamma, R)$, für $\gamma \in \Gamma$.
$\delta(q_0, \gamma, \$) = (q_1, \gamma, N)$, für $\gamma \in \Gamma$.
$\delta(q_1, 1, x) = (q_1, \varepsilon, L)$, für $x = 0$ oder $x = \$$.
$\delta(q_1, 0, x) = (q_0, 1, N)$, für $x = 0$ oder $x = \$$.
$\delta(q_1, Z_0, x) = (q_1, \varepsilon, N)$, für $x = 0$ oder $x = \$$.

Die übrigen Werte von δ (für $x = \lozenge$) sind beliebig.

   (a) (0 Punkte) Beschreiben Sie das Verhalten des Automaten bei Eingabe des Wortes a^n, für $n = 0, 1, 2, 3$ und für allgemeines n.
   (b) (5 Punkte) Bestimmen Sie die Entladefunktion $E\colon Q \times \{0, \ldots, n+1\} \times \Gamma \to Q \times \{0, \ldots, n+1\}$ bei Eingabe des Wortes a^n.

67. (0 Punkte) Wie kann man zu einem regulären Ausdruck einen regulären Ausdruck für die komplementäre Sprache finden?

Grundlagen der theoretischen Informatik, SS 2001 — 11. Übungsblatt

ergänzende Zusatzübungen zur Wiederholung und Vertiefung

freiwillige Abgabe bis Montag, 16. Juli 2001

---

68. (a) (3 Zusatzpunkte) Beschreiben Sie formal, wie man einen nichtdeterministischen Kellerautomaten, der mit akzeptierenden Zuständen arbeitet, in einen äquivalenten Kellerautmaten umwandelt, der mit leerem Keller akzeptiert.
    (b) Beschreiben Sie die Transformation in die umgekehrte Richtung.
    (c) Lösen Sie die beiden obigen Aufgaben auch für deterministische Zweiweg-Kellerautomaten.

69. (6 Zusatzpunkte) Beweisen Sie, dass die kontextfreien Sprachen abgeschlossen gegenüber dem Durchschnitt mit regulären Sprachen sind. (Hinweis: Wählen Sie ein geeignetes Maschinenmodell für die kontextfreien Sprachen und verfahren Sie ähnlich wie bei Aufgabe 19.)

70. Beweisen Sie, dass die kontextfreien Sprachen abgeschlossen gegenüber der Vereinigung mit regulären Sprachen sind.

71. (4 Zusatzpunkte) Konstruieren Sie (a) eine Grammatik, (b) einen Kellerautomaten für die Sprache $\{\, ww^R \mid w \in \{0,1\}^* \,\}$.

72. Konstruieren Sie einen deterministischen Kellerautomaten für die Sprache
$$\left\{\, w\#w^R \mid w \in \{-1,1\}^* \,\right\}.$$

73. Konstruieren Sie einen deterministischen Kellerautomaten für die Sprache der Wörter, die gleich viele Nullen und Einsen enthalten.

74. Beweisen Sie, dass die Sprachen $\{\, a^i b^i c^j \mid i,j \geq 0 \,\}$ und $\{\, a^i b^j c^j \mid i,j \geq 0 \,\}$ deterministische kontextfreie Sprachen sind. Zeigen Sie mit Hilfe dieser Sprachen, dass die deterministischen kontextfreien Sprachen nicht gegenüber dem Durchschnit abgeschlossen sind.

75. Beweisen Sie: Wenn eine Sprache von einem deterministischen Kellerautomaten (mit akzeptierenden Zuständen) akzeptiert wird, dann gibt es eine eindeutige kontextfreie Sprache für diese Grammatik.

76. Sind die Sprachen, die von deterministischen Zweiweg-Kellerautomaten akzeptiert werden, abgeschlossen gegenüber Umkehrung (Spiegelbild)?

77. (Schwierig, 5 Zusatzpunkte) Sind die deterministischen kontextfreien Sprachen abgeschlossen gegenüber Umkehrung?

78. (a) Wandeln Sie den folgenden Kellerautomaten in einen äquivalenten Kellerautomaten mit nur einem einzigen Zustand um. $\delta = \{(q_0, 0, \gamma, q_1, Z_0\gamma), (q_0, 1, \gamma, q_1, Z_1\gamma), (q_1, 0, \gamma, q_0, Z_0\gamma), (q_1, 1, \gamma, q_1, Z_1Z_1), (q_0, 1, Z_1, q_0, \varepsilon), (q_0, 0, Z_0, q_0, \varepsilon)\}$, wobei γ jeweils für ein beliebiges Kellersymbol $\gamma \in \Gamma = \{Z_0, Z_1\}$ steht.
    (b) Konstruieren Sie eine entsprechende kontextfreie Grammatik.

79. Welche Beispiele von Typ-0-Sprachen kennen Sie, die keine Typ-1-Sprachen sind? Welche Beispiele von Typ-1-Sprachen kennen Sie, die keine Typ-2-Sprachen sind? Welche Beispiele von Typ-2-Sprachen kennen Sie, die keine Typ-3-Sprachen sind?

    Welche Beispiele von kontextfreien Sprachen kennen Sie, die keine deterministischen kontextfreien Sprachen sind?