Grundlagen der theoretischen Informatik, SS 2005 — 7. Übungsblatt

Abgabe bis Montag, 13. Juni 2005, 10:15 Uhr

---

49. Ist es entscheidbar, ob eine Turingmaschine das leere Wort akzeptiert?

50. Ist es entscheidbar, ob eine Turingmaschine alle Wörter (also Σ^*) akzeptiert?

51. (a) Wie kann man das Post'sche Korrespondenzproblem über einem einelementigen Alphabet lösen?

    (b) Beweisen Sie, dass das Post'sche Korrespondenzproblem über einem zweielementigen Alphabet unentscheidbar ist.

52. Lösen Sie das Post'sche Korrespondenzproblem für folgende Eingaben:

    (a) $x_1, \ldots, x_5 = abb, a, bab, baba, aba$; $y_1, \ldots, y_5 = bbab, aa, ab, aa, a$.

    (b) $x_1, \ldots, x_5 = a, baa, abab, aabb, abb$; $y_1, \ldots, y_5 = ab, ba, bbaa, bbba, bba$.

    (c) $x_1, \ldots, x_{10} = d, de, deh, deh, deh, deha, deha, wa, wa, wah$; $y_1, \ldots, y_{10} = add, e, de, d, deha, eh, d, w, wah, wa$; Welches ist die kürzeste und die zweitkürzeste Lösung?

    (d) (zum Tüfteln für alle, denen das Post'sche Korrespondenzproblem zu leicht vorkommt) $x_1, \ldots, x_4 = 001, 01, 01, 10$; $y_1, \ldots, y_4 = 0, 011, 101, 001$.

53. Zeigen Sie, dass es unendlich viele Lösungen gibt, wenn das Post'sche Korrespondenzproblem eine Lösung hat.

54. (10 Punkte) Algorithmische Reduzierbarkeit

    (a) (5 Punkte) Die Reduzierbarkeit zwischen Problemen ist eine transitive Relation: Aus $A < B$ und $B < C$ folgt $A < C$.

    (b) Ist die Reduzierbarkeit eine Ordnungsrelation?

    (c) (5 Punkte) Beweisen Sie: Wenn A ein beliebiges entscheidbares Problem ist, so ist es auf das Halteproblem H reduzierbar: $A < H$.

55. Wie kann man jede Turingmaschine *ohne Zeitverlust* durch eine Turingmaschine simulieren, die sich nie über das erste Eingabezeichen nach links hinausbewegt?

56. (5 Punkte) Wenn eine Turingmaschine $(Q, \Sigma, \Gamma, \delta, q_0, B, F)$ ein Wort w in t Schritten akzeptiert und sie dabei nur die ersten s Felder des Bandes besucht und sich nie über das erste Eingabezeichen nach links hinausbewegt, dann ist $t \leq |Q| \cdot s \cdot |\Gamma|^s$.

57. Betrachten Sie die „Fleißige-Biber-Funktion" für Programme in der Programmiersprache SCHLEIFE aus Aufgabe 1d, eingeschränkt auf Programme, in denen die Schleifenanweisungen höchstens bis zu einer vorgegebenen Tiefe k ineinander geschachtelt sein dürfen.

    (a) Beweisen Sie, dass diese Funktion berechenbar ist.

    (b) Beweisen Sie, dass diese Funktion für ein festes $k \geq 1$ nicht mit einem SCHLEIFE-Programm der Schachtelungstiefe k berechnet werden kann.

    (c) Was ist die Fleißige-Biber-Funktion für $k = 0$ (Programme ohne Schleife)? Was ist das kürzeste Programm zur Berechnung dieser Funktion?

58. Entwerfen Sie eine kontextsensitive Grammatik für die Sprache $\{\, a^n b^n c^n \mid n \geq 1 \,\}$

59. Der erste Satz dieser Aufgabe ist ein Beispiel eines grammatikalisch korrekten Satzes. Analysieren Sie diesen Satz gemäß der nachstehenden rudimentären Grammatik:

> S $\rightarrow$ NG VG, NG $\rightarrow$ NG Att, NG $\rightarrow$ Art NG', NG $\rightarrow$ NG', NG' $\rightarrow$ Adj NG', NG' $\rightarrow$ N, NG $\rightarrow$ Pro, Att $\rightarrow$ NG, Att $\rightarrow$ Präp NG, VG $\rightarrow$ VG Erg, VG $\rightarrow$ V Obj, VG $\rightarrow$ V, VG $\rightarrow$ V Pr, Obj $\rightarrow$ NG, Erg $\rightarrow$ Präp NG, Pr $\rightarrow$ Adj, Pr $\rightarrow$ NG, Adj $\rightarrow$ Adv Adj.

(S = Satz, NG = Nominalgruppe, NG' = Nominalgruppe ohne Artikel, VG = Verbgruppe, Att = Attribut, Art = Artikel, Adj = Adjektiv, Adv = Adverb, N = Hauptwort (Nomen), Pro = Pronomen, Obj = Objekt, Erg = Ergänzung (Modalobjekt), Pr = Prädikat.)