Universität Karlsruhe (TH)

Institut für Programmstrukturen und Datenorganisation (IPD)

Informatik I WS 2003/04 http://www.infoeins.de

Dozent: Prof. Dr.rer.nat. G. Goos goos@ipd.info.uni-karlsruhe.de
Übungsleiter: Tom Gelhausen gelhausen@fzi.de

Übungsblatt 2 - (60T / 0P)

Semi-Thue-Systeme, Markov, Chomsky

Ausgabe: 24.10.2003
Abgabe: 31.10.2003
13:30 Uhr

Einwurf im Keller des Informatik-Hauptbaus (Geb. 50.34)

Aufgabe 1: Semi-Thue-Systeme und Markov-Algorithmen (24T)

1.1 Produkt-Vorzeichen (2T)

Entwerfen Sie ein Semi-Thue-System, das als Eingabe die Vorzeichen einer Multiplikationskette erhält und als Ausgabe das Vorzeichen des Produktes liefert.
(Beispiel: „(-3) · (+4) · (-3)“ bedeutet „- + -“ als Eingabe, die Ausgabe soll in diesem Falle „+“ sein)

1.2 Einerkomplement (5T)

Entwerfen Sie nun ein Semi-Thue-System, das zu einer gegebenen Bitfolge das Einerkomplement bestimmt. (Hinweis: Sie dürfen sich zur Lösung dieser Aufgabe Hilfsmarkierungen in das Eingabewort einbauen.) (3T) Kann diese Aufgabe ohne Hilfsmarkierungen gelöst werden (Begründung)? (1T) Führen Sie den Algorithmus für die Bitfolge „101010“ vor. (Fügen Sie ggf. die benötigten Hilfsmarkierungen ein, um diese Bitfolge in Ihr Eingabewort zu verwandeln.) (1T)

1.3 Subtrahieren (4T)

Formulieren Sie einen Markov-Algorithmus zum Subtrahieren von Strichfolgen (z.B. ||| - |||| = -|). Beschreiben Sie auch jeweils kurz, was die jeweilige Regel bewirkt.

1.4 Vereinigung von Mengen (6T)

Formulieren Sie ein Semi-Thue-System zur Vereinigung mehrerer Mengen, wobei Σ = {„a“, „b“, „c“ und „$\cup$“}. Jeder Buchstabe darf in der Ergebnismenge nur einmal vorkommen (z.B. ab $\cup$ ac = abc). (3T) Beschreiben Sie auch jeweils kurz, was die jeweilige Regel bewirkt und motivieren Sie, wie/warum Ihr Semi-Thue-System funktioniert. (3T)

1.5 Durchschnitt von Mengen (7T)

Formulieren Sie einen Markov-Algorithmus zum Schneiden zweier Mengen, wobei Σ = {„a“, „b“, „c“ und „$\cap$“}. Jeder Buchstabe darf in der Ergebnismenge nur einmal vorkommen (z.B. ab $\cap$ ac = a). (4T) Beschreiben Sie auch jeweils kurz, was die jeweilige Regel bewirkt. (3T)

2. Kaffeedosen-Spiel (14T)

2.1 Anwendung (2T)

Schauen Sie sich Scholtens Kaffeedosen-Spiel aus der Vorlesung noch einmal an und führen Sie dann folgende Spiele durch (Sie können annehmen, einen genügend großen Vorrat an schwarzen Bohnen außerhalb der Kaffeedose zur Verfügung zu haben.) Machen Sie dabei deutlich, welche Regel Sie an welcher Stelle anwenden:

a. [Schwarz, Weiß, Weiß, Schwarz, Schwarz, Schwarz, Schwarz]
b. [Weiß, Schwarz, Schwarz, Weiß, Schwarz, Weiß, Schwarz]

2.2 Weiterführende Überlegungen (12T)

Beweisen Sie den Satz aus der Vorlesung bezüglich Scholtens Kaffeedosen-Spiel:

> *„Das Ergebnis (Farbe der letzten Bohne) ist unabhängig von der Reihenfolge der Anordnung und der Regelauswahl“*

Hinweis: Beweisen Sie zunächst die Eigenschaften einer einzelnen Ableitung und danach die Gültigkeit dieser Eigenschaften für Ableitungsketten beliebiger Länge!

3. Chomsky-Grammatiken (22T)

3.1 Grammatik 1 (5T)

Gegeben sei die folgende Grammatik:

$$
\begin{array}{lllll}
G_1 & = \{ \sum, N, P, S \} & & & \\
N & = \{ S, I, N \} & & & \\
\sum & = \{ n, i \} & & & \\
P & = \{ & S & \mapsto & N, \\
 & & I & \mapsto & iN \mid i, \\
 & & N & \mapsto & nI \, \}
\end{array}
$$

Geben Sie den einschränkensten Chomsky Typ an (1T) und begründen Sie ihre Entscheidung (2T). Geben Sie außerdem die erzeugte Sprache an (2T).

3.2 Grammatik 2 (5T)

Gegeben sei die folgende Grammatik:

$$
\begin{array}{lllll}
G_2 & = \{ \sum, N, P, S \} & & & \\
N & = \{ S, A, B \} & & & \\
\sum & = \{ 0, 1 \} & & & \\
P & = \{ & S & \mapsto & 0B \mid 1A, \\
 & & A & \mapsto & 0 \mid 0S, \\
 & & B & \mapsto & 1 \mid 1S \, \}
\end{array}
$$

Geben Sie den einschränkensten Chomsky Typ an (1T) und begründen Sie ihre Entscheidung (2T). Geben Sie außerdem die erzeugte Sprache an (2T).

3.3 Klassifikation von Grammatiken (8T)

Geben Sie zu den folgenden Grammatiken den einschränkensten Chomsky Typ an (mit Begründung).

a. G_a = { ∑, N, P, S } (2T)
N = { S, A, B }
∑ = { 0, 1 }
P = { S ↦ 0SA | 0B,
BA ↦ 1B0,
0A ↦ A1,
B ↦ 10 }

b. G_b = { ∑, N, P, S } (2T)
N = { S }
∑ = { 0, 1 }
P = { S ↦ 01 | 0S1 }

c. G_c = { ∑, N, P, S } (2T)
N = { S, O, Z }
∑ = { 0, 1 }
P = { S ↦ 0B | 1A,
A ↦ 0 | 0S | 1AA,
B ↦ 1 | 1S | 0BB }

d. G_e = { ∑, N, P, S } (2T)
N = { S, A, B, C, D }
∑ = { 0, 1, 2, 3 }
P = { S ↦ ABCD,
CD ↦ E,
E ↦ D | ε,
A ↦ 0 | 0A,
B ↦ 012 | 0B12,
D ↦ 3,
B ↦ ε }

3.4 Klassifikation von Produktionen (4T)

Von welchem Typ ist eine Grammatik höchstens, die folgende Produktionen enthält (je 1T):

a. caABa ↦ caAcAa
b. caABa ↦ caBAa
c. B ↦ cC
d. cAd ↦ da