Universität Karlsruhe (TH) – Fakultät für Informatik
Institut für Logik, Komplexität und Deduktionssysteme
Lehrstuhl Prof. Dr. P. Deussen

Informatik III – Prof. Dr. P. Deussen im Wintersemester 2001/2002

C. Bierwald, Th. Käufl, F. Schuster   http://i10www.ira.uka.de/lehre/info3-winter2001   info3@ira.uka.de

Übungsblatt 10 – Lösungsvorschläge

11.1.2002

Aufgabe 1 Abschlußeigenschaften (Bierwald)

Eine Familie $\mathcal{F}$ von Mengen heißt abgeschlossen bezüglich Schnitt, Vereinigung oder Komplementbildung, falls für alle $A, B \in \mathcal{F}$ auch $A \cap B \in \mathcal{F}$, $A \cup B \in \mathcal{F}$ oder $\overline{A} \in \mathcal{F}$ gilt. Füllen Sie die folgende Tabelle aus und begründen Sie Ihre Antworten.

Kontextfreie Sprachen	Reguläre Sprachen	Aufzählbare Mengen	abgeschlossen bzgl.
Ja	Ja	Ja	Vereinigung
Nein	Ja	Ja	Schnitt
Nein	Ja	Nein	Komplement

KONTEXTFREIE SPRACHEN

Vereinigung. Seien L_1 und L_2 und kontextfreie Sprachen und $G_1 = (V_1, \Pi_1, S_1)$ und $G_2 = (V_2, \Pi_2, S_2)$ zwei kontextfreie Grammatiken die L_1 und L_2 erzeugen. O.B.d.A. nehmen wir an, daß G_1 und G_2 keine gemeinsamen Nichtterminale besitzen. Weiters komme das Nichtterminalzeichen S weder in V_1 noch in V_2 vor. Wir setzen

$$G = (V_1 \cup V_2 \cup \{S\}, \Pi_1 \cup \Pi_2 \cup \{S := S_1, S ::= S_2\}, S).$$

Behauptung. $L(G) = L_1 \cup L_2$
Beweis.
"$\subseteq$": Sei $w \in L(G)$. Dann $S \Rightarrow w$ und entweder $S \rightarrow S_1 \Rightarrow w$ oder $S \rightarrow S_2 \Rightarrow w$.
1. Fall. $S \rightarrow S_1 \Rightarrow w$: Es ist S_1 ein Nichterminal aus G_1. Da G_1 und G_2 keine gemeinsamen Nichtterminale besitzen und auch das neue Startzeichen S weder in V_1 noch in V_2 vorkommt, werden in der Ableitung $S_1 \Rightarrow w$ nur Produktionen aus G_1 angewandt (Beweis durch Induktion). Mithin muß das abgeleitete Wort w zu L_1 gehören.
2. Fall. $S \rightarrow S_2 \Rightarrow w$: Auf analoge Weise folgt $w \in L_2$.
Es gilt also $w \in L_1$ oder $w \in L_2$ und somit $w \in L_1 \cup L_2$.

"$\supseteq$": Sei $w \in L_1 \cup L_2$. Dann $w \in L_1$ oder $w \in L_2$. Dann $S_1 \Rightarrow w$ mit Produktionen aus Π_1 oder $S_2 \Rightarrow w$ mit Produktionen aus Π_2. Dann $S \Rightarrow w$ mit Produktionen aus Π und schließlich $w \in L(G)$. Man beachte, daß man für diese Richtung die Voraussetzung der Disjunktheit der Nichtterminale nicht benötigt.

Schnitt. Es sei $L_1 = \{a^n b^n c^m : n > 0, m > 0\}$ und $L_2 = \{a^n b^m c^m : n > 0, m > 0\}$. Beide Sprachen sind kontextfrei (geben Sie eine Grammatik an!), ihr Schnitt $L_1 \cap L_2 = \{a^n b^n c^n : n > 0\}$ bekanntermaßen nicht.

Komplement. Nehmen wir an, daß Komplement $\bar{L} = \{w \in T^* : w \notin L\}$ einer kontextfreien Sprache L wäre kontextfrei. Weiters seien L_1 und L_2 kontextfrei. Dann wären $\overline{L_1}$ und $\overline{L_2}$, $\overline{L_1} \cup \overline{L_2}$ und $\overline{\overline{L_1} \cup \overline{L_2}} = L_1 \cap L_2$ ebenfalls kontextfrei. Widerspruch!

REGULÄRE SPRACHEN

Vereinigung. Die reguläre Sprachen sind kontextfrei und daher auch bezüglich der Vereinigung abgeschlossen.

Schnitt. Hat man zwei reguläre Sprachen, so gibt es für diese auch zwei endliche Akzeptoren, die deterministisch, vollständig und frei von spontanen Übergängen sind. Der Produktakzeptor akzeptiert den Schnitt der beiden Sprachen. Damit ist der Schnitt auch regulär.

Komplement. Zu einer regulären Sprache gibt es einen endlichen Akzeptor der deterministisch, vollständig und frei von spontanen Übergängen ist. Der Komplementakzeptor akzeptiert das Komplement der Sprache. Damit ist das Komplement auch regulär.

AUFZÄHLBARE MENGEN

Seien $A, B \subseteq \mathbb{N}$ aufzählbar.

Vereinigung.
1. Fall: $A = \emptyset$. Dann ist $A \cup B = B$ nach Voraussetzung aufzählbar.
2. Fall: $B = \emptyset$. Dann ist $A \cup B = A$ nach Voraussetzung aufzählbar.
3. Fall: $A, B \neq \emptyset$. Dann gibt es zwei totale und berechenbare Funktionen φ_a und φ_b, so daß $A = Im(\varphi_a)$ und $B = Im(\varphi_b)$. Wir definieren eine Funktion $\varphi : \mathbb{N} \to \mathbb{N}$ durch

$$\varphi(n) = \begin{cases} \varphi_a(n \text{ div } 2) & \text{, falls } n \text{ gerade} \\ \varphi_b(n \text{ div } 2) & \text{, sonst} \end{cases}$$

Diese Funktion zählt abwechselnd Elemente aus A und B auf, mithin die Elemente aus $A \cup B$. Offen ist noch die Frage, ob φ total und berechenbar ist. Wir betrachten dazu das folgende erweiterte While-Programm:

```
begin
    n := X1 div 2;
    if X1 mod 2 = 0 then
        X1 := Φ(a, n)
    else
        X1 := Φ(b, n)
    fi
end
```

Wie man (in diesem Falle wirklich) leicht sieht, berechnet das Programm die Funktion φ. Da φ_a und φ_b total und berechenbar sind, terminiert das Programm für jede Eingabe. Es ist also φ total und berechenbar und es gilt $Im(\varphi) = A \cup B$.

Schnitt.

1. Fall: $A = \emptyset$ oder $B = \emptyset$. Dann ist $A \cap B = \emptyset$ aufzählbar.
2. Fall: $A, B \neq \emptyset$. Jetzt gibt es zwei Möglichkeiten:
2.1. $A \cap B = \emptyset$. Dann ist $A \cap B$ aufzählbar.
2.2. $A \cap B \neq \emptyset$. Dann gibt es mindestens ein Element $c \in A \cap B$.
Weiters gibt es zwei totale und berechenbare Funktionen φ_a und φ_b, so daß $A = Im(\varphi_a)$ und $B = Im(\varphi_b)$. Wie schon im Fall der Vereinigung müssen wir eine Funktion $\varphi : \mathbb{N} \to \mathbb{N}$ finden, die total und berechenbar ist und die das Bild $A \cap B$ hat.
Dazu konstruieren wir zunächst eine 2-stellige Hilfsfunktion $h : \mathbb{N}^2 \to \mathbb{N}$ durch

$$h(x, y) = \begin{cases} \varphi_a(x) & \text{, falls } \varphi_a(x) = \varphi_b(y) \\ c & \text{, sonst} \end{cases}.$$

Ein Programm, daß diese Funktion berechnet ist schnell geschrieben (schreiben Sie eins!) und man überzeugt sich leicht, daß diese Funktion total und berechenbar ist. Jetzt überzeugen wir uns davon, daß das Bild dieser Funktion $A \cap B$ ist:

Sei also $w \in Im(h)$. Dann gibt es x, y mit $w = h(x, y)$. Dann ist $w = c$ oder $w = \varphi_a(x) = \varphi_b(y)$. In beiden Fällen liegt w im Schnitt von A und B. Umgekehrt überzeuge man sich davon, daß jedes Element des Schnitts von A und B auch im Bild von h liegt.

Sind wir jetzt fertig? Nein. Wir haben zwar eine Funktion die total und berechenbar ist und die gewünschte Menge als Bild besitzt. Dummerweise ist unsere Funktion aber zweistellig. Das ist nicht weiter schlimm. Wie schon in der Vorlesung können wir dieses Problem mit Hilfe der Komposition und der Projektionsfunktionen beseitigen: Wir definieren dazu eine einstellige Funktion $\varphi : \mathbb{N} \to \mathbb{N}$ durch $\varphi = h \circ (\pi_1 \otimes \pi_2)$. Diese ist total und berechenbar und hat das Bild $A \cap B$.

Komplement.
Nehmen wir an, das Komplement einer aufzählbaren Menge wäre aufzählbar. Dann wäre jede aufzählbare Menge entscheidbar. Wir kennen aber mindestens eine aufzählbare, nicht entscheidbare Menge (zum Beispiel NE aus Kapitel 11 der Vorlesung). Widerspruch!

Aufgabe 2 Aufzählbarkeit (Bierwald)

Zeigen Sie, daß die Menge $M = \{i : Dom(\varphi_i) \neq \emptyset\}$ aufzählbar ist.

Wir beginnen mit der üblichen Fallunterscheidung.
1. Fall: M ist leer, dann ist M aufzählbar.
2. Fall: Es gibt ein Element m_0 in M. Wir definieren eine totale und berechenbare, dreistellige Hilfsfunktion h durch $h(i, x, y) = i$, falls $\psi_5(i, x, y) = 1$ und $h(i, x, y) = m_0$ sonst. Dann ist $Im(h) = M$. Es ist also M der Wertebereich einer totalen und berechenbaren, *dreistelligen* Funktion. Damit sind wir fast fertig. Wir müssen nur noch dafür sorgen, daß M der Wertebereich einer totalen und berechenbaren, *einstelligen* Funktion ist.

Dazu betrachten wir die Komposition $h \circ [\pi_1 \otimes (\pi_1 \circ \pi_2) \otimes (\pi_2 \circ \pi_2)]$. Diese hat den Definitions- und Wertebereich $\mathbb{N}$ und ist ebenfalls total und berechenbar. Weiters ist $Im\,(h \circ [\pi_1 \otimes (\pi_1 \otimes \pi_2) \otimes (\pi_2 \otimes \pi_2)]) = M$ und M mithin aufzählbar.

Übrigens. Das While-Programm `begin end` hat die Gödelnummer 1112 und berechnet die Identitätsfunktion. Für diese Funktion Id $(= \varphi_{1112})$ gilt sicherlich $Dom(Id) \neq \emptyset$. Somit ist M nicht leer und wir könnten direkt mit dem Wert $m_0 = 1112$ in den zweiten Teil der Fallunterscheidung einsteigen.

Aufgabe 3 Paarfunktion (Käufl)

a) Seien $f: X \to Y$ und $g: Y \to X$ zwei Funktionen mit der Eigenschaft $g(f(x)) = x$ und $f(g(y)) = y$. Zeigen Sie, daß f und g bijektiv sind, $f^{-1} = g$ und $g^{-1} = f$.

Sei $y \in Y$, dann ist $f(g(y)) = y$, also gibt es ein $x \in X$, nämlich $g(y)$, für das $y = f(x)$. Also ist f surjektiv.

Seien $x_1, x_2 \in X$ und $x_1 \neq x_2$. Da $g(f(x)) = x$ erhalten wir $g(f(x_1)) = x_1$ und $g(f(x_2)) = x_2$ und da g Abbildung, muß $f(x_1) \neq f(x_2)$ sein. Also ist f injektiv.

Analog weist man die Bijektivität von g nach.

Ist $f(x) = y$, dann ist $g(y) = x$. (Da $g(f(x)) = x$.) Also $f^{-1} = g$.

Analog folgt $g^{-1} = f$.

b) Zeigen Sie die Bijektivität der Paarfunktion.

Injektivität. $(i_1, j_1) \neq (i_2, j_2)$. Wir zeigen $\tau(i_1, j_1) - \tau(i_2, j_2) \neq 0$. Wegen

$$\tau(i, j) = i + \frac{1}{2}(i+j)(i+j+1) = i + \sum_{k=1}^{i+j} k$$

ist es zweckmäßig, die folgenden Fälle zu unterscheiden

Fall 1. $i_1 + j_1 = i_2 + j_2$. Dann $\tau(i_1, j_1) - \tau(i_2, j_2) = i_1 - i_2 \neq 0$.

Fall 2. $i_1 + j_1 \neq i_2 + j_2$. Ohne Einschränkung (der Allgemeinheit) können wir $i_1 + j_1 > i_2 + j_2$ annehmen. Dann

$$\tau(i_1, j_1) - \tau(i_2, j_2) = i_1 - i_2 + \sum_{k = i_2 + j_2 + 1}^{i_1 + j_1} k = i_1 + j_2 + 1 + \sum_{k = i_2 + j_2 + 2}^{i_1 + j_1} k > 0$$

Surjektivität. $n \in Nat$ gegeben. Sei z die größte Zahl mit $s = \sum_{k=1}^{z} k \leq n$. Also $z = i + j$. Dann $i = n - s$ und daraus ergibt sich $j = z - i$.

c) Zeigen Sie, daß jede endliche Folge von natürlichen Zahlen durch eine natürliche Zahl kodiert werden kann und daß diese Funktion berechenbar ist.

Gegeben ist die Folge $a_1, \ldots, a_n$. Wir kodieren die Folge ineinandergeschachtelter Paare $(a_1, (a_2, \ldots, (a_{n-1}, a_n)\ldots))$. Programmschema für eine Folge der Länge n:

$X_1 = a_1; \ldots X_n = a_n;$

$Y := \tau(X_{n-1}, X_n);$

$Y := \tau(X_{n-2}, Y);$

$\ldots$

$X_1 := \tau(X_1, Y)$

Die While-Sprache erlaubt nicht die Angabe von Funktionen, die „beliebig viele“ Argumente haben. Man müßte also für jede Länge n ein eigenes Programm nach dem vorangehenden

Schema schreiben.

Will man in die Kodierung Y auch die Länge n der Folge aufnehmen, erweitert man das Programm um das Argument X_{n+1} ($= a_{n+1}$) und verlängert es um die Anweisung
$X_1 := \tau(X_{n+1}, X_1)$.

Dekodierung.

$Z := 1$;
while $Z \neq K$ **do** $Z := Z + 1$; $Y := \pi_2(Y)$ **od**;
$X := \pi_1(Y)$;

Dieses While-Programm funktioniert für Folgen beliebiger Länge. Ist $K \geq 1$ nicht größer als die Länge der Folge Y, dann wird das K.-te Element ausgewählt.

Anmerkung: Die Fälle $K = 0$ und $K > n + 1$ behandelt man bei Anwendungen am zweckmäßigsten dort, wo die Dekodierung benötigt wird.

Aufgabe 4 Signumfunktion, Aufzählbarkeit endlicher Mengen (Käufl)

a) Die Signumfunktion hat den Wert 0, wenn ihr Argument 0 ist und 1 sonst. Geben Sie ein While-Programm für diese Funktion an. Wie hängt das „gedeckelte" Plus $\oplus$ mit der Signumfunktion zusammen?

$x := x \dot{-} \text{Vorg}(x)$

$x \oplus y = 1$, falls $x + y \geq 1$ und 0 sonst. Also $x \oplus y = \text{Sign}(x + y)$, als While-Programm (mit Ergänzungen): $x := x + y$; $x := x \dot{-} \text{Vorg}(x)$

b) Beweisen Sie Satz 11.8, Teil 1 (siehe Skript). Verwenden Sie dabei die in Aufgabe 6, Übungsblatt 10 als berechenbar nachgewiesene Funktion.

$A \subseteq N$ endlich. Die charakteristische Funktion f hat die Eigenschaft: $f(n) = 1$ genau dann, wenn $n \in A$, hat also nur für endlich viele Argumente einen von 0 verschiedenen Wert. Wegen Aufgabe 6, Übungsblatt 10 ist f berechenbar und damit A entscheidbar.

Ist f die charakteristische Funktion für $\overline{A}$ und $\overline{A}$ endlich, dann ist f berechenbar. $f(n) = 1$ genau dann, wenn $n \in \overline{A}$, also $f(n) = 0$ genaudann, wenn $n \in A$. $1 \dot{-} f(n)$ ist genau dann 1, wenn $n \in A$. Da diese Funktion berechenbar ist, ist A entscheidbar.