Prüfungsteilnehmer	Prüfungstermin	Einzelprüfungsnummer
Kennzahl: ____________	Frühjahr 1995	66110
Kennwort: ____________		
Arbeitsplatz-Nr.: ______		

Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen

- Prüfungsaufgaben -

Fach: Informatik (vertieft studiert)

Einzelprüfung: Automatentheorie, Algorithm. Sprachen

Anzahl der gestellten Themen (Aufgaben): 1

Anzahl der Druckseiten dieser Vorlage: 4

Bitte wenden!

Sämtliche Teilaufgaben sind zu bearbeiten!

Aufgabe 1

Gegeben sei die Grammatik Γ_1 mit $\{\mathbf{a},\mathbf{b}\}$ als Menge der Terminalzeichen, den Nichtterminalzeichen Z,A,B, dem Axiom Z und den Produktionsregeln

$$
\begin{aligned}
Z &\rightarrow \mathbf{a}B \\
Z &\rightarrow \mathbf{b}A \\
A &\rightarrow \mathbf{a} \\
A &\rightarrow \mathbf{a}Z \\
B &\rightarrow \mathbf{b} \\
B &\rightarrow \mathbf{b}Z
\end{aligned}
$$

Die Grammatik Γ_2 entstehe aus Γ_1 dadurch, daß zu diesen Produktionsregeln noch die weiteren Regeln

$$
\begin{aligned}
A &\rightarrow \mathbf{b}AA \\
B &\rightarrow \mathbf{a}BB
\end{aligned}
$$

hinzugenommen werden.

Der jeweilige Sprachschatz von Γ_1 und Γ_2 sei mit $\mathcal{L}(\Gamma_1)$ bzw. $\mathcal{L}(\Gamma_2)$ bezeichnet. ε bezeichne das leere Wort.

a) Beweisen Sie: $\mathcal{L}(\Gamma_1) = \{\mathbf{ab},\mathbf{ba}\}^* \setminus \{\varepsilon\}$.

b) Konstruieren Sie direkt aus Γ_1 einen nicht-deterministischen endlichen Automaten, der genau die Zeichenreihen von $\mathcal{L}(\Gamma_1)$ akzeptiert. Konstruieren Sie dann aus diesem Automaten einen deterministischen endlichen Automaten, der genau die Zeichenreihen von $\mathcal{L}(\Gamma_1)$ akzeptiert.

c) Geben Sie eine (deterministische) Turingmaschine T an, die außer einem Leerzeichen nur die Zeichen aus $\{\mathbf{a},\mathbf{b}\}$ verwendet und $\mathcal{L}(\Gamma_1)$ in folgendem Sinne akzeptiert: Ein Wort $w \in \{\mathbf{a},\mathbf{b}\}^*$ steht auf dem ansonsten leeren Band. Angesetzt auf das erste Zeichen von w (bzw. auf ein Leerzeichen, falls w das leere Wort ist), erreicht T genau dann nach endlich vielen Schritten einen Endzustand, wenn $w \in \mathcal{L}(\Gamma_1)$ ist.

d) Beweisen Sie: $\mathbf{aaabbabbba} \in \mathcal{L}(\Gamma_2)$.

e) Für ein $w \in \{\mathbf{a},\mathbf{b}\}^*$ entstehe $\overline{w}$ aus w, indem man jedes $\mathbf{a}$ in w durch $\mathbf{b}$ und jedes $\mathbf{b}$ in w durch $\mathbf{a}$ ersetzt.

Beweisen Sie: Ist $w \in \mathcal{L}(\Gamma_2)$, so ist auch $\overline{w} \in \mathcal{L}(\Gamma_2)$.

f) Überführen Sie Γ_2 in Chomsky-Normalform.

Fortsetzung nächste Seite!

Aufgabe 2

NAT bezeichne den abstrakten Datentyp mit der Sorte nat der natürlichen Zahlen (einschließlich 0) und den üblichen Operationen auf nat. Der abstrakte Datentyp VEKTOR sei wie folgt definiert:

```
abstract type VEKTOR
   uses NAT                  (* Alles, was NAT enthält, darf verwendet werden *)
   sorts vektor, index       (* Die Sorten von VEKTOR *)
   functions null: → vektor,
             proj: vektor × index → nat,
             succ: vektor × index → vektor
   axioms ∀x ∈ vektor ∀i,j ∈ index:
             proj(null,i) = 0,
             proj(succ(x,i),i) = proj(x,i) + 1,
             proj(succ(x,i),j) = proj(x,j)    für i ≠ j
endoftype
```

a) Die Funktion f: nat × index → vektor sei gegeben durch die Funktionsvereinbarung

```
function f(k:nat, i:index) vektor:
   if k = 0 then null else succ(f(k-1,i),i) endif
```

Beweisen Sie unter Verwendung der Axiome von VEKTOR, daß für alle $k \in$ nat und $i,j \in$ index gilt:

a1) $proj(f(k,i),i) = k$

a2) $proj(f(k,i),j) = 0$ für $i \neq j$

b) Geben Sie in Analogie zur Funktion f in Teilaufgabe a) eine rekursive Funktionsvereinbarung für eine Funktion g: vektor × nat × index → vektor an, für die für alle $k \in$ nat und $i,j \in$ index gilt:

b1) $proj(g(x,k,i),i) = proj(x,i) + k$

b2) $proj(g(x,k,i),j) = proj(x,j)$ für $i \neq j$

Beweisen Sie b1) und b2) für Ihre Lösung.

c) Für ein fest vorgegebenes $n \in \mathbb{N}$ sei nun index $= \{i \mid i \in \mathbb{N} \text{ und } 1 \leq i \leq n\}$ und vektor $= \{(x_1,\ldots,x_n) \mid x_i \in \mathbb{N}_0 \text{ für } 1 \leq i \leq n\}$. Die Funktionen *null*, *proj* und *succ* seien gegeben durch

$null = (0,0,..,0)$ ("Nullvektor"),
$proj((x_1,\ldots,x_n),i) = x_i$,
$succ((x_1,\ldots,x_n),i) = (x_1,\ldots,x_{i-1},x_i+1,x_{i+1},\ldots,x_n)$.

c1) Zeigen Sie, daß die so definierten Funktionen die Axiome von VEKTOR erfüllen.

c2) Objekte der Sorte vektor können in höheren Programmiersprachen als Reihungen der Länge n realisiert werden (Typbezeichnung etwa **array** [1..n] **of** nat). Geben Sie (in der Notation einer derartigen Programmiersprache) Algorithmen zur Realisierung der Funktionen *null*, *proj* und *succ* an.

c3) Geben Sie (in einer Notation wie in Teilaufgabe c2)) einen Algorithmus an, der für $(x_1,\ldots,x_n),(y_1,\ldots,y_n) \in$ vektor den "Summenvektor" $(x_1+y_1,\ldots,x_n+y_n)$ berechnet und dabei nur die Funktionen *null*, *proj* und *succ* verwendet. Erläutern Sie die wesentlichen Schritte des Algorithmus durch geeignete Kommentare.

Fortsetzung nächste Seite!

Aufgabe 3

Durch die Funktionsvereinbarung

```
function f(x,y,z:nat)nat:
   if z=0 then x+y
   else if y=0 then 1
        else f(f(x,y-1,z),x,z-1) endif
   endif
```

ist eine Funktion $f: \mathbb{N}_0^3 \rightarrow \mathbb{N}_0$ definiert. Ferner sei die Funktion $g: \mathbb{N}_0 \times \mathbb{N}_0 \rightarrow \mathbb{N}_0$ mit

$$g(x,y) = \sum_{k=0}^{y} x^k$$

gegeben.

a) Bestimmen Sie $f(3,4,1)$.

b) Beweisen Sie: $f(x,y,z)$ terminiert für alle $x,y,z \in \mathbb{N}_0$.

c) Geben Sie (in der Notation einer höheren Programmiersprache) unter Verwendung eines Kellers als zusätzlicher "Hilfs"-Datenstruktur einen iterativen Algorithmus an, der $f(x,y,z)$ für beliebige $x,y,z \in \mathbb{N}_0$ berechnet. Erläutern Sie Idee und wesentliche Schritte Ihrer Lösung.

d) Beweisen Sie, daß für alle $x,y \in \mathbb{N}_0$ gilt: $f(x,y,2) = g(x,y)$.

e) Beweisen Sie, daß die Funktion g sich unter ausschließlicher Verwendung von Addition und Multiplikation (auf $\mathbb{N}_0$) sowie primitiver Rekursion definieren läßt.