Prüfung aus Übersetzerbau 11.3.1994 Musterlösung

1. 25 % Gegeben sei folgende Grammatik:

(S' $\rightarrow$ S $)
S $\rightarrow$ A | B C
A $\rightarrow$ a A | b
B $\rightarrow$ d | ε
C $\rightarrow$ a b

a) (20 %) Bestimmen Sie die `First`- und `Follow`-Mengen aller Nonterminale und zeigen Sie, daß die Grammatik nicht LL(1) ist.

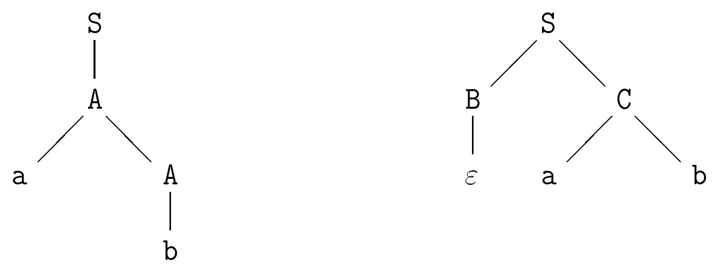
First(S) = {a,b,d}	Follow(S) = {$}
First(A) = {a,b}	Follow(A) = {$}
First(B) = {d,ε}	Follow(B) = {a}
First(C) = {a}	Follow(C) = {$}

Die Grammatik ist nicht LL(1), da First(A) $\cap$ First(BC) = {a} $\neq$ {}
(First(BC) = {d,a})

b) (5 %) Zeigen Sie, daß die Grammatik mehrdeutig ist.

Es gibt zwei verschiedene Ableitungsbäume für `ab`:

S $\rightarrow$ A $\rightarrow$ aA $\rightarrow$ ab oder
S $\rightarrow$ BC $\rightarrow$ C $\rightarrow$ ab



2. 30 % Gegeben sei folgende Grammatik, die eine Liste von Zahlen beschreibt:

```
S → L
L → L , N
L → N
N → num
```

Man nehme an, daß der Wert einer Zahl im Attribut `num.val` steht. Es soll der Mittelwert der Zahlen berechnet und im synthetisierten Attribut `S.med` zur Verfügung gestellt werden.

a) (10 %) Lösen Sie die Aufgabe, indem Sie die Grammatik zu einer Attributierten Grammatik erweitern.

Produktionen	Regeln
S → L	S.med:=L.sum/L.anz
L → L_1 "," N	L.sum:=L_1.sum+N.val; L.anz:=L_1.anz+1
L → N	L.sum:=N.val; L.anz:=1
N → num	N.val:=num.val

b) (20 %) (unabhängig von a) Lösen Sie die Aufgabe durch ein LL(1)-Übersetzungsschema.

```
S → L {S.med:=L.med}
L → N {Lr.sum:=N.val; Lr.anz:=1} Lr {L.med:=Lr.med}
Lr → "," N {Lr1.sum:=Lr.sum+N.val; Lr1.anz:=Lr.anz+1} Lr1 {Lr.med:=Lr1.med}
Lr → ε {Lr.med:=Lr.sum/Lr.anz}
N → num {N.val:=num.val}
```

oder auch ohne ererbte Attribute:

```
S → L {S.med:=L.sum/L.anz}
L → N Lr {L.sum:=N.val+Lr.sum; L.anz:=Lr.anz+1}
Lr → "," N Lr1 {Lr.sum:=N.val+Lr1.sum; Lr.anz:=Lr1.anz+1}
Lr → ε {Lr.sum:=0; Lr.anz:=0}
N → num {N.val:=num.val}
```

3. 25 % Gegeben sei folgendes Modula-Programmstück:

```
...
IF ((a < b) AND (c < d)) OR ((e < f) AND (g < h))
THEN
  x:= x+1;
ELSE
  y:= y+1;
END;
```

a) (15 %) Erzeugen Sie für das obige Programmstück **Quadrupel-Code** nach der **Kontrollflußmethode**.

```
       if a<b goto L1
       goto L2
L1:    if c<d goto Lthen
       goto L2
L2:    if e<f goto L3
       goto Lelse
L3:    if g<h goto Lthen
       goto Lelse
Lthen: t1:=x+1
       x:=t1
       goto Lend
Lelse: t2:=y+1
       y:=t2
Lend:
```

b) (10 %) Optimieren Sie den erhaltenen Code mittels **Nachoptimierung**.

```
       if a≥b goto L3
L1:    if c<d goto Lthen
L3:    if e≥f goto Lelse
L4:    if g≥h goto Lelse
Lthen: t1:=x+1
       x:=t1
       goto Lend
Lelse: t2:=y+1
       y:=t2
Lend:
```

4. 20 % Gegeben sei der Ausdruck `((a*b)-(c*d))/((e*f)-(g*h))`

a) (10 %) Erzeugen Sie optimalen Code für eine 2-Adreß-Maschine, die nur 2 Register hat (`maxReg=2`).

```
MOV e,R0
MUL f,R0
MOV g,R1
MUL h,R1
SUB R1,R0
MOV R0,t1
MOV a,R0
MUL b,R0
MOV c,R1
MUL d,R1
SUB R1,R0
DIV t1,R0
```

b) (10 %) Erzeugen Sie Code für einen RISC-Prozessor, wobei 5 Register zur Verfügung stehen und keine Leerbefehle vorkommen sollen.

```
LAD a,R0
LAD b,R1
LAD c,R2
LAD d,R3
LAD e,R4
MUL R0,R1,R0
LAD f,R1
MUL R2,R3,R2
LAD g,R3
SUB R0,R2,R0
LAD h,R2
MUL R4,R1,R1
MUL R3,R2,R2
SUB R1,R2,R1
DIV R0,R1,R0
```

Ergebnisse der Prüfung aus Übersetzerbau vom 11.3.1994

<u>Einsichtnahme:</u>

Mo 10-11 Uhr, Herr Berger,
Herr Tschernko