

Zuname _____ Vorname _____ Matr.-Nr. _____

Übungsgruppe

Punkte _____ korr. _____

- 1 (Löberbauer) Do 10¹⁵-11⁴⁵
- 2 (Löberbauer) Do 13⁴⁵-15¹⁵
- 3 (Wimmer) Do 10¹⁵-11⁴⁵

Letzter Abgabetermin

Mittwoch, 10.10.2007, 20¹⁵ Uhr

Grammatiken

1. Grundbegriffe

(2+3+3+6 Punkte)

Die Grammatik der Sprache *MicroJava* finden Sie im VO-Skriptum im Kapitel 4.2 auf Seite 7.

- a) Geben Sie den Satz mit der minimalen Anzahl von Terminalsymbolen an, den man mit dieser Grammatik erzeugen kann und der mindestens einen *Block* enthält. Aus wie vielen Terminalsymbolen besteht er?
- b) Betrachten Sie die Nonterminalsymbole *Statement*, *Term* und *VarDecl*. Geben Sie für jedes dieser Nonterminalsymbole an, ob es links-, zentral- oder rechtsrekursiv ist, und ob es direkt oder indirekt rekursiv ist.
- c) Zeichnen Sie den Syntaxbaum für folgenden Satz:

```
program Tester final int OPT = 42; { int test(int x) { return OPT - x; } }
```

 Gibt es mehrere Syntaxbäume für diesen Satz?
- d) Bestimmen Sie alle terminalen Anfänge und Nachfolger für die Regeln *VarDecl*, *Statement*, *Term* und *Addop*.

2. Konstruktion einer Grammatik

(5 Punkte)

Geben Sie eine Grammatik (in EBNF) für die Zahlen einer fiktiven Programmiersprache gemäß den folgenden Bedingungen an:

- Eine Zahl ist entweder eine Dezimal-, eine Hexadezimal- oder Binärzahl.
- Hexadezimal- und Binärzahlen beginnen mit "#". Binärzahlen enden mit einem "b".
- Dezimalzahlen enthalten Tausender-Punkte, d.h. wenn eine Dezimalzahl aus mehr als drei Ziffern besteht, muss nach jeder dritten Ziffer (gerechnet von rechts) ein Punkt (".") stehen.
- Die Terminalklassen *b*, *d* und *h* sind gegeben, *b* deckt die Ziffern 0 und 1 ab, *d* deckt die Ziffern zwischen 2 und 9 und *h* die Ziffern A bis F.

Beispiele für gültige Zahlen: 123, 1.456, #1101b, #AF, #23, #1A, #23B9

Beispiele für ungültige Zahlen: 1234, 567.8, #5476b, #2b, #b, #

3. Beseitigung von Linksrekursionen

(5 Punkte)

Gegeben sei folgender Auszug aus einer fiktiven Grammatik, der Zuweisungen beschreibt:

`Assignment = Designator "=" Expression.`

`Designator = ident | Designator "." ident.`

`Expression = (Expression "+" number) | number.`

Beseitigen Sie alle Linksrekursionen und geben Sie die transformierte Grammatik in EBNF an. *ident* und *number* sind Terminalklassen, die einen Namen (Buchstabe gefolgt von Ziffern und Buchstaben) bzw. eine Zahl (bestehend aus Ziffern) definieren.