

Aufgabenblatt zur 5. Übung

Zeitraum: 15.11. bis 19.11.2010

1. Aufgabe: (Klausuraufgabe 08.2010, AGS 2.48)

(a) Geben Sie eine EBNF-Definition \mathcal{E}' an, so dass gilt: $W(\mathcal{E}') = \{a^{i+j}b^{j+k+l}ac^{2l} \mid i, j, k, l \geq 0\}$

(b) Sei $\mathcal{E} = (V, \Sigma, S, R)$ mit $V = \{S, A\}$, $\Sigma = \{a, b\}$ und $R = \{S ::= aAb, A ::= [S] \mid b\}$. Berechnen Sie die syntaktischen Kategorien $W(\mathcal{E}, S)$ und $W(\mathcal{E}, A)$ mit Hilfe der Fixpunktsemantik. Gehen Sie dazu in den folgenden Schritten vor.

- Dokumentieren Sie zunächst 5 Iterationsschritte.
- Schreiben Sie dann in Mengenschreibweise die Sprachen $W(\mathcal{E}, S)$ und $W(\mathcal{E}, A)$ auf.

2. Aufgabe: (AGS 2.21*)

Sei $\mathcal{E} = (V, \Sigma, \langle \text{Block} \rangle, R)$ eine EBNF-Definition mit

$V = \{\langle \text{Block} \rangle, \langle \text{Decl} \rangle, \langle \text{Ident} \rangle, \langle \text{StatSeq} \rangle, \langle \text{Stat} \rangle, \langle \text{Num} \rangle\}$,

$\Sigma = \{\text{BEGIN}, \text{END}, \text{VAR}, \text{INTEGER}, a, b, c, 0, 1, 2, 3, :=, ;, :, , \}$ und

$$\begin{aligned} R = \{ & \langle \text{Block} \rangle && ::= & \text{BEGIN}[\langle \text{Decl} \rangle][\langle \text{StatSeq} \rangle]\text{END} , \\ & \langle \text{Decl} \rangle && ::= & \text{VAR} \langle \text{Ident} \rangle \{ \langle \text{Ident} \rangle : \text{INTEGER}; \} , \\ & \langle \text{Ident} \rangle && ::= & ((a|b)|c) , \\ & \langle \text{StatSeq} \rangle && ::= & \langle \text{Stat} \rangle \{ ; \langle \text{Stat} \rangle \} , \\ & \langle \text{Stat} \rangle && ::= & \langle \text{Ident} \rangle := \langle \text{Num} \rangle , \\ & \langle \text{Num} \rangle && ::= & (((0 | 1) | 2) | 3) \} \end{aligned}$$

(a) Geben Sie ein korrektes Wort (einen Block) der Sprache dieser EBNF-Definition an.

(b) Konstruieren Sie ein System von Syntaxdiagrammen aus $\text{SynDia}(V, \Sigma)$, das dieselbe Sprache wie \mathcal{E} erzeugt. Nutzen Sie hierzu die in der Vorlesung kennengelernte Übersetzungsfunktion trans.

Beachten Sie, dass die Metasymbole hier nicht mit einem Dach gekennzeichnet wurden.

3. Aufgabe: (AGS 3.4*)

Für $n, k \in \mathbb{N}_0$ mit $k \leq n$ ist der Binomialkoeffizient $b(n, k)$ definiert durch:

$$b(n, k) := \frac{n!}{k! * (n - k)!}$$

Schreiben Sie ein C-Programm, das Binomialkoeffizienten berechnet. Überlegen Sie sich Problemlösungen, die es erlauben, dass Ihr C-Programm möglichst große Zahleneingaben korrekt verarbeiten kann.

4. Aufgabe: (AGS 3.6*)

Schreiben Sie für die Berechnung von Fibonacci-Zahlen ein *C*-Programm, welches eine natürliche Zahl als Eingabe fordert und den zugeordneten Funktionswert ausgibt. Die Berechnung der Fibonacci-Zahl selbst soll mit Hilfe einer Funktion `int fib(int z)` erfolgen.

Realisieren Sie zwei Varianten der Funktion `fib`:

- (a) einen iterativ arbeitenden Berechnungsalgorithmus und
- (b) einen rekursiv arbeitenden Berechnungsalgorithmus.

Machen Sie Aussagen zur Effizienz beider Programmvarianten.