8. Übungsblatt „Programmiersprachen“ SS 99

Leif Kornstaedt, Prof. Gert Smolka

Ausgabe: 11. Juni 1999 **Abgabe:** 18. Juni 1999

Aufgabe VIII.1: Endpositionen (7 Punkte)

Eine Position in einem Ausdruck heißt *Endposition*, falls das Ergebnis des entsprechenden Teilausdrucks das Ergebnis des Gesamtausdrucks ist, vorausgesetzt, der Teilausdruck wird tatsächlich ausgewertet. Beispiel: Die Endpositionenen des Ausdrucks

```
let val x = y in if z then fn u => u else u x end
```

sind im Folgenden durch `(...)`* markiert:

```
let val x = y in if z then (fn u => (u)*)* else (u x)* end
```

Die Erkennung von Applikationen in Endposition ist für die effiziente Ausführung von Programmen notwendig, da Endapplikationen den Kellerblock der aufrufenden Prozedur wiederverwenden können.

Sei die folgende abstrakte Syntax für Ausdrücke gegeben:

```
type id = string

datatype exp =
    VarExp of id
  | IfExp of exp * exp * exp
  | LetExp of id * exp * exp
  | AppExp of exp * exp
  | FnExp of id * exp
  | EndExp of exp
```

Der Konstruktor `EndExp` dient zur Markierung von Endpositionenen. Schreiben Sie eine Prozedur

```
insertEnd: exp -> exp
```

die die Endpositionen eines Ausdrucks markiert. Testen Sie `insertEnd` mit dem obigen Beispielausdruck.

Aufgabe VIII.2: Funktoren und Strukturabstraktion $(3 + 1 + 3 + 4$ Punkte$)$

Seien die folgenden Definitionen gegeben:

```
signature S =
    sig
        type t
        val  x: t
    end

signature S' =
    sig
        include S
        type t'
        val  x': t'
    end

functor F(A: S) :> S' where type t' = A.t =
    struct
        type t  = A.t
        val  x  = A.x
        type t' = t
        val  x' = x
    end

structure B = F(type t = int val x = 22)

structure C = F(type t = int val x = 23)
```

a. Deklarieren Sie eine Signatur `SB` so, dass `SB` die prinzipale Signatur von `B` ist.

b. Ist `SB` auch die prinzipale Signatur von `C`?

c. Welche der folgenden Ausdrücke sind zulässig?

   (1) `B.x = B.x'`

   (2) `B.x = B.x`

   (3) `B.x' = B.x'`

   (4) `B.x' = C.x'`

   (5) `if true then B.x else C.x`

   (6) `B.x' + C.x' + 27`

d. Seien die obigen Deklarationen gegeben, wobei aber die **where**-Klausel in der Funktordeklaration gelöscht ist.

   (1) Deklarieren Sie eine Signatur `SB'` so, dass `SB'` die prinzipale Signatur von `B` ist.

   (2) Welche der Ausdrücke aus Teil c. sind noch zulässig?

Aufgabe VIII.3: Direkt ausführbare SML-Programme (4 + 4 Punkte)

Schreiben Sie in Moscow ML und in SML/NJ ein direkt ausführbares Programm, das die Argumente der Kommandozeile zeilenweise ausgibt.

a. Für Moscow ML soll die Quelle in einer Datei `args-mo.sml` stehen und das direkt ausführbare Programm im Kommandointerpreter mit

   ```
   mosmlc args-mo.sml -o args
   ```

   erzeugt werden. Das direkt ausführbare Programm soll im Kommandointerpreter mit

   `args` ⟨*Argumente*⟩

   ausführbar sein.

b. Für SML/NJ soll die Quelle in einer Datei `args-nj.sml` stehen und das Objektprogramm im SML/NJ-Interpreter mit

   ```
   use "args-nj.sml";
   ```

   erzeugt werden. Das Objektprogramm soll im Kommandointerpreter mit

   `sml @SMLload=args.osml` ⟨*Argumente*⟩

   ausführbar sein.

Aufgabe VIII.4: Direkt ausführbare Java-Programme (7 Punkte)

Schreiben Sie ein Java-Programm, das die Argumente der Kommandozeile sortiert und zeilenweise ausgibt. Beispielsweise würde ein Aufruf

```
java Sort doing rümpel bingo
```

die Ausgabe

```
bingo
doing
rümpel
```

erzeugen.