

ÜBUNGEN ZU ORGANISATION UND ARCHITEKTUR VON RECHNERN
SS 2002
SERIE 2

Aufgabe 4

(8 Punkte)

Es soll die in der Vorlesung vorgestellte SEMCD-Maschine in DrScheme implementiert werden. Definieren Sie dazu zunächst die notwendigen abstrakten Datentypen:

<code>(define-structure (Ap n exprlist))</code>	für Applikationen: $\overline{\text{@}}^{(n)} \text{exprlist}$
<code>(define-structure (ApM n m))</code>	für $\overline{\text{@}}^{(n)}$ -Konstruktoren auf M-Stack: $\overline{\text{@}}^{(n m)}$
<code>(define-structure (Lam n exprlist))</code>	für Abstraktionen: $\lambda^{(n)} \text{exprlist}$
<code>(define-structure (Clos bindlist lam))</code>	für Closures: $[\text{bindlist lam}]$
<code>(define-structure (Bind name expr))</code>	für Elemente des Environments: $\langle \text{name expr} \rangle$
<code>(define-structure (Var name))</code>	für Variablen
<code>(define-structure (Num val))</code>	für Zahlen

Beispiele:

- $\overline{\text{@}}^{(3)} 7 11 \lambda^{(3)} x y x \equiv (\text{make-Ap } 3 (\text{list } (\text{make-Num } 7) (\text{make-Num } 11) (\text{make-Lam } 3 (\text{list } (\text{make-Var } "x") (\text{make-Var } "y") (\text{make-Var } "x")))))$
- $[\langle z 1 \rangle : \langle x 3 \rangle : \text{nil } \lambda^{(3)} x y x] \equiv (\text{make-Clos } (\text{list } (\text{make-Bind } "z" (\text{make-Num } 1)) (\text{make-Bind } "x" (\text{make-Num } 3))) (\text{make-Lam } 3 (\text{list } (\text{make-Var } "x") (\text{make-Var } "y") (\text{make-Var } "x"))))$

Die in DrScheme eingebaute Darstellung von Konstruktor-Ausdrücken ist ziemlich unübersichtlich. Definieren Sie daher für Debugging-Zwecke eine Funktion `print-expr` zur Konvertierung eines abstrakten Ausdrucks in einen String. Für die beiden o. a. Beispiele soll `print-expr` etwa folgende Ausgabe liefern:

- `"ap{3} 7 11 lam{3} x y x"`
- `"[<z 1> : <x 3> : nil lam{3} x y x]"`

Aufgabe 5

(10 Punkte)

Implementieren Sie aufbauend auf Aufgabe 4 eine vereinfachte SEMCD-Maschine, die Ausdrücke lediglich vom C-Stack auf den S-Stack traversiert (Regeln (7) – (10) auf den Folien zu der Vorlesung).

Definieren Sie dazu eine Funktion `semcd`, die eine Liste mit fünf Stacks — Stacks werden ebenfalls durch Listen repräsentiert — konsumiert und eine Liste mit der neuen Stack-Konstellation zurückliefert.

Hinweis: Für Abstraktionen brauchen die Traversionsregeln *nicht* implementiert zu werden, deshalb ist in Aufgabe 4 auch keine `LamM`-Struktur definiert worden. Abstraktionen werden in der vollständigen SEMCD-Maschine ohnehin stets in eine Closure umgebaut und dann als Ganzes auf den S-Stack transportiert. Nehmen Sie also für diese Aufgabe an, daß in den zu verarbeitenden Ausdrücken keine Abstraktionen vorkommen.

Beispiel:

```
(semcd (list () ;; S is empty
           () ;; E is empty
           () ;; M is empty
           (list (make-Ap 2 (list (make-Num 7) ;; Expression on C
                                (make-Var "fun"))))
           ())) ;; D is empty
```

liefert das Ergebnis

```
(list (list (make-Ap 2 (list (make-Var "fun") ;; Traversed expression on S
                           (make-Num 7))))
      () ;; E is empty
      () ;; M is empty
      () ;; C is empty
      ()) ;; D is empty
```

Definieren Sie außerdem eine Funktion `print-semcd`, um SEMCD-Maschinenzustände übersichtlich ausgeben zu können. Für das obige Beispiel etwa:

Anfangszustand:

```
"( nil , nil , nil , ap{2} 7 fun : nil, nil )"
```

Endzustand:

```
"( ap{2} fun 7 : nil , nil , nil , nil, nil )" .
```

Aufgabe 6

(14 Punkte)

Erweitern Sie die Funktion `semcd` aus Aufgabe 5 zu einer vollständigen SEMCD-Maschine entsprechend den Regeln (1) – (10) auf den Folien.

Hinweis: Die Regeln (1) und (2) werden bereits vollständig durch die Regel (2a) abgedeckt.

Testen Sie Ihre Maschine unter anderem mit folgenden Beispielen:

- $\overline{\text{Q}}^{(2)} 15 \lambda^{(2)} x x$
- $\overline{\text{Q}}^{(2)} 34 \overline{\text{Q}}^{(2)} 15 \lambda^{(2)} x x$
- $\overline{\text{Q}}^{(3)} b a \lambda^{(3)} x y x$

Aufgabe 7

(4 Punkte)

Versuchen Sie, mit Ihrer SEMCD-Maschine den folgenden Ausdruck zu berechnen:

$$\overline{\text{Q}}^{(2)} 7 \overline{\text{Q}}^{(2)} 11 \lambda^{(3)} x y y$$

Begründen Sie, warum es dabei zu einer Fehlermeldung kommt. Wie muß der Ausdruck modifiziert werden, damit er vollständig ausgewertet werden kann?

Aufgabe 8

(4 Punkte)

Berechnen Sie mit Ihrer SEMCD-Maschine den folgenden Ausdruck:

$$\overline{\text{Q}}^{(2)} \lambda^{(2)} x \lambda^{(2)} y x \lambda^{(2)} k \overline{\text{Q}}^{(2)} k \overline{\text{Q}}^{(2)} k \lambda^{(2)} x \lambda^{(2)} y \lambda^{(2)} z \overline{\text{Q}}^{(2)} \overline{\text{Q}}^{(2)} z y \overline{\text{Q}}^{(2)} z x$$

- Welches Ergebnis erhalten Sie?
- Wie läßt sich die Funktionalität des Ergebnisses ermitteln?

Abgabe: Di., 14.05.2002 in den Übungen.