

# Programmiersprachen

Wintersemester 2006/2007

6. Übungsblatt

14. Dezember 2006

## Aufgabe 1:

Was liefert die Auswertung folgender Ausdrücke?

- a) `1 + callcc (fn k1 => 2 + callcc (fn k2 => throw k1 3))`
- b) `1 + callcc (fn k1 => 2 + callcc (fn k2 => throw k2 3))`
- c) `1 + callcc (fn k1 => 2 + callcc (fn k2 => throw k2 3) + throw k1 4)`
- d) `1 + callcc (fn k1 => 2 + callcc (fn k2 => throw k2 (throw k1 3)))`

## Aufgabe 2:

Schreibe eine Funktion `debug : ('a -> 'b) -> 'a -> unit`, so dass ein Aufruf `debug f x` zum Debuggen eines Funktionsaufrufs `f x` mit Hilfe von *breakpoints* dient. Breakpoints in einer Funktion `f` werden als Aufrufe einer Funktion `break : string -> unit` im Rumpf von `f` dargestellt. Ein Aufruf `break message` soll `message` an der Standard-Ausgabe schreiben und die Kontrolle an der SML-Interpreterumgebung übergeben. Die Ausführung der Funktion wird fortgesetzt durch den Aufruf einer Funktion `continue : unit -> unit`.

Z.B. sollte die Funktionsdefinition:

```
fun addL (l,a) =
  case l of nil => (break ("No more elements"^^"\na = "^(Int.toString a));
                  a)
        | h::r => (break ("h = "^(Int.toString h)^^"\n"^^
                        "a = "^(Int.toString a));
                  addL (r,h+a))
```

die folgende Interaktion erlauben:

```

- debug addL ([100,10,1],0);
h = 100
a = 0
val it = () : unit
- continue ();
h = 10
a = 100
val it = () : unit
- continue ();
h = 1
a = 110
val it = () : unit
- continue ();
No more elements
a = 111
val it = () : unit
- continue ();
val it = () : unit

```

(**Hinweis:** Benutze globale Referenzen, um die Zustände (die Continuations) der beobachteten Funktion bzw. der Interpreterumgebung zu merken. Eine Referenz auf eine Continuation vom Typ `unit cont` kann man mit Hilfe von `ref (SMLofNJ.isolate (fn () => ()))` initialisieren.)

### Aufgabe 3:

Schreibe eine Funktion `comp : ('a -> 'b) -> 'b cont -> 'a cont` zur Komposition einer Continuation mit einer Funktion. Ein Aufruf `comp f k` soll eine Continuation zurückliefern, die, wenn aktiviert mit einem Wert `v`, die Continuation `k` mit dem Wert `f v` aktiviert.