
Proseminar: Fortgeschrittene Programmierung

<http://proglang.informatik.uni-freiburg.de/teaching/proseminar/2007ws/>

Hinweise

Sie finden die Folien unter <http://proglang.informatik.uni-freiburg.de/teaching/proseminar/2007ws/> zum Nachschlagen.

Aufgabe 1

Schreiben Sie einen Interpreter, der mitzählt, wie viele binäre Operationen bei der Auswertung durchgeführt werden. Sie können dazu entweder die State Monade verwenden oder aber einen eigenen Datentyp, wie z.B.

```
data Count a = Count (a,Counter)
type Counter = Integer
```

Aufgabe 2

Verändern Sie den Interpreter mit Exceptions so, dass im Falle einer Exception ein alternativer Term ausgewertet wird. Erweitern Sie dazu den Term-Typ um `Try Term Term`. Beispielsweise soll `eval (Try ((Con 1) Div (Con 0)) (Con 42))` zu 42 ausgewertet werden.

Aufgabe 3

Verändern Sie den Interpreter mit Protokoll so, dass nur noch bestimmte Teile der Auswertung mitprotokolliert werden. Erweitern Sie dazu den Term-Typ um `Trace Term` und `Untrace Term`.

Aufgabe 4

Verwenden Sie geeignete Monadentransformer, um die Funktionalität der einzelnen vorgestellten Interpreter zu kombinieren.