

Das M-Wort in der Compilerbauvorlesung

Johannes Waldmann, F-IMN, HTWK Leipzig

Ich berichte über meine Vorlesung [5] zum (Interpreter- und) Compilerbau auf Grundlage des Buches *Design Concepts in Programming Languages* [4] (vgl. auch Besprechung in [2]), die ich in den letzten Jahren mehrfach gehalten habe.

Ich folge dem Buch in der Darstellung verschiedener semantischer Bereiche (durch Interpreter für Teilsprachen), verschiedener Varianten der Typisierung (durch abstrakte Interpretation), und schließlich der Übersetzung in maschinennahen Code (durch CPS-Transformation ähnlich [1]).

Während der Vorlesung entwickle ich konkreten Haskell-Code—mit Lücken, die in Übungen und Hausaufgaben gefüllt werden.

Die semantischen Bereiche stelle ich dabei durch `Monad`-Instanzen dar:

- `data Id a = Id a` für einen einfachen Interpreter,
- `data Maybe a = Nothing | Just a` für die Modellierung von Exceptions,
- `data Action a = Action (State -> (a, State))` für Zustandsänderungen,
- `data CPS a = CPS ((a -> Action Val) -> Action Val)` für Continuations.

Wie in der Vorlage geht es um Semantik, die Syntax ist abstrakt. Zur Behandlung konkreter Syntax erkläre ich nur die Monade `data Parser c a = Parser ([c] -> [(a, [c])])` und verweise ansonsten auf `Parsec` [3].

Die Autoren von [4] scheinen eifrig bemüht, das „M-Wort“ zu vermeiden, obwohl sie ständig über *bind* und *return* sprechen. Ich denke, man sollte da keine Angst haben, und stelle fest, daß Master-Studenten der Informatik auch mit wenig Haskell-Vorkenntnissen gut mitkommen.

Literatur

- [1] Jr. Guy Lewis Steele. *Lambda: The ultimate declarative*. MIT AI Lab Memo AIM-379, 1976. <http://repository.readscheme.org/ftp/papers/ai-lab-pubs/AIM-379.pdf>.
- [2] Eugene Kirpichov. *Recommending Design Concepts in Programming Languages*. message in mailing list <http://www.haskell.org/pipermail/haskell-cafe/2009-April/060709.html>, 2009.
- [3] Daan Leijen and Erik Meijer. *Parsec: Direct style monadic parser combinators for the real world*. Technical Report UU-CS-2001-35, Departement of Computer Science, Universiteit Utrecht, <http://legacy.cs.uu.nl/daan/pubs.html#parsec>, 2001.
- [4] Franklyn Turbak and David Gifford with Mark A. Sheldon. *Design Concepts in Programming Languages*. MIT Press, 2008.
- [5] Johannes Waldmann. *Vorlesung Compilerbau*. Folien <http://www.imn.htwk-leipzig.de/~waldmann/edu/ws12/cb/folien/main/>, Quelltexte <http://dfa.imn.htwk-leipzig.de/cgi-bin/gitweb.cgi?p=ws12-cb.git>, 2012.