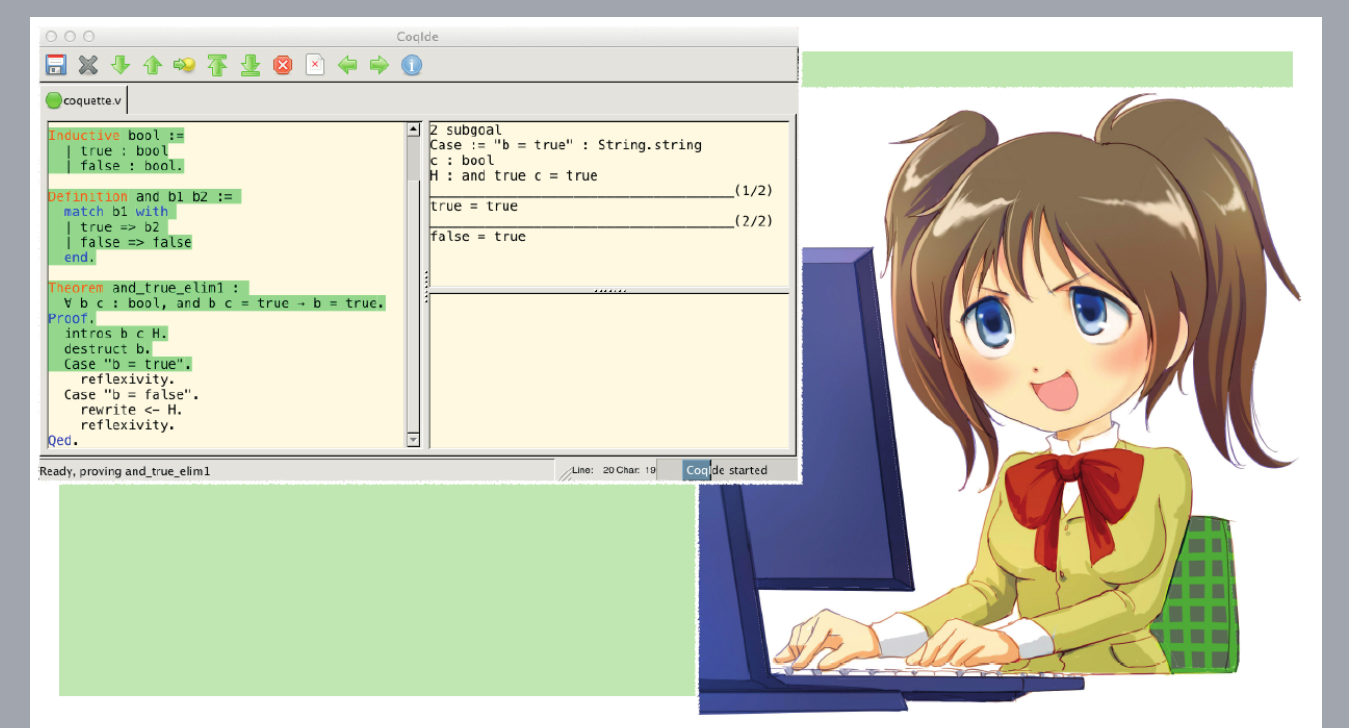


Lehrstuhl für Informatik 8 (Theoretische Informatik)

Lehrveranstaltungen SoSe 2023

Praktische Semantik von Programmiersprachen

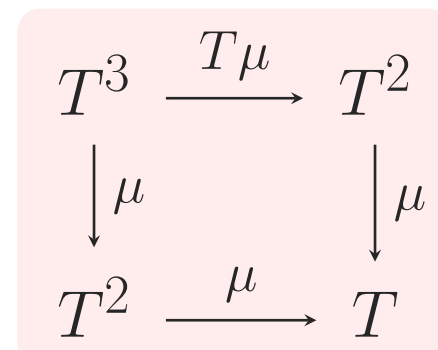
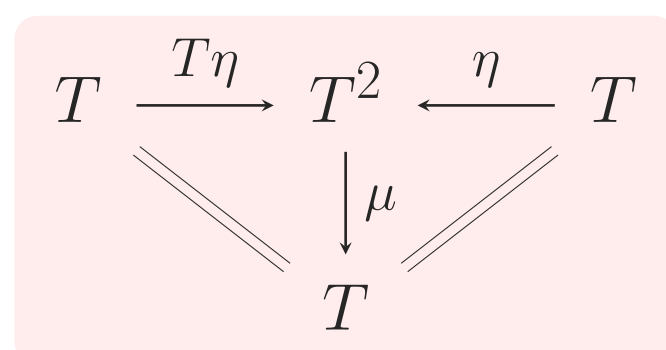
- Das Ziel: Semantik und Typsysteme im richtigen Leben verwenden
- Das Werkzeug: Der Beweis-Assistent Coq
- Das Motto: Beweise sind Programme und Programme sind Beweise!



Monad-Based Programming

```
class Functor m => Monad m where
  return :: a -> m a      -- η
  bind   :: m a -> (a -> m b) -> m b  -- μ

-- associative law: (fmap bind) = bind . bind
-- unit laws: (fmap unit) = bind . unit = id
```



Computational monads are a centerpiece of modern semantics and functional programming, prominently manifested in HASKELL and influencing SCALA, ML and F#. We provide

- a systematic theoretical and practical monad-oriented account of various computational aspects of program semantics
- a uniform monad-based presentation of such diverse computational effects as store, non-determinism, exceptions, and continuations
- an introduction to advanced theoretical and practical features of HASKELL programming

Seminar Koalgebraische Logik

Das Seminar führt ein in semantische und algorithmische Methoden der koalgebraischen Logik, die einen vereinheitlichenden Rahmen für modale Logiken in der Wissensrepräsentation und der Modellierung nebenläufiger Systeme bietet. Insbesondere deckt sie dabei Logiken jenseits der üblichen relationalen Semantik ab, z.B. für

- unsicheres oder ungefähres Wissen („Dieses Tier ist sehr groß“)
- zufälliges Verhalten („Dieses Programm wird wahrscheinlich terminieren“)
- Koalitionen von Agenten („Alice und der Print-Server können erreichen, dass mein Job gedruckt wird“)

