Implementation of Type Classes
and Type Constructor Classes
for PAKCS and KiCS2

Finn Teegen

Institut für Informatik, CAU Kiel, D-24098 Kiel, Germany,
`fte@informatik.uni-kiel.de`

Abstract. The programming language Curry combines multiple declarative programming paradigms, such as functional, logic, and concurrent programming. Its syntax and semantics are very similar to those of the programming language Haskell, but Curry lacks some of its more advanced features. One of them is Haskell's approach to ad hoc overloading, namely type classes and type constructor classes.

In this work, we illustrate our implementation of type classes and type constructor classes for two existing Curry systems: PAKCS[1] which compiles Curry to Prolog and KiCS2[2] that outputs Haskell code. We make use of the well-known dictionary passing technique which in turn requires us to annotate the abstract syntax tree with type information. An additional difficulty for the KiCS2 arises from the fact that its target language is strongly typed. While it is sufficient for the PAKCS to discard the type annotations after introducing the dictionaries, we keep them throughout all compilation phases in order to be able to generate correct Haskell code in the KiCS2.

---

[1] `https://www.informatik.uni-kiel.de/~pakcs/`

[2] `https://www-ps.informatik.uni-kiel.de/kics2/`