

free-theorems: Syntaxbaum

Thomas Rossow

September 8, 2016

1 BasicSyntax

Das Paket `free-theorems` ermöglicht das Parsen von Haskellcode in die selbstdefinierte Datenstruktur `BasicSyntax`. Diese Datenstruktur beinhaltet lediglich die Konstrukte, die für die Generierung von freien Theoremen benötigt werden. Intern nutzt das Paket die Parser aus `Language.Haskell.Exts` und transformiert den resultierenden Syntaxbaum in die `BasicSyntax`.

Im Folgenden soll an einem Beispiel gezeigt werden, wie der `BasicSyntax`-Baum aufgebaut ist.

1.1 Funktionssignaturen

```
test :: a -> b

[TypeSig
  (Signature {
    signatureName =
      Ident {
        unpackIdent = "test" },
    signatureType =
      TypeFun
        (TypeVar
          (TV
            (Ident {
              unpackIdent = "a"
            })
          )
        )
      (TypeVar
        (TV
          (Ident {
            unpackIdent = "b"
          })
        )
      )
    )
]
```

1.2 data

```
data Test a = Test String a

[DataDecl
  (Data {
    dataName =
      Ident {
        unpackIdent = "Test",
      },
    dataVars = [
      TV (Ident {unpackIdent = "a"})
    ],
    dataCons = [
      DataCon {
        dataConName = Ident {unpackIdent = "Test",
        dataConTypes = [
          Unbanged {
            withoutBang =
              TypeCon (Con (Ident {unpackIdent = "String"})) [],
          },
          Unbanged {
            withoutBang =
              TypeVar (TV (Ident {unpackIdent = "a"}))
          }
        ]
      }
    ]
  })
]
```