

Haskell – eine etwas andere Sprache



Wo kriege ich Haskell her?

- Vollständige Haskell-Umgebung
 `sudo apt-get install haskell-platform`
- Kleiner Interpreter zum Schnuppern
 `sudo apt-get install hugs`

Geschichte

- 1985: Miranda als erste funktionale Sprache
- 1990: Haskell 1.0
 - erster Version. Entwickelt von Komitee
- 1998: Haskell 98
 - erster verbreiteter Standard
- 2010: Haskell 2010
 - aktueller Standard, fügt Features hinzu

Was ist Haskell?

- Deklarativ-funktionales Paradigma
 - »Was ist x?« statt »Wie berechne ich x?«
 - Beispiel: Fakultät

```
long factorial(long x) {  
    long a;  
    for (a = 1; x > 0; a*=x--);  
    return a;  
}
```

```
factorial x = product [1..x]
```

Was ist Haskell?

- Deklarativ-funktionales Paradigma
 - keine Statements; alles ist ein Term
 - IO über spezielle Modellierung (Monaden)

Was ist Haskell?

- Bedarfsauswertung
 - Terme werden erst wenn nötig berechnet
 - unbenötigtes wird nicht berechnet
 - Unendliche Datenstrukturen sind möglich
`nums = [1..]` -- alle natürlichen Zahlen

Was ist Haskell?

- automatische Speicherverwaltung
- Alles ist konstant → gut zu optimieren
 - Zuweisungen sind eher Definitionen

```
foo = take 10 a where
```

```
  a = 0 : b
```

```
  b = 1 : a
```

Was ist Haskell?

- strenge, statische Typisierung
- Volle Typinferenz nach Hindley-Milner
 - Typdeklarationen fast immer unnötig
 - Ersetzt Templates
 - Wiederverwendbarer Code einfach

Wofür eignet sich Haskell?

- numerische Berechnungen
- Datenverarbeitung
- Webanwendungen
- Compilerbau

Wofür eignet sich Haskell weniger?

- Interaktive Anwendungen
- hardwarenahe Programmierung
- Algorithmen mit viel Mutabilität
- kleine Kompilate

Vorteile

- Kurze Programme
 - Windowmanager Xmonad: 2600 Zeilen Haskell
- Hohe Sicherheit durch starke Typisierung
- Modularität durch Vermeidung von Abhängigkeiten
- Automatisch threadsicherer Code
- Parallelisierung kann zum Teil automatisch erfolgen
- Hohe Effizienz durch gute Optimierbarkeit

Nachteile

- akute Gehirnverknötung
- Laufzeitverhalten von Programmen zum Teil schwer nachvollziehbar
- Effizienz, Zeit- und Platzkomplexität kann stark von Compileroptionen abhängen

Wohinter steckt Haskell?

- xmonad (Windowmanager)
- Pugs (Referenzimplementierung von Perl 6)
- Darcs (Versionsverwaltung)
- ghc (Glasgow Haskell Compiler)
- 4400 Programme und Bibliotheken verfügbar

Weiterführendes

- <http://haskell.org>
- <http://learnyouahaskell.com>
- <http://book.realworldhaskell.org>