

初学者のための高階関数に着目した Haskell ソースコード可視化手法

情報科学科 粟野 頁多

指導教員：大久保 弘崇

1 はじめに

本学では純粋関数型言語 Haskell を対象とした「プログラミング入門」が開講されているが、その受講者はプログラミング初学者であり関数型言語の特徴の一つである高階関数の存在からソースコードの理解や演習問題のコーディングが難しいと感じている。そこで、本論文ではソースコードの可視化とビジュアルプログラミングというアプローチでこの問題を解決する。

2 プログラムの可視化とビジュアルプログラミング

プログラムの可視化とは、ソースコードを視覚的にわかりやすく表示することである。既存の手法として Visual Haskell[1]がある。しかし、Visual Haskell は関数を第一級の要素として扱っておらず Haskell の特徴を無視している。即ち高階関数を一般性のある方法で扱うことができない可視化手法である。

一方、ビジュアルプログラミングは視覚的オブジェクトを用いてプログラミングを行う手法である。この手法の例として手続き型言語に基づく初学者向けの Scratch[2]がある。これは色や形が構文によって分かっているパズルのような視覚的部品を用いてプログラミングをする。

3 提案する可視化手法

本論文では関数を第一級の要素として扱うことができる Haskell の特徴を捉えた新しい可視化手法を提案する。初学者向けの支援のため Scratch の視覚的部品の手法を取り入れた。

3.1 基本方針

本手法において定数や変数はその値の持つ型によって形状が定まる。Bool 型の値を六角形で表す。その他の値は四角形で表す。

形状の規則は関数適用や演算子にも同様に適用される。例えば、図 2 全体は `sum xs` という Bool 型でない関数適用の式である。このように関数適用を、関数を表す形状に引数を表す形状をはめ込むことで表現する。よって一引数関数そのものは図 1 のように L 字形で表現する。

二引数関数は図 3 のように階段状になる。これは第一引数だけを部分適用した結果はカーリー化のため一引数関数であるが、それは図 4 のように一引数関数を意味する L 字形になる。そしてもう一つ引数が適用されると図 5 のように四角形になる。

3.2 高階関数

本手法では高階関数を図形で一般的に表現する規則を提案する。例として、`map (3+) xs` を考える。これを本手法を適用して可視化した結果を図 6 に示す。ここで高階関数はオレンジ色で表現し、その他の関数は紫色で表現する。



図 1 `sum` の表現

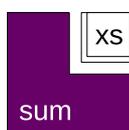


図 2 `sum xs` の表現

`map` の第一引数は L 字形で表現される一引数関数である。引数のブロックを関数にはめ込むメタファーを延長し、関数を引数にとる高階関数を、L 字形や階段状の形をはめ込むための欠けのある形状で表現する。また、引数の型を引数の欠け部に表示することで、どのように値が渡されるのかを推測できるようにした。このように、本手法では型情報に基づいて高階関数を可視化する。

4 評価

本章では以下の指針を用いて評価を行った。

1. 本手法の Haskell 構文の対応率を検証した。ここで、本手法は初学者向けであるので『すごい Haskell 楽しく学ぼう』[3]の第一章-第五章で評価を行った。
2. 本手法がどの程度初学者の理解支援につながっているかを測るために「プログラミング入門」(旧:計算機言語論)以降、Haskell の学習経験が無い者を対象にアンケートを行った。

5 おわりに

本論文では、Haskell ソースコードの可視化手法を提案した。関数を第一級の要素として扱ったことにより Haskell の特徴であるカーリー化などを視覚的にとらえることが可能になった。

今後の課題としては、データ型、`do` 構文への対応や本手法を「プログラミング入門」の講義に導入することなどが挙げられる。

参考文献

- [1] Hideki.John Reekie, “Visual Haskell:A First Attempt”, Reserch Report 94.5, Key Centre for Advanced Computing Sciences,University of Technology,Sydney, 1994
- [2] JOHN MALONEY, “The Scratch Programming Language and Environment”, ACM Transactions on Computing Education, Volume 10, Issue 4, No.16, 2010
- [3] Miran Lipovača 著, 田中英行・村主崇行 共訳, 『すごい Haskell 楽しく学ぼう』, オーム社, 2012.



図 3 `take` の表現

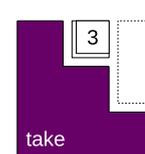


図 4 `take 3` の表現

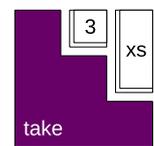


図 5 `take 3 xs` の表現

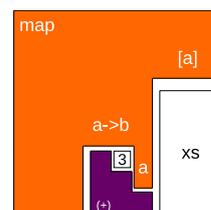


図 6 `map (3+) xs` の表現