

K-030

関数呼び出しのパターン分析による個人学習特性の同定

Identification of Personal Learning Characteristics
Using Pattern Analysis of Procedure Calls谷川 紘平[†]
Kohei Tanigawa稲葉 大祐[‡]
Daisuke Inaba原田 史子[†]
Fumiko Harada島川 博光[†]
Hiromitsu Shimakawa

1. はじめに

大学のプログラミング教育において、学習者はプログラミング言語とアルゴリズムを学習する。正しく動作するプログラムを作成するためには、これらの知識に加えて、プログラムの処理の流れを正しく理解する必要がある。しかし現状では、処理の流れが理解できていないために、思い通りのプログラムを作成できず、行き詰る学習者が多い。そのため、学習者に処理の流れを理解させるための演習が必要である。

プログラムの処理の流れに対する理解の仕方や過程は学習者によって異なる。そのため、指導者は学習者の特性に応じた指導が必要となる。現状では学習者の特性を把握するために、指導者は学習者の作成したソースコードを評価している。しかし学習者の理解度を把握するには、指導者はソースコードをていねいに読まなければならない。学習者の作成した最終結果であるソースコードから、学習者の理解の過程や行き詰る内容を見つけ出すことは難しい。学習者が試行錯誤を重ねた過程の形式的記録があれば、これを計算機で分析できる。

そこで本論文では、学習者のプログラミング演習における操作の履歴から得られる関数呼び出しの履歴を用いて、学習者の学習特性を同定する手法を提案する。これにより、指導者は学習者の特性に応じた指導ができる。

2. プログラミング初心者が行き詰る原因

プログラミング教育の初期段階において、学習者がプログラムを正しく作成するためには、処理の流れを理解することが重要である。しかし、プログラムの処理がどのように流れているかといったソースコードを記述するために必要な基本的な知識がプログラミング初心者には欠如している場合が多い。プログラミング初心者にとってそれらを理解することは容易ではなく、行き詰る学習者も少なくない。

処理の流れがイメージできることとプログラミングの理解度には関係がある[3]。プログラミング初心者にはプログラムを学習させるには処理の流れのイメージを理解させることが有効であるが、イメージの持ち方は人によって異なる。そのため、指導者は個々の学習者の特性に応じた指導が必要となる。

現状では、学習者のソースコードや操作履歴から学習者の理解を把握したり、行き詰る内容を発見することで学習者の理解を支援するようなシステムが研究されている[1][2]。しかし、課題達成までの学習過程から学習者の特性を推測し同定することで、学習者の特性に応じた指導を支援するような研究はなされていない。

3. 関数呼び出し履歴取得による学習特性同定

3.1 関数呼び出し履歴の取得

本論文では、学習者がプログラミングを学習する上で学習者の学習特性を推測し同定する手法を提案する。本手法では、学習者がプログラムを作成するときに行った操作履歴から得られる関数呼び出し履歴を分析することにより、学習者の学習特性を同定する。指導者は、学習者に図を描画させる課題を出題する。その課題を作成するさい、指導者は関数呼び出し履歴からプログラムの制御や処理内容が推測できるように課題を作成する。それにより、コンパイルエラーの解析やソースコードの解析のような複雑な処理をせずに関数呼び出し履歴を分析するだけで、学習者の学習特性を推測することができる。

本手法の概要を図1に示す。本手法を用いた演習において、学習者は、指導者の提示する図と同様の図を描画することを目標にソースコードの作成をする。そのとき、学習者はGUI出力のために指導者が用意した関数である描画命令関数とGUI出力用のツールであるGUIツールを用いる。描画命令関数を用いて作成されたソースコードをコンパイルし、そのプログラムを実行することで、GUIツール上に図が描画される。図の描画のために呼び出された描画命令関数の呼び出し履歴がGUIツールによって記録される。

描画命令関数の呼び出し履歴を用いて、学習者の学習特性が推測される。これにより、指導者は学習者の学習特性を把握でき、個々の学習者の学習特性に応じた指導をすることができる。学習者自身の学習特性に応じた指導により、学習者は理解を深めることができる。

3.2 GUIによる出題

指導者は、学習者に演習課題として目標となる図を出題する。指導者は図を用いた演習課題を与えることで、学習者に図を描くために必要な制御と描画命令関数を

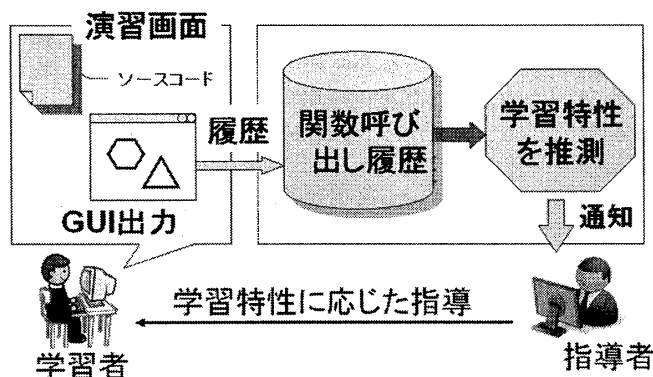


図1: 手法の概要

[†]立命館大学 情報理工学部[‡]立命館大学大学院 理工学研究科

推測させる。それにより、学習者の課題に対する解答パターンを絞り込むことができ、指導者は本手法を用いたプログラミングの演習から得られる関数呼び出し履歴を想定しやすくなる。

指導者は、本手法を用いた演習から得られる関数呼び出し履歴を想定し、その履歴から学習者が作成したプログラムの制御と処理内容を推測できるような課題を考案する。たとえば、引数で指定された長さの直線を描画する関数と引数の値に応じて直線の色を変える関数を用いて、「6辺を順番に3色で塗り分けた六角形の渦巻きを描く」という課題を出題した場合を考える。この場合、直線を描画する関数と直線の色を変える関数が交互に呼び出され、それぞれ6回、3回の周期があると想定される。

3.3 関数呼び出し履歴

学習者がGUIツールと描画命令関数を利用し、ソースコード作成の演習をすることにより、描画命令関数の呼び出し履歴が関数呼び出し履歴として記録される。関数呼び出し履歴には、呼び出された関数の種類とその引数・関数の呼び出し順・関数の呼び出された時刻が記録される。学習者がソースコード作成途中で試験的に実行したさいの関数呼び出し履歴もすべて記録される。これにより、学習者の学習特性を同定するために必要となる課題達成までの演習過程を収集することができる。

3.4 学習特性の同定

本手法では、関数呼び出し履歴を分析し、指導者が期待する関数呼び出しの履歴と学習者のプログラミング演習履歴から取得した関数呼び出し履歴を比較することで学習特性が同定される。学習特性を同定することにより、学習者の間違えのパターンや演習過程が分かる。

図2に、演習課題として指導者が学習者に提示した図とある学習者が演習過程の中で描画した図を示す。たとえば、学習者が図2の「指導者が提示する図」を描画することを考える。この図は小さな渦巻きが右回りに6箇所に表示されている図である。直線のみで1つの渦巻きを描画するためには反復処理が必要である。この図を描画するには、1つの渦巻きを描画してから次の渦巻きの中心まで直線を描画するという処理を6回反復させる2重ループ構造が必要である。しかし、次の渦巻きの中心まで直線を描画するという処理をしていない場合は図2の「ある学習者の途中経過」のようになる。この場合、指導者の期待した関数呼び出しと比べて、学習者から取得した関数呼び出し履歴には次の渦巻きの中心まで直線を描画するための描画命令関数の履歴が記録されていないため、このような間違えを発見することができる。

関数呼び出し履歴から課題達成率・所要時間・実行回数・実行毎のソースコードの修正時間・実行毎の関数呼び出し履歴の変化量といった学習者の演習過程を求めることができる。課題達成率とは、課題に設けられたチェックポイントの達成率を表すものである。個々の学習者の履歴から得られたこれらの多次元のデータについて、プログラミングについての正しい知識をもつ学習者、機能を少しずつ実現し結果をみてソースコードを修正していく学習者、手を動かさない学習者など、多様な学習者のそれぞれについて、その特性を示すグラフを選び出す。これにより個人の学習特性を同定する。

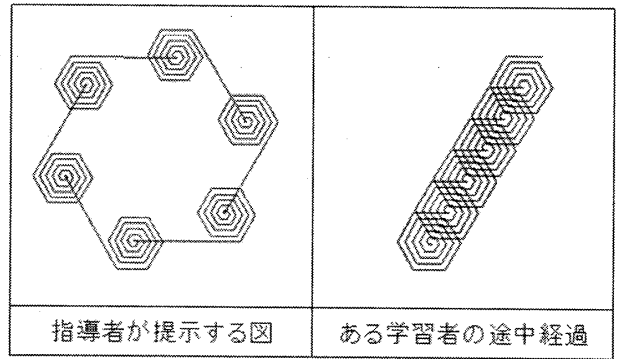


図2: 指導者が提示する図とある学習者が作成した図

4. 関連研究

関連研究として、ソースコードを解析することで学習者のプログラミングスタイルを推測するシステム [1] やデバッガやコンパイラが出力するエラーを解析し、エラー箇所や原因を判断するシステム [2] が挙げられる。関連研究 [1] では、構文解析や意味解析といった複雑な処理が必要となり、学習者の特性を得ることは難しい。関連研究 [2] では、デバッガやコンパイラが出力するエラーを解析している。そのため、ソースコードの文法は正しいがアルゴリズムが間違っているために、正しい実行結果が得られない学習者を発見することができない。また、課題を達成するまでの過程に学習者の特性が表れると考えられるが、ソースコードを解析したり、デバッガやコンパイラのエラーを解析するだけでは指導者がその過程を容易に把握することはできない。

本手法では、指導者が演習後に学習者から得られる関数呼び出し履歴を想定し課題を作成することによって、関数呼び出し履歴だけに着目して容易に学習特性を得ることができる。

5. おわりに

本論文では、GUI出力ツールと描画命令関数を用いた演習から得られる関数呼び出し履歴を分析することにより、学習者の学習特性を同定する手法を提案した。今後、本手法を用いた演習とその評価をする予定である。

参考文献

- [1] 鈴木 恵, 横山 節雄, 宮寺 庸造, "プログラミング教育のためのモジュール自動分割システムの開発と評価", 電子情報通信学会技術研究報告, Vol.103, No.697, pp.143-148, Feb. 2004.
- [2] 西 輝之, 劉 渤江, 横田 一正, "デバッガとの連携によるC言語学習支援システムの提案(新しいインターネット技術の教育環境への利用/一般)", 電子情報通信学会技術研究報告, Vol.106, No.583, pp.173-178, Mar. 2007.
- [3] 長谷川 聡, 山住 富也, "プログラミング教育と学習者のイメージ形成(その2)", Journal Article, Vol.23, pp.9-14, Apr. 1998.