

プログラミング演習における行き詰まり検出器と静的解析器を利用した 個別指導支援システムの構築

Development of an Individual Instruction Support System Using Impasse Detector and Static Analyzer for Programming Exercises

池亀 智紀^{*1}, 野口 靖浩^{*2}, 小暮 悟^{*2}, 山下 浩一^{*3}, 山本 頼弥^{*4}, 小西 達裕^{*2}, 伊東 幸宏^{*5}
Tomoki Ikegame^{*1}, Yasuhiro Noguchi^{*2}, Satoru Kogure^{*2}, Koichi Yamashita^{*3}, Raiya Yamamoto^{*4},
Tatsuhiko Konishi^{*2}, Yukihiko Itoh^{*5}

^{*1} 静岡大学大学院総合科学技術研究科

^{*2} 静岡大学情報学部 ^{*3} 常葉大学経営学部 ^{*4} 山陽小野田市立山口東京理科大学工学部 ^{*5} 静岡大学

^{*1} Graduate School of General Science and Technology, Shizuoka University

^{*2} Faculty of Informatics, Shizuoka University ^{*3} Faculty of Business Administration, Tokoha University

^{*4} Faculty of Engineering, Sanyo-Onoda City University ^{*5} Shizuoka University

Email: ikegame.tomoki.17@shizuoka.ac.jp

あらまし：プログラミングの演習形式の授業では、教師や TA が机間巡視を行いながら学生に個別指導を行うことが多いが、行き詰っている学生を発見することやプログラム中のどの部分で行き詰っているのかを瞬時に特定することは難しい。この問題を解決するために学生の演習情報から検出した行き詰まり情報と、静的解析によって行き詰まりが正解プログラムのどの部分にあたるかという情報をプログラム中に表示するシステムを構築した。

キーワード：プログラミング、個別指導支援、机間巡視支援、行き詰まり検出、教師支援

1. はじめに

プログラミングの演習形式の授業では、少数の教師や TA が机間巡視を行いながら、多数の学生の質問に個別指導のような形で対応する場合が多い。しかし、この形式には以下に示すような問題点がある。

問題点 1: 行き詰まり学生を発見することの困難

学生の演習を妨げずにソースコードやエラーメッセージを確認するシステムは存在する⁽¹⁾が、ソースコードやエラーメッセージのみを見ただけでその学生が行き詰まっているかを判断することは困難である。

問題点 2: 行き詰まり箇所を特定することの困難

行き詰まりを起こしている学生は自分自身でソースコード上の行き詰まり箇所を説明することが難しいため、その学生がどの部分で行き詰まっているのかを特定することが困難である。

問題点 3: クラス全体の行き詰まり状況を把握することの困難

少数の教師や TA が多数の学生に指導を行うため、クラス全体として学生がソースコードのどの部分で、どのような内容で行き詰まっているかを把握することは困難である。

先行研究のシステムである行き詰まり検出器⁽²⁾は、学生の一定時間ごとのソースコード、コンパイルごとのソースコード、コンパイルエラー、実行ログといった演習情報を収集し、解析することによって学生が行き詰まりを検出し問題点 1 の解決策となる。

同じく先行研究のシステムである個別指導支援情報の抽出⁽³⁾は、行き詰まり検出器の検出結果から行き詰まりを起こした学生プログラムの行（行き詰ま

り情報 Lv.1) を検出し、静的解析によって行き詰まりを起こした行に対応する正解プログラムの行（行き詰まり情報 Lv.2) を検出し、正解プログラムの各行に学習項目を設定しておくことで行き詰まりを起こした行に対応する学習項目（行き詰まり情報 Lv.3) を検出する。このように、より詳細な行き詰まり情報を 3 段階で検出することで問題点 2 の解決策となる。

先に挙げたように、先行研究のシステムによって、問題点 1, 2 に対する解決策が講じられた。しかし、問題点 3 の解決策となるシステムは存在しない。また、行き詰まり検出器や個別指導支援情報の抽出は検出結果がテキストベースであるため、視覚的に行き詰まり情報を得ることが困難といった問題もある。

そこで本研究では、行き詰まり検出器や個別指導支援情報の抽出で得られた検出結果を個別指導支援システム⁽⁴⁾というプログラミング演習における個別指導を総合的に支援するシステムで GUI 表示するという形でシステムを統合する。このシステムの実現によって、問題点 1, 2 に対しては行き詰まり情報の視覚化によって教師や TA にとってよりわかりやすい解決策となる。問題点 3 に対しては、行き詰まり情報 Lv.3 をクラス全体で集約し表示することでクラス全体の行き詰まり状況を把握することができると考えられるため、解決策になる。

2. システム統合後の指導の流れ

3 つの先行研究のシステムを統合することによって実現される指導の流れと機能によって軽減される問題点を以下のように提案する。

1. 学生一覧に行き詰まり学生を強調表示（問題点1の軽減）
2. プログラム閲覧画面に行き詰まりを起こした行と対応する正解プログラムの行を強調表示（問題点2の軽減）
3. 行き詰まり情報 Lv.3 を集約し、表示することでクラス全体の行き詰まり状況を把握（問題点3の軽減）

3. 実装したシステムの機能

先の指導の流れを実現させるために、次に示す機能を実装した。

3.1 学生一覧から行き詰まり学生の表示

行き詰まりを検出した学生を、個別指導支援システムの学生一覧表示画面に強調表示する。この機能によって、学生を一人ひとり確認することなく、行き詰まり学生を瞬時に発見することができる。

3.2 行き詰まり情報 Lv.1 の表示

個別指導支援システムの学生プログラム表示画面の行き詰まりを起こした行の行番号の左にボタンを配置する。このボタンをクリックすることで、その行き詰まりがどのような内容であるのかを表示する。この機能によって、行き詰まり情報 Lv.1 を視覚的に得ることができるようになる。

3.3 行き詰まり情報 Lv.2 の表示

行き詰まり情報 Lv.1 のときに配置したボタンをクリックすることで、そのクリックしたボタンに対応する学生プログラムと正解プログラムの行が強調表示される。この機能によって、行き詰まり情報 Lv.2 を視覚的に得ることができるようになる。

3.4 クラス全体の行き詰まり情報の表示

ある課題に対応する正解プログラムとその各行に対応する学習項目はすべての学生に共通であるため、行き詰まり情報 Lv.2 と Lv.3 はクラス全体で情報を集約することができる。集約した行き詰まり情報は図1のように表示される。

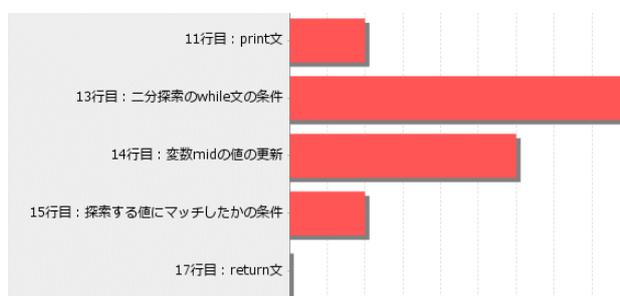


図1 全体の行き詰まり情報の表示部（抜粋）

各項目は、行き詰まりを起こした行とその行に対応する学習項目であり、横軸は人数を表す。この表示によって、クラス全体で正解プログラムのどの部分で行き詰まりを起こしているのか、その学習項目は何か、という情報を視覚的に得られるようになる。

4. システムの予備評価

本システムの有効性を評価するために、教師1名とTA経験者の学生3名に対して予備評価を実施した。被験者は、あらかじめ集めた学生の演習情報の入ったシステムを利用し、行き詰まり学生の発見、行き詰まり箇所の特定、クラス全体の行き詰まり状況の確認、といった課題を行い、どういった指導を行うべきか記述してもらった。

評価結果として、被験者4名全員にシステムの提示する情報から課題を行うような記述がみられた。また、3つの問題点の軽減に関する項目に対して4段階評価のアンケートを実施した。その結果を表1に示す（数値が高いほど軽減になる）。

表1 アンケート結果（1～4点）

	問題点1	問題点2	問題点3
被験者A	4	3	4
被験者B	4	4	3
被験者C	4	4	4
被験者D	4	4	2
平均	4.00	3.75	3.25

表1より、どの問題点の軽減に関する項目に対しても3以上の評価を得ている。したがって本研究で構築したシステムは、問題点1～3の軽減に有効であると考えられる。一方で、アンケートの自由記述欄には、学生の過去の行き詰まり情報も見ることができれば、より学生の状況を把握することができるとの回答も見られた。

5. まとめ

本研究ではプログラミング演習中の学習者の行き詰まりの把握に関する3つの問題点を解決するためのシステムを構築した。予備評価の結果から、システムの有効性が示された。

今後の課題として、過去の行き詰まり情報の表示機能やオンライン・オンデマンドでの授業に対応することなどが挙げられる。

参考文献

- (1) 安留 誠吾: “TA 活動支援のための共有ホワイトボード”, 情報処理学会, インタラクション 2019 論文集, pp.438-443 (2019)
- (2) K.Yamashita, T.Sugiyama, et al.: “An Educational Support System Based on Automatic Impasse Detection in Programming Exercises”, Proceedings of ICCE2017, pp.288-295 (2017)
- (3) Y.Noguchi, K.Ayabe, et al.: “Experimental Design of Automated Extraction for 3-Level Tutoring Support Information in Programming Exercises”, Proceedings of ICCE2020, pp.255-260 (2020)
- (4) Y.Noguchi, R.Osawa, et al.: “Networked Tutoring Support System for a Programming Class based on Reusable Tutoring Content and Semi-automatic Program Assessment”, Proceedings of ICCE2016, pp.252-257 (2016)