

プログラミング演習における 指導が必要な学習者の自動検出

河瀬雅弥^{*1}, 北英彦^{*1}, 高瀬治彦^{*1}

^{*1} 三重大学工学研究科電気電子工学専攻

Automatic detection of learners who need guidance in programming exercises

Masaya Kawase^{*1}, Hidehiko Kita^{*1}, Haruhiko Takase^{*1}

^{*1} Department of Electrical Engineering, Graduate School of Engineering, Mie University

The authors have developed PROPEL, a system for beginner programming exercises, and use it in their classes. One of its features is to suggest to instructors and TAs which learners to teach based on the presence or absence of compilation errors in the learners' code and the time they failed to resolve them. In addition to this, this time, based on the code creation history of exercises conducted before last year, the characteristics of learners who failed to complete their codes are extracted, and learners who should be taught are detected based on these characteristics.

キーワード: プログラミング, 初学者, 講師支援

1. はじめに

コンピュータの発展に伴い、さまざまな分野でプログラミング能力が必要とされている。能力を身に着けるため、実際にプログラムを作成して理解を深めるプログラミング演習が著者らの大学で行われている。しかし、現在行われているプログラミング演習は講師の数が学習者に比べて少なく、指導が必要な学習者に十分な指導が行われていないと考えられる。これまでも著者らの所属している研究室で作成されたプログラム演習システム PROPEL を用いて、指導が必要な学習者を探す研究がされていたが、十分ではない。本研究の最終目的は、演習中に躓いている学習者を見つけ出し講師に指導する学生を提案するシステムを構築することである。

2. プログラミング演習システム PROPEL

著者らの所属する研究室では、プログラミング演習を目的としたシステムを開発・運用してきた。そのうちのひとつが学習者のプログラミング作成状況への迅速

な対処を目的としたプログラミング演習システム PROPEL である。図 1 に PROPEL のシステム構成を示す。PROPEL を用いることで、プログラミング演習において、プログラムの作成に行き詰っている学習者を早い段階で見つけ出し講師がアドバイスすることが可能となっている。また、講師も学習者全体の理解度を容易に把握することができ、プログラミング演習の質の向上と効率が可能である。図 1 に PROPEL のシステム構成を示す。また、図 2 にシステムの内部構成を示す。

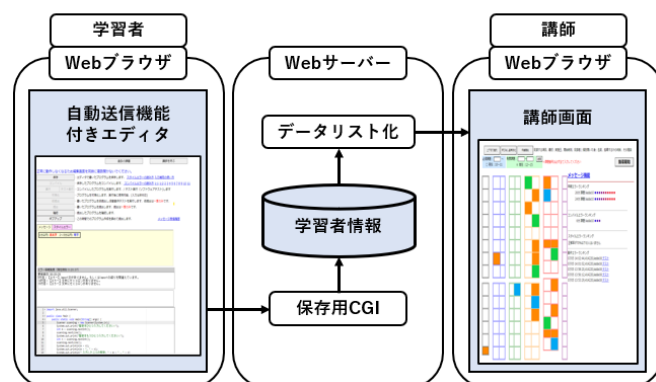


図 1 PROPEL のシステム構成

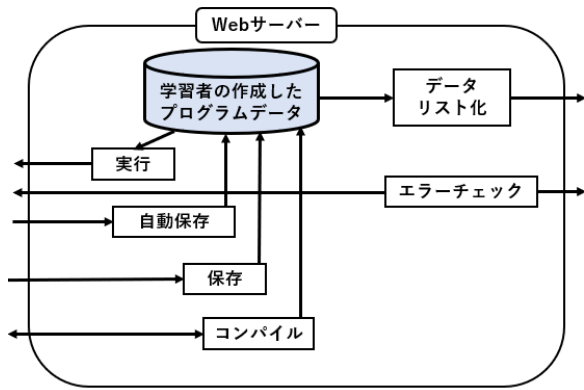


図 2 システムの内部構成

2.1 現状の講師画面

図 3 に PROPEL で講師が利用する画面を示す。図 3 では学生の名前と課題は伏せてある。画面の左半分は講義の際の座席表を表示している。学習者が「講師を呼ぶ」ボタンを押すと席が赤く変化するので、講師を呼んでいることが分かるようになっている。

画面の右半分は学生のプログラム作成状況、すなわち、コンパイル前に 1 行単位でわかる早期エラー、コンパイルエラー、コンパイルエラー、スタイルエラーを残存時間の長い順で表示している。これにより待機時間が長い学生から順番に対応できる。



図 3 現状の講師画面

2.2 現状の問題点

現状の問題点を挙げる。

- ・学習者の進捗過程が把握できない。
- ・課題を諦めてしまふ学習者を判別できない。

1 点目については、その時点でのプログラム作成状況のみ表示されるシステムになっているため、視覚的に判断しづらく、指導に要する時間が長くなってしまふ。

2 点目については現状のシステムでは課題を諦めてしまふ学習者を提案することが出来ず、指導することができない。

2.3 新たに提案するシステム

提案するシステムの画面を図 4 に示す。システムはタブレット端末で運用する想定で作成した。システムでは学習者の進捗状況を縦軸にソースコードの行数、横軸に経過時間としたグラフを全学習者分表示し、グラフをタップすることによりその時点でのソースコードのスナップショットを表示する。また、直近に起こしたコンパイルエラーや課題の進捗度を全学習者でソートすることが出来る。

このシステムによって、学習者の進捗度を視覚的に判断でき、迅速な指導が出来るようになると考えている。

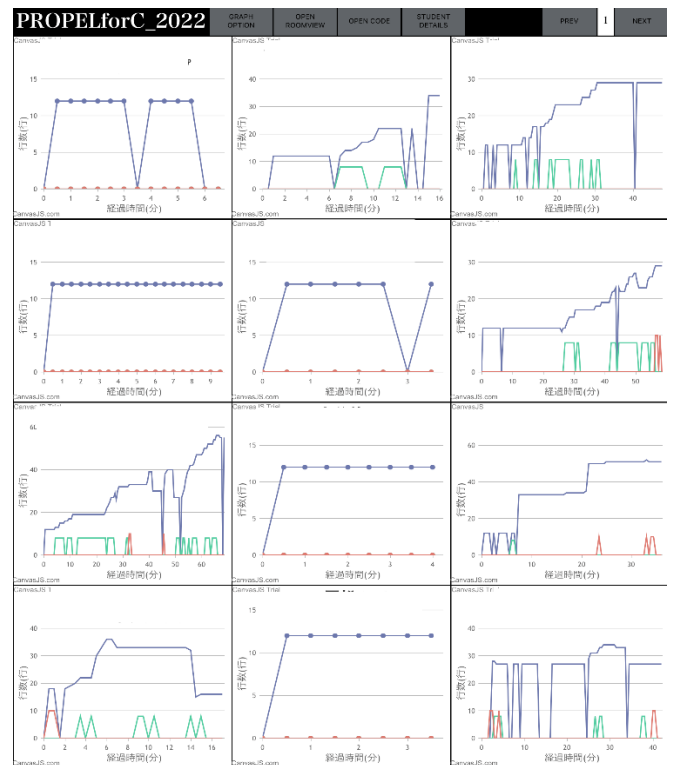


図 4 提案するシステムの画面

3. まとめ

今回は問題点の一つである「学習者の進捗過程が把握できない」と言う点を解決すべくシステムを作成した。しかし、もう一つの問題点「課題を諦めてしまう学習者の判別できない」という点は現時点では解決することが出来ていない。今後の課題としては、如何にして課題を諦めてしまう学習者を抽出することである。

参 考 文 献

- (1) 伊富昌幸, 小島佑介, 高橋功欣, 北英彦: プログラムの作成状況を把握する機能を持つプログラミング演習システム, 2010PC カンファレンス (2010)
- (2) 小川正, 西口大亮, 北英彦: プログラミング演習における iPad などの携帯デバイスの利用による指導の円滑化, PC カンファレンス (2011)
- (3) 小島佑介, 高橋功欣, 北英彦: プログラミング演習における効率のよい指導のためのエラー早期指摘, コンピュータ利用教育協議会, PC カンファレンス (2011)
- (4) 森一樹, 田中昂文, 橋浦弘明, 樋山淳雄, 古宮誠一: プログラミング演習支援のための細粒度履歴収集環境の開発, 情報処理学会 (2013)
- (5) 杉本優太, 伊藤福晃, 北英彦, 高瀬治彦: プログラム動作の可視化によるプログラムの動作確認支援に関する研究, コンピュータ利用教育協議会, 2019PC カンファレンス (2019)