

# 高校生を対象としたC言語学習時の編集履歴による学習状況の分析

石田崇<sup>†1</sup> 梅澤克之<sup>†2</sup> 齋藤友彦<sup>†3</sup> 中澤真<sup>†4</sup> 平澤茂一<sup>†5</sup>  
 高崎経済大学<sup>†1</sup> 湘南工科大学<sup>†2</sup> 東京都市大学<sup>†3</sup> 会津大学<sup>†4</sup> 早稲田大学<sup>†5</sup>

## 1. はじめに

著者らはこれまでに、WEB ブラウザ上でプログラミング学習を行いその際の編集状況の詳細なログを取得し把握するための編集履歴可視化システムの開発を行ってきた[1][2]。本発表では、高校生のC言語プログラミング学習の際にこのシステムを使用し、そこで収集された編集履歴情報、及びプログラミング学習前に行った基礎学力に関するテストの結果、学習前後に実施したアンケートの結果を総合的に分析し、学生の特性と学習行動との関係性について検証を行った結果を報告する。

## 2. プログラムの学習環境と編集履歴可視化システム

本研究で用いているC言語学習システム[1][2]では、学習者はWEBブラウザ上でプログラミングを行い、実行結果もそこに表示される。このシステムではOSに依存しないシンプルなプログラムの実行環境を提供している(図1参照)。さらに実行環境導入の手間も省けることでC言語の初學者の負担を減らしている。また、サーバ上に学習履歴が残ることから、学校や自宅などどこにいてもこれまでの学習内容を参照しながら学習することが可能である。

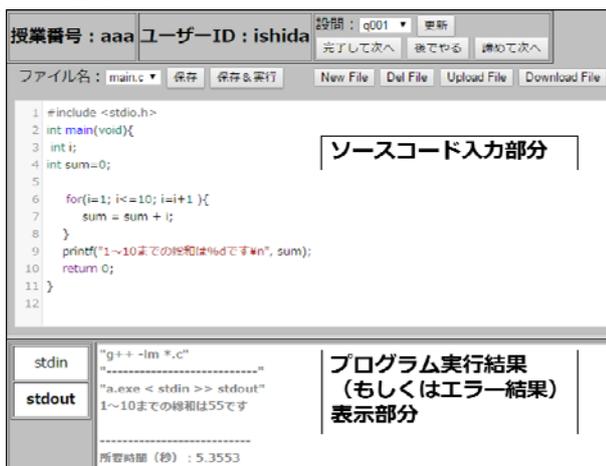


図1 実行環境 (学習者側)

An analysis of learning state by editing history during C language learning for high school students.

<sup>†1</sup> TAKASHI ISHIDA, Takasaki City University of Economics

<sup>†2</sup> KATSUYUKI UMEZAWA, Shonan Institute of Technology

<sup>†3</sup> TOMOHIKO SAITO, Tokyo City University

<sup>†4</sup> MAKOTO NAKAZAWA, The University of Aizu

<sup>†5</sup> SHIGEICHI HIRASAWA, Waseda University



図2 編集履歴確認画面 (教員側)

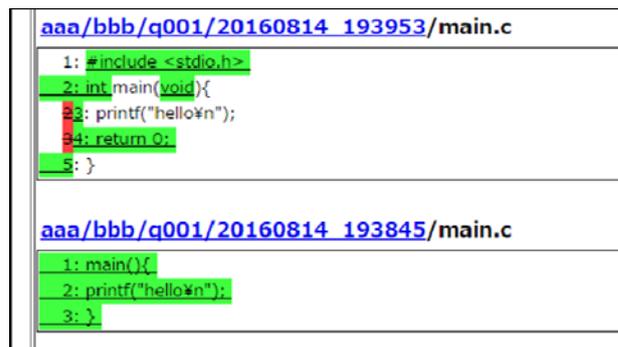


図3 編集履歴の差分表示

一方、教員の側では学習者全体のプログラミングの編集履歴や実行結果の履歴が分かりやすく可視化されて表示されるようになっており、学習者の進捗状況をリアルタイムで確認できる、学習者のつまずきのポイントを把握できる、ソースコードと一緒に実行結果の履歴も確認できる、といった特徴がある(図2参照)。

このシステムでは編集の際に削除された部分と新規で追加された部分が色塗りや取り消し線で可視化され、編集の差分が把握しやすくなっている。これにより学習者のプログラム作成過程をきめ細かく捉えることができる(図3参照)。

## 3. 学習履歴データの分析

### 3.1 プログラミング課題

今回の分析では2016年8月に新潟県の高校生を対象に開催したプログラミング実習[3]を受講した生徒18名の学習履歴データを用いた。

学習者らは C 言語の初歩的な事項について一通り説明を受けた後で, for 文を用いた printf 文の反復処理や整数の総和計算, if 文による整数の偶数奇数判定などの処理をサンプルプログラムを実際に実行しながら学習した. さらにサンプルプログラムを改良したり組み合わせたりすることで作成が可能な次の 4 つの課題に取り組んでもらった.

- (1) 課題 1 : for 文で反復回数だけを変更するプログラムの作成
- (2) 課題 2 : for 文でカウンタの刻み幅を変更するプログラムの作成
- (3) 課題 3 : for 文を用いた整数の総和計算で整数の範囲だけを変更するプログラムの作成
- (4) 課題 4 : if 文と組み合わせて奇数だけの総和を求めるように変更するプログラムの作成

なお, プログラムの演習とは別に数学などの教科についての簡単な基礎学力テストや実習に関するアンケートも実施している.

### 3.2 学習履歴の集計結果と考察

ここでは, 課題 1 と課題 2 の学習履歴データについて集計した結果を示す (図 4, 図 5). これらの図では横軸が 18 名の学習者, 縦軸は時刻であり各課題におけるコンパイルを行った時刻 (1 回目 (×印) とプログラムの完成時 (◆印)) がマークされている. またラベルの数値は課題が完了するまでのコンパイル回数を表している. なお, 横軸の学習者 1~18 は, 数学の基礎学力テストの得点の降順に並んでいる.

学習履歴データの分析から以下のような知見が得られた.

- (a) 課題 1 では全ての学習者が 1 回のコンパイルで課題を完成させた. これは for 文で 5 回の反復処理を 10 回に書き換えるだけの課題であり容易な課題であった. ただし, 課題を完了するまでの時刻には差が見られる. 数学の得点上位者の方が完了時刻が早い傾向がある.
- (b) 課題 2 は課題 1 をさらに書き換えて, for 文のカウンタの初期値を 10 として 10 刻みで 100 まで反復処理を行わせる課題である. 書き換えのポイントは 3 点あり, for 文において (1) 初期値を  $i=1 \rightarrow i=10$  へ, (2) 終了条件式を  $i \leq 10 \rightarrow i \leq 100$  へ, (3) 更新式を  $i=i+1 \rightarrow i=i+10$  とする. これら 3 つを同時に修正し, 1 回で課題を完了した学習者が 6 名いた. これらはほぼ数学の得点上位者であった. また, 数学の得点が下がるにつれて課題の完了時刻が遅くなり, 実行回数も増えてくる傾向が見られる.
- (c) 課題 3, 4 についても同様の傾向が見られる. 課題がやや複雑になってくると, 初回のコンパイルまで時間をかけて吟味して少ないコンパイル回数で課題を完了する学習者や, 一部を修正する度にこまめにコンパイルして課題を進める学習者な

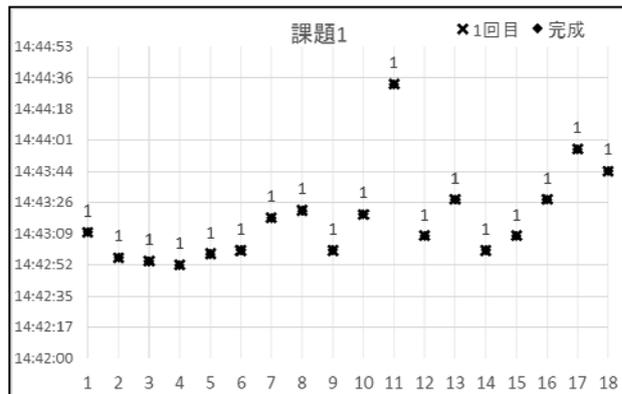


図 4 コンパイル実行時刻と回数(課題 1)

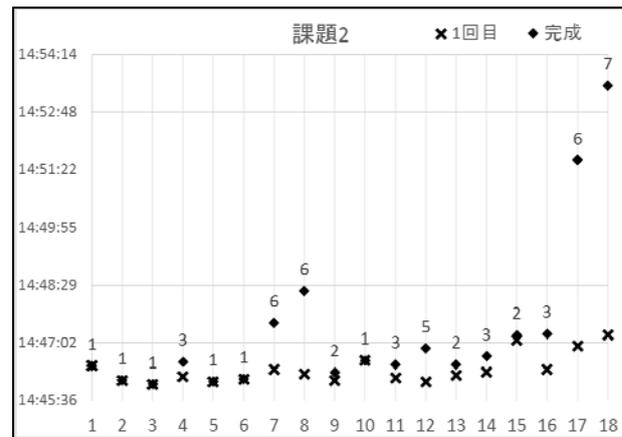


図 5 コンパイル実行時刻と回数(課題 2)

どがいて, プログラム作成の過程にいくつかの特徴が見られた.

## 4 おわりに

本稿ではプログラミング学習履歴データの分析を行い, 学習者の個人特性とプログラミングスキルとの関連性や, プログラム作成過程の個人差, パターンの発見などの示唆が得られた. 今後はこれらの知見を実際の授業の中で活用していく方法についても検討したい.

### 謝辞

本研究の実施にあたって新潟県立松代高等学校校長 長谷川雅一先生, (有)早稲田松代協力会代表 木戸一之氏には数々のご支援を頂いた. 本研究の一部は独立行政法人日本学術振興会学術研究助成基金助成金基盤研究(C)16K00491の助成による.

### 参考文献

- [1] 小林学, 後藤正幸, 荒本道隆, 平澤茂一, “プログラミング編集履歴可視化システムとその実践,” 日本経営工学会 2015 年秋季大会, (2015.11)
- [2] 荒本道隆, 小林学, 中澤真, 中野美知子, 後藤正幸, 平澤茂一, “編集履歴可視化システムを用いた Learning Analytics~システム構成と実装,” 情報処理学会第 78 回全国大会予稿集, pp.4-527-4-528, 横浜, (2016.3).
- [3] 梅澤克之, 中澤真, 石田崇, 齋藤友彦, 平澤茂一, “高校生を対象とした学習時の閲覧編集履歴と生体情報の収集とその分析,” 経営情報学会(JASMIN) 2016 年秋季全国研究発表大会, D2-1 (2016).