

## 学習履歴を活用したプログラミング学習支援システムのための基盤環境の開発

### Development of a Base Environment for the Programming Learning Support Systems Utilizing Learning History

大橋 旭雄<sup>\*1</sup>, 野崎 要<sup>\*1</sup>, 森本 康彦<sup>\*1</sup>, 中村 勝一<sup>\*2</sup>, 宮寺 庸造<sup>\*1</sup>

Akio OHASHI<sup>\*1</sup>, Kaname NOZAKI<sup>\*1</sup>, Yasuhiko MORIMOTO<sup>\*1</sup>, Shoichi NAKAMURA<sup>\*2</sup>, Youzou MIYADERA<sup>\*1</sup>

<sup>\*1</sup>東京学芸大学 <sup>\*2</sup>福島大学

<sup>\*1</sup>Tokyo Gakugei University <sup>\*2</sup>Fukushima University

Email: m133302s@st.u-gakugei.ac.jp, miyadera@u-gakugei.ac.jp

あらまし：本研究では、学習中に発生する学習履歴の活用によるプログラミング学習支援システムのための、プログラミング学習基盤環境を開発した。基盤環境により学習履歴の収集と一元的な管理を行うことで、支援システムによる学習履歴を共有・活用したプログラミング学習者/教授者の支援を可能にした。本稿では、本基盤環境及びこれを用いた本学でのプログラミング学習支援の現状について述べる。

キーワード：プログラミング学習, 学習基盤環境, 学習履歴, 学習支援

#### 1. はじめに

初学者のプログラミングの学習には様々な困難が生じ、学習が思うように進まないことが多い。そのため、初学者の学習には様々な支援が必要とされる。

我々は以前より、学習中に発生する学習履歴の活用によりプログラミング学習を支援することを考え、支援に用いるシステムの開発を行っている。これに先立ち我々は、これらシステムのために学習履歴の収集を行うプログラミング学習基盤環境を開発してきた。本稿では、基盤環境の開発の経緯や機能、基盤環境を活用した支援の現状について述べる。

#### 2. プログラミング学習基盤環境

##### 2.1 要件

本学の初学者向けプログラミング授業では、C 言語文法の習得を主としたプログラミングの基礎を学習する。C 言語の開発環境としては統合開発環境 (IDE)、Web インタフェース、Windows アプリケーション等々あるが、本学では標準的な UNIX 環境上でコマンド操作を用いたプログラミングを行わせている。これは、情報技術者育成の観点から、UNIX やコマンドラインインタフェース (CLI) に触れるべきである、というポリシーによる。このポリシーを踏襲しつつ、学習履歴を活用した効果的な支援という目的を達成するための基盤環境の要件として、学習者に対して特別な意識や操作をさせることなく、また学習を阻害することなく、発生するあらゆる種類の学習履歴をリアルタイムに収集出来ることが挙げられる。

##### 2.2 設計・開発

2.1 節で述べた要件に基づき、基盤環境の設計・開発を行った。環境の概要を図 1 に示す。基盤環境は、UNIX と CLI を用いた C 言語開発環境や学習履歴収集機能を備えたクライアント部と、授業におけるポータルサイトと収集した学習履歴を蓄積するデータベースを備えたサーバ部に大別される。

本学の学習者は全員 Windows のノート PC を所持しているため、クライアント部は VMware Player による Linux (Debian) の仮想マシン環境とした。ここに学習履歴収集機能を搭載し、学習履歴の収集を行う。仮想マシンを用いることで、Windows 上に UNIX 環境を容易に導入でき、また場所や時間、ネットワークの状態等の制約を受けない学習が可能になる。導入の容易な UNIX 環境としては他に KNOPPIX 等の Live CD があるが、Live CD では学習履歴収集のための各種設定やソースコード等ファイルの保存が困難である。仮想マシンは事前に設定済みのイメージを配布するだけで良く、ファイル保存も物理マシンと同様に行えるため、こちらを採用した。

サーバ部は一般的な LAMP 環境を採用し、様々な支援システムが本基盤環境と連携し、学習履歴を共有・活用することが出来るように配慮した。

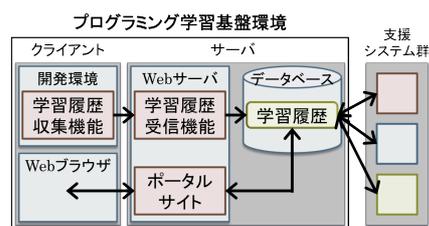


図 1 プログラミング学習基盤環境の概要

#### 3. 学習履歴の収集

##### 3.1 収集する学習履歴と収集の流れ

本基盤環境は 2004 年度より継続的に運用を行い、改良を重ねてきた。改良の過程で学習履歴の種類は増え、現在では数多くの種類が収集出来ている。

以下、学習者の活動に沿った学習履歴の収集について述べる。授業では、学習者はまず仮想マシン内の Web ブラウザを使ってポータルサイトにアクセスし、課題を閲覧する。ここでは、誰がいつどの課題を取り組み始めたか知るため、学生 ID、課題 ID、時刻を収集する。なお、これら 3 種は以下殆どの場

面での履歴と共に収集している。学習者は、次にテキストエディタを用いてコーディングを行う。エディタは終了時に起動時間、及び入力文字数を収集する。続いて学習者は、コマンドによりコンパイラを起動し作成したソースコードをコンパイルする。この際、ソースコード、エラーメッセージを収集する。またソースコードから構文解析木を生成するシステム<sup>(1)</sup>を用い、サーバは学習履歴受信時に構文解析木を生成し蓄積する。エラーがなければ学習者は生成されたプログラムを実行し、結果を確認する。ここでは発生した入出力や異常時のシグナルを収集する。何かしらのエラーがあれば、学習者はソースコードの修正とコンパイル・実行を繰り返すが、この際にも上記の学習履歴を収集している。こうしてソースコードを完成させた学習者は、ポータルサイトにて成果物であるソースコードと共に課題に関する幾つかのアンケートの回答を提出する。

本稿では、学習履歴を性質に応じ「自動的に収集出来るもの」と「自動的に収集出来ないもの」に分類した。両者の特徴と収集方法を以下に述べる。

### 3.2 自動的に収集出来る学習履歴

自動的に収集出来る学習履歴とは、コマンド操作等システム的な行動により収集出来るものを指す。学習者は学習中に様々なコマンドの起動や、Webシステムの操作を行う。この際、学習者に意識させることなく学習履歴の自動収集を図った。

コーディング・コンパイル・実行時の学習履歴は、対応するコマンドのラッパーを用いて収集する。ラッパーは、正規プログラムの起動、学習履歴の取得・蓄積及びサーバへの送信を行う。サーバは受信した学習履歴をDBに蓄積する。ラッパーの処理は隠蔽されているため、学習者がラッパーの存在を知ることなく、操作も通常と同様である。これにより、学習者に特別な意識や操作をさせない学習履歴収集を実現した。ネットワークの接続状況にも配慮し、未接続時は仮想マシン内に履歴を蓄積し、接続が検知された時にサーバへ転送される。

### 3.3 自動的に収集出来ない学習履歴

自動的に収集出来ない学習履歴とは、収集に際し学習者による何かしらの入力が必要なものを指す。例えば課題に対しての自己評価などが該当するが、これは課題提出時に選択/自由記述式アンケートを学習者に課すことで取得している。一方で、課題解決・提出までの過程における自己評価も該当するが、こちらへの配慮は十分でなく、収集されていなかった。そこで、学習に対する自己評価を行えるインタフェースを用意した。動作画面を図2に示す。画面にはある時点でのコンパイル時のソースコードと、関連する学習履歴を提示する。それまでに発生していたエラーの原因やそれに対するアプローチ、その結果や結果から得た知見、といった学習中の試行錯誤を視覚化でき、これに対する自己評価の記入が出来る。記入された自己評価は他の履歴と同様にソー

スコードに紐付けられる。本機能は、開発環境とは別のインタフェース上で動作するため、学習を阻害せずに課題解決過程における自己評価を収集することが可能になる。

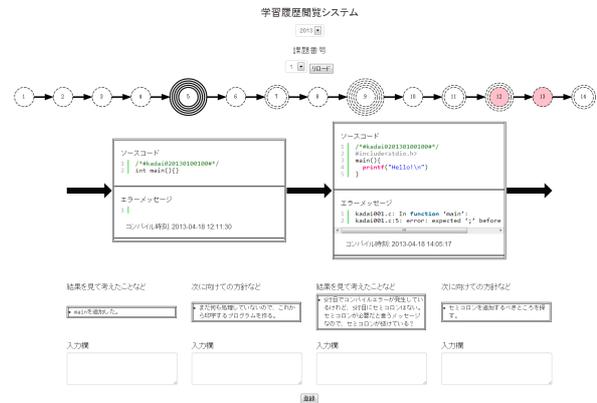


図2 自己評価入力インタフェース動作例

## 4. 学習履歴の活用

本学では学習履歴を支援に活用するための様々な研究を行い、支援システムとして実装してきた。支援システムには、教授者への学習状況把握支援を行うもの<sup>(2)(3)</sup>や学習者への学習支援を行うもの<sup>(3)(4)(5)(6)</sup>、蓄積された学習履歴の分析支援を行うもの<sup>(1)(7)</sup>等がある。学習基盤環境により学習履歴を収集、一元的に管理することで、これらのシステムによる学習履歴を共有・活用した効果的な支援が可能になった。

## 5. おわりに

本稿では、学習履歴の活用によるプログラミング学習支援を行うため、基盤環境を開発した。基盤環境であらゆる種類の学習履歴を収集・管理することで、多種の支援システムによる学習履歴を活用した支援が可能になる。

今後の課題として、収集した学習履歴の分析による支援の足がかりとなる知見の獲得や、最近になって収集を始めた実行結果や学習者の考えを用いた支援システムの開発が挙げられる。

### 参考文献

- (1) 鈴木恵介 他, “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 (2004)
- (2) 倉澤邦美 他, “プログラミング演習における一斉指導のための学習状況把握支援システムの開発”, 電子情報通信学会技術研究報告, Vol.104, No.703, pp.19-24 (2005)
- (3) 野崎要 他, “学習履歴推移グラフによるプログラミング学習状況把握支援システム”, 日本教育工学会第27回全国大会講演論文集, 3a-208-02, pp.787-788 (2011)
- (4) 藤井亮平 他, “アクティブ変数解析によるプログラミングスタイル学習支援システムの開発”, 情報処理学会全国大会講演論文集, Vol.72, No.4, pp.693-694 (2010)
- (5) 山本耕大 他, “エラー要因事例ベースの動的学習手法を導入したC言語教育システムの開発と基礎的評価”, 電子情報通信学会技術研究報告, Vol.108, No.146, pp.67-72 (2008)
- (6) 山本耕大 他, “プログラミング教育における学習者の状況に適応的な支援ツール選出システムの開発”, 電子情報通信学会技術研究報告, Vol.110, No.312, pp.7-12 (2010)
- (7) 野崎要 他, “プログラミング学習履歴分析のための系列パターンマイニング手法の開発”, 電子情報通信学会技術研究報告, Vol.112, No.300, pp.49-54 (2012)