

# 学習者の作成したプログラムを用いた トレース課題自動生成システムの評価

## Evaluation of Automatic Generation System for Tracing Tasks Based on Learner's Programs

茂木 誠拓<sup>\*1</sup>, 立岩 佑一郎<sup>\*2</sup>, 東本 崇仁<sup>\*3</sup>, 赤倉 貴子<sup>\*4</sup>

Tomohiro MOGI<sup>\*1</sup>, Yuichiro TATEIWA<sup>\*2</sup>, Takahito TOMOTO<sup>\*3</sup>, Takako AKAKURA<sup>\*4</sup>

<sup>\*1</sup> 千葉工業大学大学院情報科学研究科

<sup>\*1</sup>Graduate School of Information and Computer Science, Chiba Institute of Technology

<sup>\*2</sup> 名古屋工業大学大学院工学研究科

<sup>\*2</sup>Graduate School of Engineering, Nagoya Institute of Technology

<sup>\*3</sup> 千葉工業大学情報変革科学部

<sup>\*3</sup>Faculty of Innovative Information Science, Chiba institute technology

<sup>\*4</sup> 東京理科大学工学部

<sup>\*4</sup>Faculty of Engineering, Tokyo University of Science

Email: tomo1238hi@gmail.com

**あらまし:** プログラミングにおいてデバッグの能力は重要であるが、デバッグを明示的に学習する機会は多くない。著者らはこれまで、プログラムの実行順や変数遷移などを解答するトレース課題を自動生成するシステムを開発してきた。このシステムを用いてデバッグについての学習が実現できると考えた。本稿では、この学習の実現可能性を確かめるため、学習者の作成したソースコードからトレース課題を自動生成する実験を行った。

**キーワード:** プログラムトレース課題, 自動生成, トレース学習, プログラミング学習, デバッグ

### 1. はじめに

プログラミング学習において、自身の作成したコードがどのような順番で実行されるのか、変数の値はどのように変化するかなどを認識（トレース）し、想定する振る舞いと異なっている場合には修正する（デバッグ）能力を身につけることは重要である。デバッグ能力を身につけるためには、何度も繰り返しデバッグの演習をすることが効果的だと考えられるが、初学者にとってデバッグは容易ではなくその能力を身につけることも難しい。

そこで著者らは、これまで開発してきたソースコードからトレース課題を自動生成するシステムである *larc*<sup>(1)</sup>を用いることで、学習者のデバッグ活動にフィードバックを与え、効率的な学習が実現できると考えた。これまで、教科書のソースコードからトレース課題を生成できることは確認<sup>(2)</sup>されており、*larc*を用いたデバッグ学習が行える可能性が示唆されている。しかしデバッグ学習をする上で、与えられたソースコードではなく、自身の作成したソースコードを用いたデバッグをすることも効果的だと考えられる。しかし、学習者の作成したソースコードは教科書等の洗練されたものとは異なり、不要な処理や変数宣言が含まれる冗長なソースコードである可能性があり、*larc*を用いて自身の作成したソースコードのデバッグを行うためには、冗長なソースコードからもトレース課題を自動生成できる必要がある。

本稿では、*larc*を用いて学習者のソースコードからトレース課題が自動生成できることを確認するた

め、実際に学習者が作成したソースコードからトレース課題を生成する実験を行った。

### 2. プログラムトレース課題と *larc*

本章では、プログラムトレース課題およびその自動生成システムである *larc* について説明する。

プログラムトレース課題とは、学習者にプログラムを提示し「どの行がどの順番で実行されるのか」や「変数の値や出力・条件分岐など」を解答させるものである。*larc* は、ソースコードを入力することで、このトレース課題を自動生成できるシステムである。*larc* における学習の流れについて説明する(図1)。学習者はまず、(I)の「プログラム登録」ボタンを押すことでプログラムを登録する画面に遷移する。次に「タイトル」と「プログラム」にそれぞれ内容を記入し、(II)の「登録」ボタンを押すことによつて

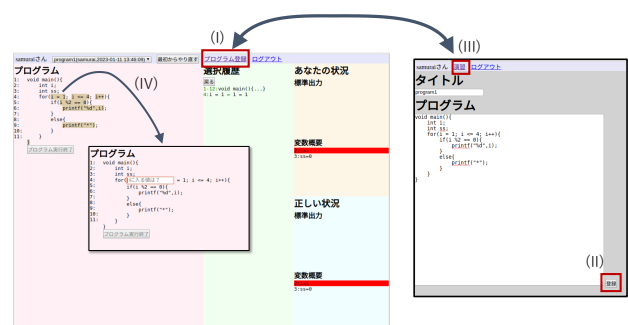


図1 自動生成システムのインターフェース

トレース課題が自動生成される。そして(III)の「演習」ボタンを押すことで、トレース課題に取り組む画面に遷移する。トレース画面ではまず、どの部分が実行されるかを考えクリックする。例えば、図1の例における「i=1」が初めに実行されると考えクリックすると、(IV)のように「iに入る値は？」と聞かれるため、キーボードを用いて解答する。larc では、正しくトレースした場合の標準出力や変数の値を「正しい状況」として表示する機能や、学習者の解答通りにトレースされた場合の標準出力や変数の値を「あなたの状況」として表示する機能を持っている。これらのフィードバックは、答えを教えるもの

表1 自動生成の成否

課題番号	ソースコードの説明	収集	成功	成功率
1	整数の最大値を求める	8	8	100.0%
2	整数の合計値を求める	8	8	100.0%
3	3つの関数を main 関数から呼び出す	8	8	100.0%
4	うるう年を判定する	7	7	100.0%
5	整数の平均値を求める	7	7	100.0%
6	テストの点からそれぞれの偏差値を求める	6	6	100.0%
7	2変数の値を入れ替え	7	7	100.0%
8	文字情報を整形して出力	8	8	100.0%
9	簡単なタイピングチェック	8	8	100.0%
10	簡単な文字列操作	7	7	100.0%
11	乱数を用いたおみくじ	7	7	100.0%
12	標準偏差を計算	6	6	100.0%
13	構造体を用いて複数学生の平均値を計算	7	7	100.0%
14	ユークリッドの互除法	7	7	100.0%
15	簡単なターン制RPG	10	10	100.0%
16	線形探索による検索	8	8	100.0%
17	二分探索による検索	9	9	100.0%
18	入力に当てはまる学生を二分探索で検索し平均点を出力	5	5	100.0%
19	乱数で生成した整数をバブルソートで並べ替え	8	8	100.0%
20	乱数で生成した整数を選択ソートで並べ替え	7	7	100.0%
21	バブルソートの計算時間を求める	14	13	92.9%
22	選択ソートの計算時間を求める	13	12	92.3%
23	乱数で生成した整数を挿入ソートで並べ替え	5	5	100.0%
24	挿入ソートの計算時間を求める	8	8	100.0%
25	乱数を生成し入力した値を二分探索で検索	4	4	100.0%
26	二重 for 文で整数を出力	3	3	100.0%
合計		195	193	99.0%

ではなく、解答がなぜ、どのように間違っているのかを学習者自身に考えさせることを目的としている。

### 3. 評価実験

本章では、larc を用いて学習者のソースコードからトレース課題が生成できることを確認するための実験とその結果について説明する。

実験には、A大学の2年次プログラミング講義で提出されたソースコードを8人の大学生から収集し使用した。講義では26の課題が出題されており、再提出や未提出を含め209個のソースコードが収集された。その内、コンパイルできない14個を除いた195個を実験に使用した。また、本システムはscanf関数やファイル読み込みに対応していないため、適当な数字や文字の代入に変更した。これらの変更を行なったプログラムをシステムに取り込み、トレース課題が自動生成できることを確認した。自動生成の結果を表1に示す。

実験の結果、195個中193個(99.0%)のソースコードで自動生成することができた。自動生成できなかったソースコードを確認してみたところ、失敗したコード全てに「#include <unistd.h>」の行が含まれていることがわかった。詳細な原因については調査中だが、現状のシステムでは<unistd.h>はサポートできていない可能性がある。しかし、その他のソースコードについては問題なく自動生成できたことから、「larc を用いて学習者のソースコードからトレース課題が生成できることを確認する」という目的は概ね達成できたと考えている。

### 4. おわりに

本稿では、larc を用いたデバッグ能力育成(デバッグ学習)の実現に向け、学習者の作成したソースコードからトレース課題を自動生成する実験を行った。その結果、ほとんど全てのソースコードから生成可能であることがわかった。

今後の課題として、自動生成できなかったソースコードの原因調査とその対応や、システムを用いたデバッグ学習の学習効果の確認などが挙げられる。

### 謝辞

本研究の一部は科研費・基盤研究(C)(22K12322)、基盤研究(B)(24K00454)の助成による。

### 参考文献

- (1) Mogi, T., Tateiwa, Y., Tomoto, T., and Akakura T: Proposal for Automatic Problem and Feedback Generation for use in Trace Learning Support Systems, HCI International 2023, vol. 14016 LNCS, pp. 310-321, (2023)
- (2) Mogi, T., Tateiwa, Y., Tomoto, T., and Akakura T: Evaluation of an Automatic Generation System for Tracing Tasks Based on Textbook Programs, Proceedings of the International Conference on Computers in Education ICCE 2023, pp. 327-332, (2023)