

ナビゲーションによるデバッグプロセス体験学習システム*

浅井 俊伍^{†1} 酒井 三四郎^{‡2}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

1 研究の背景・目的

一般的にプログラムのデバッグは、自身の過去の経験に基づいてバグの潜む箇所を予測したり、デバッグプロセスに従ってバグの潜む箇所を探したりして行われる。しかし、プログラミング初学者はデバッグ経験が乏しく、デバッグプロセスも知らない。そのため、デバッグに行き詰り、プログラミングに苦手意識を持ってしまう可能性がある。この原因は、1) デバッグ学習をする機会が少ない、2) 学習する機会を作ろうとしても他に優先すべきことが多く、時間が足りない、の2つであると考えられる。本研究では、デバッグ学習を行うことができるシステムを提案し、デバッグに行き詰ってしまう問題を解決する。原因2)に対処するために、システムを用いた学習はアルゴリズムやデータ構造など他の学習と同時進行で行えることを目指す。

2 学習内容・方法

本システムで学習すべき学習要素は、a) 実行中の変数の値を確認する方法 (print 文の活用)、b) デバッグプロセス、c) デバッガに関する知識、の3つとする。

a) の変数の値の確認は、デバッグの中で最も基本的な作業であり、その方法を明示的に学習することで確実に身に付けられるようにする。

b) のデバッグプロセスは、デバッグの方法を体系的にしたものであり、バグのありかが予測できないような場合に必要になる。プログラミング初学者はデバッグ経験が乏しく、バグのありかの予測が難しいため、デバッグプロセスの学習は重要だと考えられる。デバッグプロセスは、デバッグに関する書籍 [1] と筆者らの経験に基づいて以下のように定義する。

1. 障害の発見
2. 障害の再現性の確認
3. 分割統治法を利用して欠陥が潜むコードの範囲を絞り込む
 - (a) コードをまとまりに分割する
 - (b) まとまり毎に動作を検証する

*Experiential learning system of debugging process by the guidance function

[†]Asai Shungon · Graduate School of Integrated Science and Technology, Shizuoka University

[‡]Sakai Sanshiro · Faculty of Informatics, Shizuoka University

4. 欠陥の潜む範囲をある程度まで絞り込めたら、その範囲を精査する

c) のデバッガは、デバッグ作業を支援するためのツールであり、プログラムの規模が大きくなってくると必要になる。デバッガの存在や役割、使い方を簡単に知り、その利便性を知ることによって、後にデバッガを利用する機会の増加に繋げる。

上記で述べている学習要素を学習する方法として、本研究では実際にデバッグを体験することで学習する体験学習方式を採用する。この方式を採用する理由は、参考書などのようにシステムが知識を教示する方式では受動的な学習になってしまうのに対し、体験学習は能動的な学習にすることができ、学習効果を高めることができるからである。

3 関連研究

プログラミング初学者に対しデバッグ学習支援を目的とした研究はこれまでも行われている。

江木らは、デバッグに行き詰った際に、その状態を打開する方法としてトレースが有用であると述べ、トレース指導を行うデバッグ支援システム DESUS を提案している [2]。この研究では、トレースについて学習することができるが、トレースはデバッグプロセスの一部 (上記の定義で4.に相当) であり、デバッグプロセス全体を学習することができないという問題がある。

山本らは、体系的デバッグ手順 (本研究でのデバッグプロセスと同義) を学習するためのシステムを提案している [3]。この研究では、体系的デバッグ手順を学習することができるが、プログラムの状態を確認する方法がシステム頼りになってしまう問題がある。また、システムを利用するためには正解プログラムが必要になり、学習者が作成するプログラムに合わせて用意する必要がある。

4 システムの設計

本システムへの要求は、print 文を用いたデバッグが体験できることと、デバッガの利用体験ができること、の2つである。

これらの要求を満たすために、次に何をすればいいのかを指示するナビゲーションを実装し、ナビゲーション

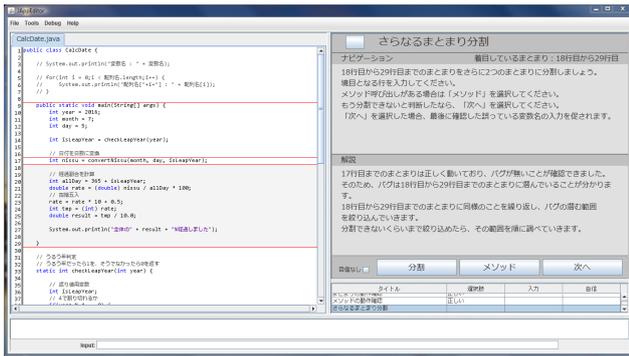


図 1: 提案システムのインターフェース

ンの指示に従うことでデバッグプロセスに従ったデバッグを体験できるようにする。ナビゲーションには指示だけではなく、その指示の理由も表示し、デバッグの考え方も身に着けられるようにする。また、原因2)に対処するために、ナビゲーションには汎用性を持たせ、任意のプログラムをデバッグできるようにし、講義の演習などで作成したプログラムのデバッグに利用できるようにする。

本システムのインターフェースを図1に示す。本システムは、ナビゲーション機能を持った統合開発環境になっており、ナビゲーションの指示に従って print 文の挿入・出力の確認などがその場で行える。エディタ部分は、ナビゲーションの進捗に応じてバグの潜む範囲の可視化などを行い、デバッグの補助を行う。デバッグ機能にも同様にナビゲーションが表示される。

5 評価実験

5.1 実験計画

本システムを実際の講義に導入する前に、本システム利用による学習効果を検証するための評価実験を行った。評価実験では、以下の仮説の検証を行った。

仮説1 デバッグ体験することはデバッグ学習に有効である

仮説2 デバッグの利用体験をすることはデバッグの利便性を知ることができる

評価実験は、プログラミングの学習経験のある情報系の学部3年生5名に対し、プレテスト、システムを利用した学習、ポストテスト、アンケートを行った。テストでは、予めバグを含んだプログラムをシステムを利用せずデバッグする問題を課した。システムを利用した学習では、ナビゲーションの指示に従い例題を解くことで、デバッグ体験とデバッグの利用体験を行い、2章で述べた内容を学習した。

5.2 仮説の検証

プレテストとポストテストでのデバッグの様子を比較すると、ほとんどの被験者が、プレテストではコードとプログラムの解説を見比べていたのに対し、ポス

トテストでは print 文を活用し、途中の状態の確認を行っていた。しかし、明確にまとまりに分割して検証している様子は見受けられず、2分割程度のまとまり分割しか見られなかった。アンケートでは、学習したことを書く記述欄に print 文の活用やまとまりに分割して検証する方法について書いていた被験者が多かった。また、学習目的を伝え、そのことについて理解できたかを尋ねる質問に対し、全員から肯定的な意見を得ることができた。以上のことから、仮説1は支持されているといえる。

デバッガに関するアンケートでは、被験者全員がデバッガの役割や使い方を知ることができたと回答していた。加えて、「今後デバッガを使いたいと思いましたか」という問いに全員が「使いたい」と回答しており、その理由は「print 文を入力してデバッグをするよりもわかりやすく、簡単だから」といった便利だったからという内容のものが多かった。以上のことから、仮説2は支持されているといえる。

6 まとめ

本研究では、知識や経験の少ないプログラミング初学者がデバッグに行き詰まってしまう問題に対し、ナビゲーションに従いながらデバッグを体験をし、デバッグ学習を行うことができるシステムを提案した。本システムを利用してデバッグ体験をすることでデバッグ学習が行えるかを検証するために評価実験を行い、システムの有効性を示すことができた。

今後の課題は、プログラミング講義の中で継続してシステムを利用してもらい、他の学習と同時にデバッグ学習が行えるかを検証することである。システム改良の面では、評価実験時のシステム利用の様子からナビゲーションの指示の理解に時間がかかったり、誤解したりする場面があったため、より分かりやすいナビゲーションにする必要がある。

参考文献

- [1] Andreas Zeller: "Why Programs Fail, Second Edition: A Guide to Systematic Debugging", Morgan Kaufmann Publisher (2009)
- [2] 江木鶴子, 竹内章: "プログラミング初心者にとレースを指導するデバッグ支援システムの開発と評価", 日本教育工学会論文誌, Vol.32, No.4, pp.369-381 (2009)
- [3] 山本頼弥, 野口靖浩, 小暮悟, 山下浩一, 小西達裕, 伊東幸宏: "ワークシートと同期観察ツールを用いた体系的デバッグ手順学習支援環境の構築", 電子情報通信学会技術研究報告, Vol.115, No.74, ET2015-12, pp.7-12 (2015)