

初学者に体系的デバッグ手法を学ばせるための学習支援環境

An Educational Environment to Assist Novice Programmers Learning Systematic Debugging Process

山本 頼弥^{*1}, 野口 靖浩^{*2}, 小暮 悟^{*1}, 山下 浩一^{*3}, 小西 達裕^{*1}, 伊東 幸宏^{*4}
Raiya YAMAMOTO^{*1}, Yasuhiro NOGUCHI^{*2}, Satoru KOGURE^{*1},
Koichi YAMASHITA^{*3}, Tatsuhiro KONISHI^{*1}, Yukihiro ITOH^{*4}
^{*1}静岡大学大学院情報学研究科, ^{*2}静岡大学情報学部

^{*1}Graduate School of Informatics, Shizuoka University, ^{*2}Faculty of Informatics, Shizuoka University
^{*3}浜松大学ビジネスデザイン学部, ^{*4}静岡大学
^{*3}Faculty of Business Design, Hamamatsu University, ^{*4}Shizuoka University
Email: gs13047@s.inf.shizuoka.ac.jp

あらまし：コーディング演習において、時間をかけてもデバッグが完了しない学習者が一定数存在する。そのような学習者に対し、デバッグ活動支援を行うツールが数多く開発されているが、それらのツールは初学者が抱える典型的問題点(1)ツールから得られる情報から自分のプログラムの正誤を判断できない(2)正しいデバッグの手順を身につけていない、を必ずしも解消できない。本稿では、対象世界上での同期観察環境・正しいデバッグ手順を学ぶためのワークシートに基づくデバッグ学習支援環境を提案する。

キーワード：プログラミング教育支援, デバッグ学習支援, 可視化

1. 研究の背景・目的

コーディング演習では、時間を費やしても演習課題を終わらせることができない学習者が一定数存在する。我々は、(1)デバッグ支援ツールから得られる情報を基に自分のプログラムの挙動の正誤を判断できない(2)正しいデバッグ手順を身につけていないために場当たりのデバッグを行ってしまう、という2つの問題点がこの原因であると考えている。

問題点(1)について、プログラムの挙動の可視化により正誤判定を支援するツール(AZUR⁽¹⁾)などが数多く開発されている。これらは一定の支援効果を持つと考えられるが、以下の2つの理由から必ずしも十分な支援にならない。(1-1) アルゴリズムやプログラムの仕様は、データ構造が持つ役割や空間的特徴、相対的位置関係などといった概念を含む形で与えられることが多い。我々はこのような概念を表現する場を対象世界と呼ぶ。しかしながら、多くの可視化ツールでは基本的に挙動の再現の場がデータ構造(変数、配列など)を直接視覚化したものである。そのため、座学で学んだアルゴリズムの挙動とツールで再現された挙動を比較できない(1-2)正しいプログラムの挙動を想起できないため、可視化された自分のプログラムの挙動が正しいか判断できない。

以上より本研究では、学習者と正解プログラムの挙動を対象世界上で((1-1)に対応) 並べて可視化すること((1-2)に対応) によって、両者を比較しつつ観察できるツールを提供することで問題点(1)を解消する。以下、このツールを同期観察ツールと呼ぶ。

問題点(2)について、デバッグ学習を支援するシステムに DESUS⁽²⁾があるが、DESUS の支援対象は関数の中の制御に着目したトレースが中心であり、デバッグ手順全体の学習支援には主眼をおいていない。

そこで我々は、一定の体系に基づいたデバッグ方法(体系的デバッグ)を学ばせるために、デバッグ手順を示すワークシート(以下 WS)を学習者に提示し、WS 上での作業を繰り返すことにより自然に適切なデバッグ手順を身につけられる学習環境を提案する。WS 中には上述の同期観察ツールが組み込まれる。これらによって問題点(2)を解消する。

本稿では以上の議論をもとに同期観察ツール・デバッグ方法学習 WS を含む学習支援環境を提案する。

2. 体系的デバッグ手順と初学者の問題点

本システムで採用したデバッグ戦略では、推定によるデバッグ⁽³⁾(ステップ 1, 2)と逆戻りデバッグ⁽³⁾(ステップ 3)を組み合わせることでバグの原因箇所の絞り込みを行う。表 1 に各ステップにおける初学者が抱える問題点と本システムの解決策を示す。

表 1 各ステップにおける問題点

([]:問題点(1), { }:問題点(2), に対応。():その他)

ステップ	初学者の問題点	解決策
ステップ 1 (プログラム内の呼び出し関係の整理)	{A}呼び出し関係の整理手順を想起できない	{A}WS により想起を支援
ステップ 2 (バグのある関数の絞り込み)	{B}絞り込み手順を想起できない	{B}WS により想起を支援
ステップ 3 (関数内におけるバグ範囲の絞り込み)	[C] 着目関数中のプログラムの挙動がわからない [D] 自分のプログラムの挙動と正しい挙動の差がわからない	[C] 対象世界上で挙動を再現するツール [D] 同期観察ツール
ステップ 4 (設計や実装の修正)	(E) 誤った挙動を正すための修正箇所・内容がわからない。	(E) 後述

ステップ1ではプログラム内の関数呼び出し構造を整理する。ステップ2では関数呼び出し構造を元にバグ誘発入力データの流れを分析し、流れに関与していない関数(=今回のバグの原因となりにえない関数)を除外する。更にデータの流れ上の関数をテストし、そのテスト結果に基づいて除外し、バグ原因を内包する関数を絞り込む。ステップ3では、バグ原因を内包する関数コードから、正しくない結果が生じた点を発見し、そこから逆戻りして論理が誤っている点を発見する。

ステップ4の問題(E)について、今回提案するシステムでは直接的支援は難しい。しかし、ステップ3までの支援でバグ範囲の絞り込みは行われており、修正内容についても挙動の不適切な点に着目させているので、少なくとも場当たりの修正を繰り返すことは防げると考えている。

3. 支援方法

3.1 ワークシート

図1, 2の各WSの①, ②, ...は作業手順を示す。(ア) プログラム呼び出し関係整理のワークシート

図1はステップ1の問題{A}を支援するWSである。学習させたい手順は(i)プログラムに含まれるモジュールを全て想起する,(ii)モジュール間の呼び出し関係を整理する,である。この手順(i)を実行させるためにボタン①があり、押下すると領域<1>に各モジュールを表す楕円オブジェクトが置かれる。これによってモジュールをリストアップするという作業を学習者に想起させる。このオブジェクトを使って、領域②上でモジュールの呼び出し関係を作図させる(手順(ii))。これにより学習者に呼び出し関係のまとめ作業を想起させる。その際、必要に応じて領域<3>のソースコードを確認できる。最後にボタン③で正誤判定を受ける。

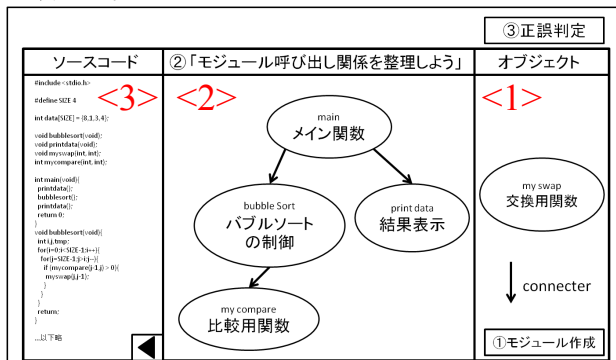


図1 プログラム呼び出し関係整理ワークシート

(イ) バグ関数絞り込みワークシート

図2はステップ2の問題{B}を支援するWSである。学習させたい手順は(i)テストする関数を選ぶ,(ii)テスト入力データを定める(iii)それに対して期待される出力を想起する(vi)関数を実行し実出力と期待出力を比較する,である。まず、①「テストする関数を選ぼう」に注目した学習者に手順(i)を行わせる。次に、②の入力データ領域にテストデータをセットさせる(手順(ii))。そして、③に期待出力をセットさ

せる(手順(iii))。最後に、④に従って、関数の実行と結果の比較を行わせる(手順(iv))。

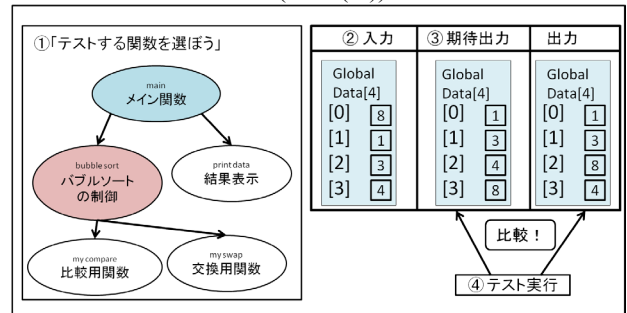


図2 バグ関数絞り込みワークシート

3.2 対象世界上での同期観察

同期観察ツールは対象世界上でプログラムの振舞いを再現する。左側が学習者のプログラム、右側が正解プログラムについての領域である。このシステムでは、<1>が学習者のプログラムの挙動を,<2>が正解プログラムの挙動を再現する対象世界となっており、各プログラムの振舞いを対象世界上で観察することができる(ステップ3の[C]の解決)。<3>に置かれた prev,next ボタンを押すことにより、両プログラムを同期させながら実行状態を進めたり前に戻ることができる。これにより、学習者は自分のプログラムと正解プログラムの挙動の違いを観察し、比較することができる(ステップ3の[D]の解決)。

同期観察ツールは現時点で実装済みであり、小規模ながら評価実験を行って、学習者支援効果を示唆する結果を得ている。

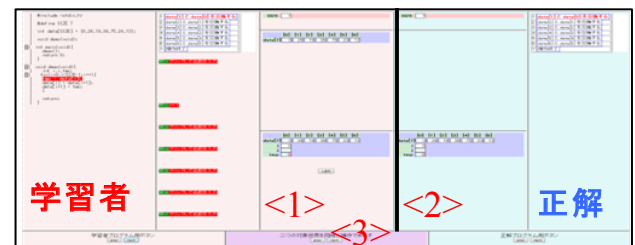


図3 同期観察システムの外観

4. まとめ

本研究では、コーディング演習におけるデバッグ学習支援について考察し、体系的デバッグ手順とその手順における初学者の問題点をまとめた。更に、学習支援システムの枠組みを提案し、その一部を実装した。今後は、未実装の機能を実装し、システム全体としての学習支援効果の評価を行う予定である。

参考文献

- 泉俊幸,橋浦弘明,松浦佐江子,古宮誠一:“プログラミング学習支援環境 AZUR : ブロック構造と関数の可視化”,電子情報通信学会技術研究報告. KBSE, 知能ソフトウェア工学,110(386), pp.61-66(2011)
- 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, 32(4),pp.369-381(2009)
- G.J.マイヤーズ(著), 長尾真(監訳), 松尾正信(訳):“ソフトウェア・テストの技法”, 近代科学社, 東京 (1979)