

段階的詳細化によるオブジェクト指向開発を誘導する

Java 演習支援システムの実装

石井怜央^{†1} 辻健人^{†2} 富永浩之^{†3}香川大学^{†1} 香川大学^{†2} 香川大学^{†3}

1. はじめに

大学情報系学科では、C++/Javaなどの言語を用いたオブジェクト指向プログラミングの入門授業が開講される。そこでは、クラス継承、インスタンス生成、メソッド定義などの、基本的な文法を習得する。しかし、これらの授業で扱う課題は、単なる文法の確認課題となりがちで、オブジェクト指向らしい表現まで扱うことは難しい。その結果、手続き型言語のような処理の記述に留まってしまいう学生も多い。

本研究では、それに続く、オブジェクト指向による設計と実装を実践的に体験する演習を提案している。整列算法を題材とした問題を構築し、演習支援システム TooDex を開発している[1][2]。継続的にコードを改良することで、各オブジェクトの責務を意識させる演習を目指す。関連研究として、学生の解答を、テストを用いて自動的に評価するシステム ArTEMiS がある[3]。本研究では、より初級的な学生を対象とした教育支援を目指す。

本論では、支援システムの本実装を行う。いくつかの課題に解答してもらい、システムの UI や使用感、改善点などについて、アンケートを行う。

2. OOP 開発を意識した Java 演習の概要

本演習では、模範コードの一部を隠蔽した穴埋め形式の設問として出題し、チームで分担して解答する。これらの穴埋め箇所はメソッドやクラスなどの単位とする。出題側で、クラスの関係やメソッドの仕様を定め、オブジェクト指向らしい設計やコーディングを体験させる。また、段階的な設問により、継続的インテグレーションやリファクタリングを意識させる。

学生の提出は、システム側で自動的にテストを実行し評価する。本演習で用いるテストには実行性能テストと品質検証テストの2種類がある。

また、それぞれに、ある1つの問題の穴埋め箇所のみを模範コードに差し替えて実行する個別テストと、関連する問題全体の穴埋め箇所を差し替えてテストする集結テストがある。これらは、それぞれ、実際の現場における単体テスト、結合テストに近いものである。

実行性能テストは、ブラックボックスへの機能要件としての外部評価である。入出力サンプルと実行結果の照合や処理速度などを検証する。

品質検証テストは、非機能要件であるソースコードの品質を注目した内部評価である。ソフトウェアメトリクスの計測や試練コードで、凝集度や結合度、コードの安全性や拡張性などを検証する。

3. 課題の構成と演習の流れ

本演習の出題は、設問、大問、小問の3つのモデルから構成される。設問は、学生が演習に取り組んだ際の最終目標である。開発現場におけるプロジェクトと対応し、例えば、「代表的な整列算法の実装」などである。大問は、設問を達成するための、段階的な目標である。整列算法における、データの生成、ソート、出力などである。小問は、大問をさらに詳細化、分割化したものである。整列算法におけるデータの生成という大問では、「標準入力によるデータ列の生成」や「ファイル読み込みによるデータ列の生成」などが該当する。学生は、各大問における、各小問の実装とリファクタリングを繰り返す。全ての実装が完了すれば、次の大問に移る(図1)。

4. 演習支援システム TooDex

演習支援システム TooDex の、出題および解答を行う演習画面を図2に示す。大問ごとにタブを作成し、それぞれの大問および小問ごとに、対応する穴埋め箇所を含むファイルを表示する。

演習のはじめに、穴埋め箇所が隠蔽されたコードをシステムから配信する。学生は、ローカル環境で穴埋め箇所を補完する形で実装を行い、コピーアンドペーストでシステムに提出する。

小問が提出されると、該当の穴埋め箇所を模範コードに差し替えてテストを実行する。大問が提出された場合は、その大問の解答と、紐づ

Implementation of a Support System of Java Programming Exercises for Object-Oriented Development by Stepwise Refinement.

^{†1}Reo ISHII, Kagawa University

^{†2}Kento TSUJI, Kagawa University

^{†3}Hiroyuki TOMINAGA, Kagawa University

けられた小問に対するこれまでの解答を合わせて、模範コードに差し替える。これらのテストは、ビルドツール Gradle を用いて実行する。テストの実行には、ある程度時間がかかる場合も多いため、ジョブスケジューラの Sidekiq を用いて、非同期に実行している。結果は、ソケット通信を用いて通知し、画面を更新している。

5. 試験的な実践

主要な機能を実装した TooDex を用いて実験を行った。B3 から M2 までの 5 人の学生を対象とし、整列算法の課題から、大問を 1 問と小問を 3 問出題した。内容は表 1 の通りである。出題数を絞った関係で、3 章で述べた「データ生成」に関する実装なども、穴埋め箇所を減らし小問とした。また、実行性能テストのみを用いた。所要時間は一人あたり 30 分～1 時間程度である。実際の演習を想定し、ローカル環境に配信したファイルで実装を行い、支援システム上にコピーアンドペーストで提出してもらった。その後、アンケートを行い、システムの使用感や改善点などについて自由に回答してもらった。

表 1 実験で用いた課題

大問	交換ソートの実装
小問	NumData クラスの実装
小問	NumDataSequence クラスの実装
小問	NumSequenceGenerator クラスの実装

アンケートの結果、まず、システムの動作について、テストに時間がかかることや、テストが実行されていることがわかりづらいという意見があった。問題の場所を歩き来するのが大変という意見もあった。

次に、出題に関する意見として、実装済みのメソッドに関するリファレンスやクラス図がほしいという意見が目立った。

そのほかに「モチベーションが上がる」「必要なモジュールが用意されていて、OOP らしいコードを書くのに集中できそうだった」などの肯定的な意見もあった。

6. おわりに

本研究では、段階的な開発スタイルを誘導する Java 演習を提案し、支援システム TooDex の開発を行っている。TooDex は、部分的に完成したコードの提出に対し、模範コードの該当部分に差し替え、各種のテストを実行する。

段階的にテストケースを満足するように、継続的にコードを改良することで、各オブジェクトの責務を意識させる演習を目指している。

本論では、TooDex の主要な機能について実装を行い、実際の授業での運用に向けた試験的な実験を行なった。アンケートの結果、表示や動作速度について否定的な意見が挙げられた。今回の実験で用いた支援システムの動作環境が、本番の環境に比べ、少し非力であったことも原因であると考えている。実際の環境での動作も確認し、状況に応じてデータベースのクエリやロード画面などを見直す。また、出題に関する意見でも挙げられた、クラス図やリファレンスの提供についても検討する。また、出題する穴埋め箇所の妥当性などについても議論し、実際の授業での本番運用を行う。

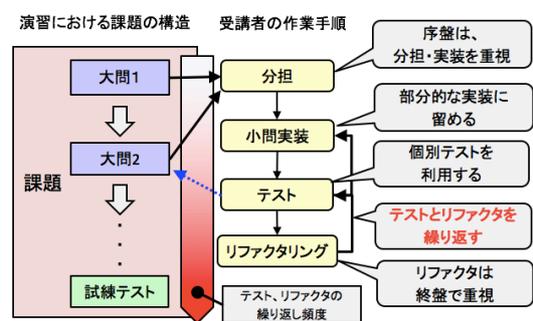


図 1 受講者の演習の手順



図 2 TooDex の演習画面

参考文献

- 1) 辻健人, 富永浩之: オブジェクト指向における段階的な開発スタイルへの誘導する初級 Java 演習の支援システムの試作, 第 80 回全国大会講演論文集, Vol.80, No.1, pp.817-818 (2018).
- 2) 石井怜央, 辻健人, 富永浩之: 段階的詳細化によるオブジェクト指向開発を誘導する Java 演習支援システムの出題採点機能の実装, 第 81 回全国大会講演論文集, Vol.81, No.1, pp.541-542 (2019).
- 3) Krusche, S., Seitz, A.: ArTEMiS - An automatic assessment management system for interactive learning, Proceedings of the 49th ACM Technical Symposium on Computer Science Education, Vol.40, pp.284-289 (2018).