

JUnit に対応したオンラインジャッジシステムの開発

Development of online judge system for JUnit

長尾 和彦, 古谷 勇樹
Kazuhiko NAGAO, Yuki FURUTANI
弓削商船高等専門学校
National Institute of Technology, Yuge College
Email: nagao@info.yuge.ac.jp

あらまし: プログラミングの自己学習支援環境として, LMS や OJS の活用が検討されている. OJS はプログラムの検証に複数のテストを登録する必要がある. 本校では Java プログラミング演習にテストファーストを導入しており, JUnit 形式のプログラムを直接処理できるオンラインジャッジシステムの構築を行い, 問題作成の効率化を実現した. 本報告では, 提案システムの処理効率など, 従来方式との比較検討を行う.

キーワード: オンラインジャッジシステム, テストファースト, JUnit, Java, Moodle

1. はじめに

近年, IT の急速な普及により, IT 技術者の不足が深刻化している⁽¹⁾. IT 技術は社会基盤を支えるものであり, 技術者の不足は国力低下につながるため, 世界各国ではさまざまな取り組みが行われている⁽²⁾. 日本においても, 2012 年から中学校でのプログラミング教育の必修化が行われた⁽³⁾.

プログラミング教育では多くのプログラム作成を行う必要があり, 授業時間のみならず授業時間外での自学自習によるプログラミング体験を増やすことが必要不可欠である. 多くの大学, 高専, 専門学校ではプログラム課題を実施し, 生徒の自学自習を促している.

プログラミング課題による学習には, ①問題の作成, 評価に関する教員の負荷が大きい, ②生徒へのフィードバックが遅い, ③学習履歴の管理ができない, などの問題がある. これらは Web 用 LMS サーバやオンラインジャッジシステム(OJS)の利用によってある程度軽減することが可能である.

本報告では, フリーLMS サーバ moodle⁽⁴⁾で利用できる OJS の比較, Java のテスト環境と親和性の高い OJS の活用について考察する.

2. 本校の学習環境

2.1 プログラミング開発環境とカリキュラム

本校情報工学科では IT 社会に対応できる技術者を育成するために, Java 言語によるプログラミング学習を行っている. Java はオブジェクト指向プログラミング言語であり, 開発範囲が広いことから, エンジニアから高い支持を得ている⁽⁵⁾. 演習は統合開発環境 Eclipse を使い, 実際の開発と同等の環境を用いている.

プログラミング学習は 2～5 年次に配置され, プログラミング基礎 (2 年 4 単位), プログラミング応用 (3 年 3 単位) で重点的に行っている.

・プログラミング基礎

演算子, 制御文, 配列などの基礎的プログラム
アプレットによる GUI プログラム

- ・プログラミング応用
オブジェクト指向プログラム
代表的なアルゴリズムを用いたプログラム
XP 手法⁽⁶⁾を用いたプログラム開発

2.2 extreme programming(XP)

本校では, 効率的なプロジェクト開発手法として知られている XP(extreme programming)のうち, ペアプログラミング, テストファースト, リファクタリングを演習に導入し, 理解度の向上を進めている.

演習では 2 人がペアとなり, moodle に登録された eclipse プロジェクトをダウンロードし, プログラム作成を行う. プロジェクトには JUnit に対応したテストケースが含まれ, 学習者は問題の確認, テストケースの確認と追加, コーディングの作業をペアで進めていく. テストにパスすることで完成となり, 教師や他のグループとのコーディングの比較を行うことで, プログラムの品質を向上させている.

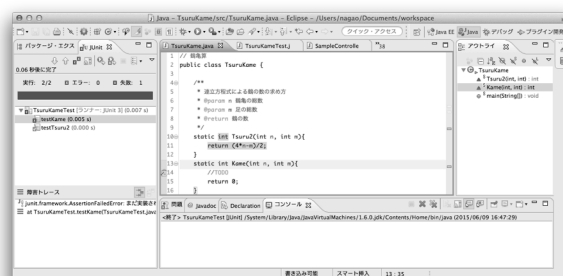


図 1 Eclipse によるテストファースト

2.3 OJS と moodle

OJS はプログラムの自動採点を行うシステムであり, TopCoder⁽⁷⁾や AIZU ONLINE JUDGE⁽⁸⁾などのプログラミングコンテストで利用されている. ユーザは Web ページを介してプログラムの提出を行い, ジャッジサーバでプログラムのコンパイル・実行・採

点を行う。採点は正解と出力結果の比較による。

2.4 LMS サーバ moodle

LMS サーバとは Web ベースで学習管理を行い、生徒・教師の活動を支援する。Moodle は大学・高専で一番多く利用されている LMS サーバである。

Moodle には多くのプラグインが提供されており、ユーザは学習にあったプラグインを組み合わせ、システムをカスタマイズすることができる。

3. OJS と moodle の連携

OJS を用いてプログラム学習を行うためには、学習履歴の管理のため、LMS と連携することが必要である。OJS に対応した moodle プラグインの調査を行った結果、以下の 3 つが確認された。

- Online Judge Plugin for Moodle⁽⁹⁾(OJPM)
- CodeRunner⁽¹⁰⁾
- Upchecker⁽¹¹⁾

各プラグインの特徴を表 1 に示す。本校では、独自のジャッジサーバを構築でき、エディタ機能の充実した CodeRunner を採用し、50 問程度の課題を作成し、学生の自学自習や小テスト用に利用している。

学習者は、問題を確認し、直接 Web 画面でコーディングを行うか、Eclipse で完成したプログラムを提出することで課題提出としている。CodeRunner の問題は JUnit に対応していないため、教師は 1 つずつ問題を修正して登録する必要がある。

表 1 各 OJS の特徴

	OJPM	upchecker	CodeRunner
解答方法	ファイル提出	ファイル提出	コーディング
問題範囲	クラス	クラス	クラス メソッド
プログラミング言語	Java含む 60 言語	Java	Java, C 他
テストファースト(Junit 互換)	×	○	△
セキュリティ	○	×	○
ジャッジサーバ	Ideone	自作サーバ	Jobe
サーバの用意	×	○	○
実装方法または 利用方法	Ideone ア カウント の作成	フルスクラ ッチ	インストー ラーを利用
実装のコスト	◎	×	○

4. JUnit に対応した OJS の構築

教師は JUnit 対応のプロジェクトを作成するため、OJS 用に自動的に変換できることが望ましい。予備実験として、2 つの方法について検討を行った。

4.1 Eclipse プロジェクトファイルの評価

完成した Eclipse プロジェクトファイルを直接評価できるジャッジサーバを構築した。プロジェクト

ファイルには Ant Build ファイルが含まれ、コンパイル・評価することができる。プロジェクト全体を学生が編集可能であるため、脆弱性が問題となった。

4.2 Eclipse による問題の自動登録

教師は問題作成に Eclipse を利用することが多い。あらかじめ作成した JUnit テストケースから、直接 moodle の問題登録が行えるプラグインの開発を現在進めている。JUnit ソースを解析し、moodle XML 形式に変換、インポートすることで効率的に問題を登録できる。

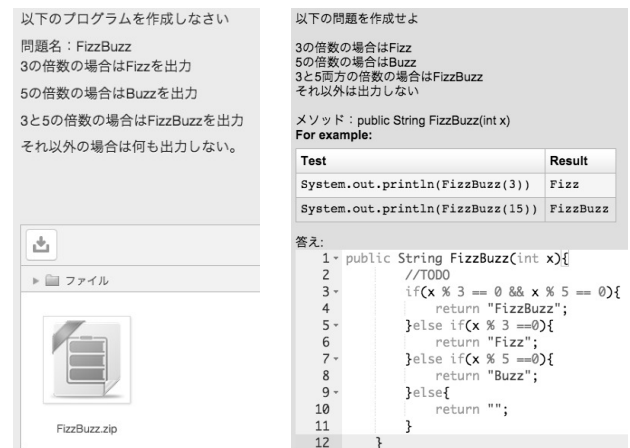


図 2 回答提出画面 (左: upchecker 右: CodeRunner)

5. まとめ

JUnit に対応した OJS の実装を行った。JUnit から問題を作成することで、CodeRunner の長所をいかしつつ、問題作成作業の大幅な改善が実現された。Eclipse プラグインの完成度を高め、授業での活用を進める予定である。

参考文献

- (1) Business Journal: “IT 技術者不足”, http://bi-journal.jp/2014/06/post_5239.html
- (2) Paiza 開発日記: “プログラミング教育を強化した国で何が起きているのか? 世界の教育事情”, <http://paiza.hatenablog.com/entry/2015/02/24/プログラミング教育を強化した国で何が起きてい> (2015)
- (3) 文部科学省高等教育局専門教育課: “文部科学省における情報技術者育成の取組および今後の施策について” (2012)
- (4) moodle : <https://moodle.org/>
- (5) paiza 開発日記: 【Java が恐ろしく強い】転職時に希望するプログラミング言語ランキング, <http://paiza.hatenablog.com/>
- (6) Kent Beck: “XP エクストリーム・プログラミング入門—ソフトウェア開発の究極の手法” (2000).
- (7) Top Coder : <http://www.topcoder.com/>
- (8) AIZU ONLINE JUDGE: <http://judge.u-aizu.ac.jp/>
- (9) Online Judge Plugin for Moodle : <https://github.com/hit-moodle/onlinejudge/wiki>
- (10) CodeRunner: <https://github.com/trampgeek/CodeRunner>
- (11) 伊藤恵, 美馬義亮, 大西昭夫: “コース管理システムと授業固有の課題チェック機能の web サービスによる連携”, 情報処理学会論文誌, Vol.52, No.12 (2011).