

チームプログラミングを可能とした 教育支援アプリケーションの開発

本間 皇成[†], 松本 翔太^{††}, 松本 章代^{†††}, 松原 俊一[†], Martin J. Dürst[†]

[†] 青山学院大学工学部情報テクノロジー学科

^{††} 青山学院大学大学院理工学研究科理工学専攻 ^{†††} 東北学院大学教養学部情報科学科

1 はじめに

情報技術の発展により、技術者の育成が急速に進んでいる。特にプログラミングはその中核にあり、重要性が高まっている。そこで教員はより効果的なプログラミング教育を授業に取り入れることが望まれている。

筆者の所属学科では学部二年生の実習で用いるプログラムチェックシステム（以下 PCS）を三年前から開発し、運用している。PCS は C 言語プログラムを自動採点する Web アプリケーションである。教員は導入前に費やしていた採点の時間をプログラミングの指導に割り当てられるようになった。しかし既存の PCS は個々の学生が作成するプログラムをチェックするもので、チームに対応していない。これは単純で利用しやすい反面、チーム開発で必要な知識を習得できないという欠点がある。本研究ではチーム開発の教育方針を掲げ、既存の PCS を拡張する。

2 教育方針

これまでのプログラミング教育において、教員は学生の作成したプログラムを一つずつ実行し、結果をチェックしていた。人手の採点は時間がかかり、その採点結果は曖昧になることが多かった。PCS はプログラムのチェックをコンピュータに任せることにより、迅速な結果の返答と厳密なチェックを可能にした。しかし既存の PCS は個人開発にのみ対応している。PCS にチーム開発を対応すれば、システムの持つ利点を保ちつつチームで開発する教育を学生に提供できる。

Development of an Educational Support Application for Team Programming

Kiminari Homma[†], Shota Matsumoto^{††}, Akiyo Matsumoto^{†††}, Shunichi Matsubara[†] and Martin J. Dürst[†]

[†] Department of Integrated Information Technology, College of Science and Engineering, Aoyama Gakuin University

^{††} Graduate School of Science and Engineering, Aoyama Gakuin University

^{†††} Department of Information Science, Faculty of Liberal Arts, Tohoku Gakuin University

2.1 教育上の課題

既存の PCS は一人の学生が作成した一つのプログラムをチェックする。この場合、学生は個人開発に慣れてしまい、ソースコードの可読性を意識することなくプログラミングを学習する恐れがある。ソースコードの可読性を高める要素として、インデントの統一 [1] や適切な変数名・関数名、コメントが挙げられる。

教員にとってチームで行うプログラミング教育は各学生の評価・採点を難しくする。チームで作ったプログラムをチェックする場合、一人一人の評価方法を定めることが必要となる。

個人開発では学生の学習意欲の課題が挙げられる。梅村ら [2] は、e-Learning の大きな欠点は「PC の前に一人で向かって学習するため、やる気が持続しない」という孤独感からのモチベーション低下であると指摘している。また、元々プログラミングに興味のない学生は問題に取り組み始めず、反対に優秀な学生は時間を持って余してしまう。既存の PCS にも同様の課題が挙げられる。

2.2 解決策

まず学生は基本的なプログラミングの課題を一人でやる。基礎を身につけ次第、徐々に複数人のチームに分かれて課題に取り組む。ここで学生の座席や能力を考慮し、与える課題の目的に応じた柔軟なチームの振分けが必要となる。

学生が提出したソースコードを同じチーム内の学生に読めるようにする。チーム内でお互いに修正すべき点を指摘し合い、意志疎通をとりながら一つの課題に取り組ませる。学生はこの過程でソースコードの可読性を自然と意識できるようになる。また学習の孤独感を無くすことで、PCS を利用する学生のモチベーション向上が見込める。

採点者側が学生を評価しやすいように、チームのメンバーごとに提出すべきプログラムを割り当てる。学生が一つのプログラムを提出するたびに PCS でチェックができれば、学生ごとの採点ができるようになる。

3 システム設計と実装

本研究の拡張では、PCS に Team と Subassignment の概念を導入した。Subassignment は課題の中の提出すべき一つのファイルを表す。これらの概念を PCS に取り入れ、チーム開発の環境と複数ファイルの管理を実現している。

3.1 課題・チームの作成

課題を作成すると同時にチームの作成を行えるように実装した。管理者の理解しやすい作成画面にするために、課題とチームの作成ページは一つにまとめて表示している。

はじめに管理者はチームメイトの表示の可否を決める。もしここでチームメイトを表示させない場合、チームメイトの提出したソースコードの閲覧を禁止する。この設定の場合、複数ファイルから成るプログラムを独力で書く能力を習得させられる。学生をメインプログラム班とライブラリ班に分け、それぞれが与えられた仕様を読みながらプログラミングを学習することが可能になる。

次に学生の所属するチームと Subassignment を決める。新しくチームを作成する場合、管理者はチームの人数と振分け方法を指定する。または以前のチーム振分け方法の中から一つ選択する。指定したチームの人数に応じて、各チームのメンバーが提出すべき Subassignment を定める。各メンバーの Subassignment 名は複数及び重複を許す。新規でチームを振り分ける画面を図 1 に記す。図 1 ではチーム振分け方法の一つである checked_attempt を指定している。これはそれまでの課題で達成数の多い順に学生を並び替え、チームに割り当てる方法である。

課題の指定が終わると、課題とチームの作成、チームの振分けを PCS 側で行う。チームのメンバーに不足が出た場合はそのチームに教員を一人ずつ割り当てる。

3.2 チーム開発の環境

学生はチームメイトが今まで提出したファイルのソースコードや提出時間、チェック結果を知ることができる。提出時刻の新しい順にファイルを並び替えているため、チームメイトの修正箇所を確認しながらプログラミングできる環境になる。

3.3 複数ファイルの受付

複数に分割された課題の場合、学生は課題提出ページで提出するファイル名を Subassignment の中から選択する。そして提出（アップロード）したファイルは提出期限などをチェックした後、コンパイルされる。課題で要求されたすべてのファイルが整った時点でコン

図 1: 新規チーム作成画面 (一部)

パイル及び実行のチェックを行う。最後にそのチェック結果を学生に知らせる。

4 まとめ

本システムを大学の授業で運用することで、チーム開発の教育要素を取り込める。また、複数ファイルから成るプログラムのチェック機能の追加は、より効果的なプログラミングの教育を学生に与えられる。

今後の課題として、学生間のコミュニケーションをオンラインで行える機能やバージョン管理システムのようなファイル管理機能、類似したプログラムを検出する機能などの追加が挙げられる。

謝辞

本研究は文部科学省科学研究費補助金（基盤 C，課題番号 21500905）の交付を受けている。

参考文献

- [1] 松本翔太, 松本章代, Dürst Martin J. C 言語用のプログラミングスタイル評価システムの構築. 情報処理学会全国大会講演論文集, Vol. 72, No. 4, pp. 685–686, 2010.
- [2] 梅村透, 赤倉貴子. e-learning における仮想的集団学習実現のための機能. 信学技報, Vol. 106, No. 288, pp. 7–12, 2006.