

経験の積み重ねを重視したコーディングスキル向上支援システムの開発 Development of a Coding Skill Improvement Support System that Emphasizes Accumulation of Experience

佐野 智哉^{*1}, 佐々木 整^{*1}
Tomoya Sano^{*1}, Hitoshi Sasaki^{*1}

^{*1} 拓殖大学工学部

^{*1}Faculty of Engineering, Takushoku University
Email:santomo716@eitl.cs.takushoku-u.ac.jp

あらまし：日本の IT 関連企業の成長に伴い、IT 人材育成の必要性が高まっている。最近では、高等学校でのプログラミング教育が必修になるなど、プログラミングに触れる機会も増加している。しかし、プログラミング学習を行った後に、学んだことを活かして実際にコーディングを行い、プログラミングの経験を積み重ねることができない学習者が多く存在する。その原因は、プログラミング学習を終えたばかりの学習者が、何も無い状態からコーディングを行う際に、何から始めればよいかわからないことにあると考えられる。このような学習者に対して、自発的なコーディング機会を与えると同時に、継続的にコーディングを行い、経験を積み重ねられるような環境を提供し、コーディングスキルを向上できるシステムの開発を行う。

キーワード：プログラミング教育、模写コーディング、経験の積み重ね、コーディングスキル向上

1. はじめに

現在、日本の IT 関連企業の成長に伴い、IT 人材の不足が深刻化している。2030 年には最大で約 79 万人もの IT 人材が不足するという試算も出ている⁽¹⁾。そのような中で、IT 人材育成の必要性が高まっている。令和 4 年度から、高等学校でもプログラミング教育が必修となり、情報系の学生でなくてもプログラミングを学ぶ機会が増加している。しかし、学校などでプログラミング学習を行った後、学んだことを活かして自らコーディングを行う場面が少ないという現状がある。自らコーディングを行い、経験を積み重ねることで、プログラミングのスキルを伸ばすことができるが、基礎学習を終えたばかりの学生などは自らコーディングを行う際に、実践的な経験がないため何から手を付けたらよいかわからず、経験を積むことが難しい場合が多い。

そこで本研究では、学習者のコーディングスキル向上を支援するためコーディング経験を積み重ねられる環境を提供するシステムの開発を行う。

2. プログラミング教育における学習方法

2.1 写経型学習

現在、大学などで行われているプログラミング教育では、「写経型学習」と呼ばれる学習方法が行われている。写経型学習は、教授者がお手本となるコードを学習者に与え、それを学習者が入力し実行することで動作確認を行う学習方法である。この学習方法は主にプログラミング初学者を対象に行われ、与えられたプログラムの命令や文法を学習者自身が一部改変したり、組み合わせたりすることで、プログラミングの基本となる「文法の理解」において学習効果があると岡本ら⁽²⁾の研究で示されている。

このことから、写経型学習は多くのプログラミン

グ教育の現場で行われ、プログラミングにおける基礎学習の環境が整いつつある。

2.2 模写コーディングによる学習支援

先行研究では、2019 年頃に注目されるようになった「模写コーディング」という学習方法に着目し、プログラミングの基礎を学んだ学習者が次の段階に進むため、自発的なコーディングの機会を与えるシステムの開発を行った⁽³⁾。模写コーディングは、既存の Web ページやアプリケーションの画面などを模写対象とし、同じ見た目になるようにコーディングを行う学習方法である。模写コーディングには、「実践的なスキルを身に付けることができる」、「実際の Web ページやアプリケーションを模写対象にできる」等のメリットがある。

しかし、模写コーディングは、Web ページで使用されている画像やテキストの準備など、コーディングとは関係のない作業が多く、コーディングを始めるまでに時間がかかってしまうという問題や、ソースコードを参照しない学習方法であるため、基礎学習を終えたばかりの学習者にとっては難易度が高すぎるといった問題が存在し、個人で取り組むには、ハードルが高い学習方法となっている。

そこで先行研究では、模写コーディングで使用する Web ページのファイルを、ディレクトリ構造を保持したまま取得する機能と、編集を行う CSS ファイルをコーディングがしやすいように加工し、提供する機能を開発し、システムに実装することで、学習者が円滑にコーディング学習をはじめられるような環境を提供した。開発機能のイメージを図 1 に示す。しかし、この支援だけで学習者がスクラッチでコーディングを行えるようになることは難しく、先行研究を発展させ、模写コーディングを継続的にを行い、

コーディングスキルを向上させる様々な経験を積むためのさらなる支援を行う必要があると考えた。

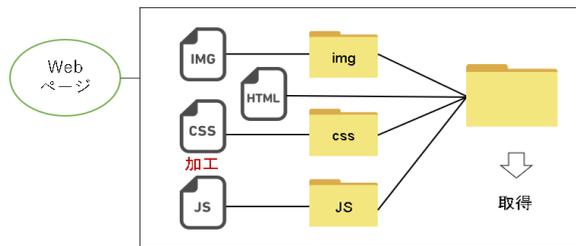


図1 開発機能のイメージ

3. スキル向上を目的とした研究

前章でも述べた通り、模写コーディングはプログラミングの基礎を学んだばかりの学習者にとって難易度が高く、そこから経験を積み重ねるのは非常に大変である。そこで、本研究のシステムで学習者が実践的な学習（本稿では模写コーディング）に移行する際のハードルを下げ、コーディング学習に取り組みやすくすることで、段階的に経験を積み重ねていくことができる環境を提供し、学習者のコーディングスキルを向上させることを目指す。

4. 支援内容の検討

模写コーディングを行い、スキルを向上させるには、試行錯誤を繰り返して、経験を積み重ねていく必要があり、そのためには高いモチベーションの維持が大切である。しかし、コーディング学習は自由度が高く、明確な正解が存在しないため、学習方針に悩む時間が多くなりモチベーションの維持ができずに学習の途中で挫折してしまうことが少なくない。

そこで、本研究で開発するシステムでは、局所的な模写を行うことができる機能や、模写対象のページと作成したページとの違いを視覚的に表示する機能を実装し、学習者のレベルにあった環境で、新たな気づきや達成感を感じさせることによって、モチベーションを高く保ちながらコーディング学習を行うことの支援を検討している。

5. システム実現に向けての取り組み

5.1 局所的な模写

局所的な模写を行うことができる機能では、まず取得してきた Web ページの DOM (Document Object Model) や CSS の解析を行う。次に解析結果を使用して、ページの一部分を模写対象とし、その部分のコードを白紙にすることで、学習者がコーディングを行えるようにする。また、模写対象は学習者が選択できるようにして、学習者の習得状況に応じた場所からコーディングがはじめられるようにする。そうすることで、学習者は、学習が進むにつれてコーディングを行う場所を増やしていくことができ、段階的にコーディングスキルを身に付けていくことができると考えられる。

5.2 模写対象との比較

模写対象のページと作成したページとの違いを視覚的に表示する機能では、ビジュアルリグレッションテストの導入を検討している⁽⁴⁾。ビジュアルリグレッションテストでは、画面に表示されたページや要素が期待通りに表示されているかを修正前後の画像比較で検証することができる。このテストを導入することによって、目視以外での比較が可能となり、学習者自身では気づくことができなかったコードの間違いや見え方の違いを発見することができ、改善点を設定しやすくなると考えられる。また、比較を行うことによって模写対象と見た目が一致しているところを確認し、現在の自分のコーディングスキルを把握することや、達成感を得ることによって学習者のモチベーションが維持できると考えている。

6. おわりに

本稿では、学習者がコーディングの経験を積み重ねられることを主眼に置いた、模写によってコーディングスキル向上を支援する学習支援システムの取り組みについて報告した。写経型学習などによってプログラミングの基礎を学んだ学習者の次のステップとして模写コーディングによってコーディングスキルの向上が期待できると考えているが、基礎を学んだばかりの学習者が一からコーディングを行うことは容易ではなく、模写コーディングによる学習を継続することが難しい。本稿では、その対策として局所的な模写と比較の方法について検討している内容について報告を行ったが、これらだけで十分であるとは考えていない。どのような仕組みを学習者に提供すれば、模写コーディングを断念することなく続けて行くことができるのか、学生研究発表会で皆様のご意見を伺いたい。

参考文献

- (1) 経済産業省, “IT 人材需給に関する調査”, https://www.meti.go.jp/policy/it_policy/jinzai/houkokusyo.pdf, (2023年1月25日確認)
- (2) 岡本雅子, 村上 正行, 吉川 直人, 喜多 一: “プログラミングの写経型学習過程を対象としたつまずきの分析とテキスト教材の改善: 作業の自律的遂行と作業介した理解のための支援と工夫”, 京都大学高等教育研究, 第19号, pp47-57, (2013)
- (3) 濱田 惇也, 佐々木 整, 佐野 智哉, “I 模写コーディングによるプログラミング学習支援システムの開発”, 教育システム情報学会研究報告, vol37, no.5, pp64-71, (2023-01)
- (4) RAKUS Developers Blog: “手軽に始めるビジュアルリグレッションテスト【導入記】”, 2021-10-08, <https://tech-blog.rakus.co.jp/entry/20211008/visualregression>, (2023年1月28日確認)