

Scratch 学習環境におけるプログラムの自動採点と フィードバック機能の構築

Development of an Automated Grading and Feedback System in Scratch Learning Environment

畠中 明哉^{*1}, 内藤 翔太^{*2}, 広瀬 啓雄^{*1,2}, 尾崎 剛^{*1,2}
Akiya Hatanaka^{*1}, Shota Naito^{*2}, Hiroo Hirose^{*1,2}, Takeshi Ozaki^{*1,2}

^{*1} 公立諏訪東京理科大学大学院

^{*1}Graduate School of Engineering and Management, Suwa University of Science

^{*2} 公立諏訪東京理科大学工学部

^{*2}Faculty of Engineering, Suwa University of Science

Email: GH23518@ed.sus.ac.jp

あらまし: 小学校プログラミング教育が開始され、様々な実践事例や支援方法について研究が行われてきた。著者らはこれまで小学生が学ぶべきプログラミングの要素についてドリル形式で学ぶことができる楽手システムについて研究してきた。本研究では、児童が作成 Scratch のプログラムを取得し、正誤判定およびフィードバックを行う機能を構築した。本学工学部の学生を対象に運用テストを行ったところ、正しく正誤判定とフィードバックを行うことができた。

キーワード: プログラミング教育, プログラミング的思考, Scratch, 個別最適化された学び

1. はじめに

日本では、IT 人材不足の解消を目的に、2020 年度から小学校でのプログラミング教育が導入された。文部科学省が公開している「小学校プログラミング教育の手引(第三版)⁽¹⁾」では、「プログラミング的思考」の育成や各教科等での学びをより確実なものにすることなどがねらいとして挙げられている。現在、小学校では Scratch などのビジュアル型プログラミング言語を活用した学習指導が展開されている。

初學者向けのビジュアル型プログラミング言語の教材として、ゲームを作りながら学ぶ教材や物語形式で学ぶ教材がある。これらは、プログラミング言語を習得するという点では楽しく学ぶことができるため非常に有効である。しかし、「プログラミング的思考」や「プログラミングの基本制御構造」などの学習理解度を客観的に確認しづらいという問題点がある。また、小学校のプログラミング教育は多様な教科等に取り入れることが可能であり、そのためには、児童がプログラミングに必要な知識・スキルを身につけるための学習活動を行う必要がある。そのため、教員が授業時間を調節してプログラミングを学ぶ時間を確保しているが、十分な時間を確保することができない可能性がある。しかし、その学習活動を各教科等に組み込んでしまうと、プログラミング教育本来の目的が達成されない可能性がある。そこで、児童が短時間で効率的な学習を可能にし、どの程度プログラミング的思考やプログラミングに関する知識・スキルを習得したかを確認するための方法を明らかにする必要がある。

太田ら⁽²⁾は、小学校におけるプログラミング教育での教師や児童の利用を想定しており、児童が自由に作成したプログラムに対して、分析や評価を行う

ための基準を新たに提案し、その基準に基づいてプログラムの評価を行う学習支援システムを開発した。このシステムはプログラムの評価を行うものであり、特定の課題に対して児童が解いた問題の正誤判定を行うシステムではない。

本研究は、小学生がプログラミング学習を通して学んだ成果を明らかにし、効率的な学習を可能とすることを目的に、Web ベースでの Scratch 学習環境「Scratch ドリル」を開発した。さらに、Scratch ドリルで小学生が学ぶべきプログラミング要素を段階的に学習できる問題を作成し、各問題に対して作成したプログラムの正誤判定およびフィードバックシステムを構築する。

2. Scratch ドリルの学習内容

Scratch ドリルに掲載する問題は、「プログラミング的思考」および「プログラムの制御構造」を考慮し、学習者が自身のペースで学習を進めることができるよう、カテゴリを 11 段階に分けて作成した。カテゴリレベル 1 から 10 の問題は、問題の構成でレベル 1~3 に分類した。また、問題ごとにプログラム作成のためのヒントを 3 種類用意した。Scratch ドリルでは、模範解答を問題に対して最も動きの少ないブロック数で作成できるプログラムとした。

3. Scratch ドリルシステム

Scratch ドリルシステムは、サインアップページ、ログインページ、問題選択ページ、解答作成ページなどで構成される。学習者は、ログインページでログインした後、問題選択ページで問題を選択し、解答作成ページでプログラムを作成する。作成したプログラムは、解答作成ページの「答え合わせ」ボタンを押すことで正誤判定およびフィードバックが行われる。

3.1 正誤判定およびフィードバックシステム

学習者が選択した問題に対し作成したプログラムから、正誤判定に必要なブロック情報を取得・整形し、正誤判定およびフィードバックを実行できる採点オブジェクト(図1)を作成した。また、問題に対する模範解答プログラムを採点オブジェクトに格納したものを模範解答オブジェクトとした。



図1 採点オブジェクトへの変化

正誤判定は、採点オブジェクトと模範解答オブジェクトを比較することで行った。判定結果は「正解」「不正解」の2種類で評価した。判定結果が正解のときは、図2のように解答にかかった時間を表示した。判定結果が「不正解」のときは、図3のように作成したプログラムの振り返りを行うことができるよう、「間違えた箇所」のフィードバックを行った。



図2 判定結果：「正解」



図3 判定結果：「不正解」

4. 運用テスト

構築したScratchドリルの動作を確認するために、本学工学部の学生10人を対象に運用テストを行った。

4.1 運用テストの詳細

運用テストでは、Scratchドリルについての名称や機能などを簡単に説明した。その後、カテゴリレベル1から10のレベル1とレベル3の問題を1問ずつ、計20問解いてもらった。レベル3の問題に関しては、実行映像を見てプログラムを作成することができたかの調査も行った。被験者には、判定結果が「正解」になるまで、問題を解いてもらった。

4.2 運用テストの結果と考察

被験者に、カテゴリレベル1から10のレベル3の問題に対し、実行映像を見てプログラムを作成することができたかを調査した結果、約6割の問題は

実行映像のみで解答可能であり、ヒントまで含めると約8割の問題が解答可能であった。残りの約2割の問題は映像で示す情報が不足していたことが原因で解答ができなかった。

正誤判定については、カテゴリレベル1から8の問題は、すべて正しく正誤判定とフィードバックが行うことができた。しかし、カテゴリレベル9と10の問題は、正しく正誤判定とフィードバックを行うことができなかった。カテゴリレベル1から8の問題は、Scratchに用意されているブロックのみを使用して作成することができる問題である。一方、カテゴリレベル9と10に含まれる変数やリストのブロックは、Scratchの利用者が自由に設定して作成することができるブロックであり、ブロックの名前にブロックidが割り振られている。すなわち、ブロックを作成するたびにブロックの名前が変化し、表示文字列によるブロック名の特定が本研究の方法ではできなかった。そのため、変数やリストのブロック情報の取得が困難であり、正誤判定を行うことができなかった。これらのブロックは、Scratchドリルの判定対象外の問題とし、児童同士でプログラムについて自由に意見を交わすことができる発展問題とし、教員のみが閲覧可能にすることで小学校等での活用が可能であると考えられる。

5. まとめ

本研究は、小学生がプログラミング学習を通して学んだ成果を明らかにし、効率的な学習を可能とすることを目的に、WebベースでのScratch学習環境「Scratchドリル」を開発した。さらに、Scratchドリルで小学生が学ぶべきプログラミング要素を段階的に学習できる問題を作成し、各問題に対して作成したプログラムの正誤判定およびフィードバックシステムを構築した。構築したScratchドリルの運用テストを行った結果、約8割の問題が実行映像とヒントで解答が可能であった。また、カテゴリレベル1~8の問題は、正しく正誤判定とフィードバックを行うことができた。

Scratchドリルを小学校で活用してもらうために、小学生にScratchドリルを使って学習を行ってもらい、問題の難易度・学習効果・受容性、プログラミング的思考やプログラミングに関する知識・スキルの習得度合いを確認するための方法として有効であるか検証する必要がある。

謝辞

本研究は、JSPS 科研費 22K02893, 23K02715 の助成を受けたものです。

参考文献

- (1) 文部科学省：“小学校プログラミング教育の手引(第三版)”(2020)
- (2) 太田剛, 加藤浩, 森本容介：“コンピューショナル・シンキング概念に基づくプログラム自動評価機能を持つScratch用学習支援システムの開発”, 教育システム情報学会誌, Vol.35, No.2, pp.204-214 (2018)