

ブロックプログラミング言語による開発履歴を用いたコンピュータシヨナル・シンキングスキル評価手法の検討

布目 咲梨¹ 山川 祥輝¹ 藤原 賢二^{1,a)}

概要: 小・中学生を対象としたプログラミング教育においては、問題解決のための抽象化や自動化を考える、コンピュータシヨナル・シンキング (CT) を生徒らが身につけたかを評価する手法が重要となる。本研究では、ブロックプログラミングの開発履歴が CT スキルの評価に活用可能かを検討する。

1. はじめに

日本においては 2020 年度から小学校におけるプログラミング教育が必修化されており、それにともない、小・中学生、高校生を対象としたプログラミング教育が注目されている。小学校におけるプログラミング教育のねらいの一つは「プログラミング的思考」をプログラミングを体験することで培うことである [1]。そのため、学校教育においては生徒らが単にプログラミング課題をこなしたかだけでなく、プログラミング的思考を身に付けたかを評価することが求められる。

プログラミング的思考は、Wing が提唱しているコンピュータシヨナル・シンキング (以降, CT) [2][3] と類似点が多い。CT の基本は、問題解決のための問題の抽象化と解決方法の自動化であり、プログラミング学習者の CT スキルを評価する方法として、Moreno-León らにより Dr. Scratch[4]^{*1} が公開されている。Dr. Scratch は学習者が作成した Scratch 作品を投稿するだけで作品中のプログラムに含まれるブロックを自動で解析し点数化を行うため、容易に利用することができる。一方で、点数化には最終成果物のみを用いるため、学校教育において生徒らに同一のプログラミング課題を課した場合、最終的なプログラムが似通ったものとなり、評価結果が似通ったものになってしまうと考えられる。そこで、本研究ではプログラムの開発履歴 (時系列データ) を活用することで、生徒らの CT スキルをより詳細に評価可能かどうかを検討する。具体的には、以下の二つのリサーチクエスチョンに取り組むことで時系

列データの有効性を評価する。

RQ1. Dr. Scratch に代表される、最終成果物に基づく評価は初学者の CT スキルを適切に評価できているか？

RQ2. ブロックプログラミングにおいて、時系列データは初学者の CT スキルの評価に有効か？

各 RQ に取り組むにあたり、小・中学生を対象としたプログラミングワークショップを開催し、ブロックプログラミング言語によるプログラムの開発履歴を収集する。収集にあたっては、著者らが開発している micro:bit^{*2} を対象とした履歴収集システムを利用する [5]。ここで、micro:bit は Scratch と同様に小・中学生らを対象に幅広く利用されているブロックプログラミング環境である。

2. ブロックプログラミングの評価手法

本節では本研究の関連研究である、ブロックプログラミングを対象とした、プログラムの評価手法を紹介する。

Dr. Scratch[4] は小・中学校におけるプログラミング教育で広く活用されている Scratch^{*3} のプログラム中に使用されているブロックを解析し、7つの CT 概念の習熟度をそれぞれ 0 点から 3 点で算出することで学習者の作品を評価する。安東らはこの Dr. Scratch に投稿された作品群を分析することで、Scratch 利用者の CT スキル習熟度予測モデルを提案している [6]。

太田らも Dr. Scratch と同様に、Scratch プログラムから CT スキルを自動採点することでプログラミングを得意としない教師が生徒らに適切な指導を行えるように支援するシステムを開発している。彼らの手法では、CT スキルを 8 項目に分類し、それぞれ 4 段階で評価する。

¹ 東京都市大学
Tokyo City University, Ushikubo-Nishi 3-3-1, Tsuzuki-ku,
Yokohama, Kanagawa, 224-8551, Japan

a) kenjif@tcu.ac.jp

^{*1} <http://www.drscratch.org>

^{*2} <https://microbit.org>

^{*3} <https://scratch.mit.edu>

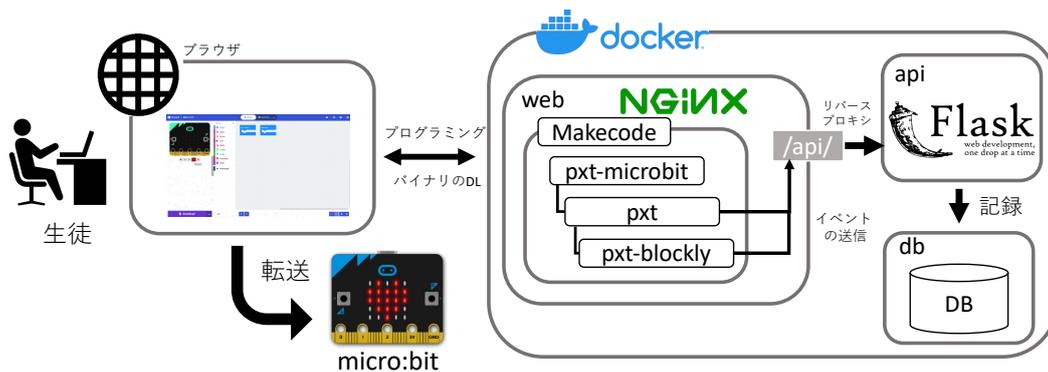


図 1 ブロックプログラミングの開発履歴収集システムの構成

榎原らは Scratch プログラムに対してコードメトリクスを定義し、開発履歴からメトリクスの時系列変化を可視化するシステム CRAVER を開発している [7]。彼女らの研究においては、教員が学習者に指導する際に参照したメトリクスの分析を行っている。

本研究では、先に述べた RQ1 に取り組むにあたり、Dr. Scratch と太田らによる CT スキルの評価手法を参考に、micro:bit の最終成果物を対象とした CT スキルの自動評価手法を検討する。

3. 開発履歴の収集

本研究で分析する micro:bit での開発履歴を収集するシステムの構成を図 1 に示す。このシステムは Microsoft が公開している micro:bit の開発環境である Makecode を改良することで、開発履歴の収集機能を実現している。具体的には、docker コンテナ上で改造済みの Makecode を動作させ、生徒らはブラウザを使って収集機能付きの Makecode にアクセスを行う。この Makecode が提供する Web ページには、ブロックの操作イベントなど、データ収集の起点となるイベントが発生したタイミングで別のコンテナで稼働している API サーバに情報を送信する処理が追加されている。API サーバは受け取ったイベント情報や、生徒が作成しているプログラムの情報を受け取り、データベースに書き込みを行う。

システムによって収集可能な情報は、ブロックの操作に関係するイベントの種類及び発生時刻、ブラウザ上で micro:bit をシミュレートできるシミュレータの操作イベントの種類及び発生時刻などが挙げられる。また、これらのイベントが発生した時点におけるプログラム（ブロック）の情報を XML 形式でデータベースに保存する。

4. 実験計画

現在、著者らの所属する大学が開催する小・中学生向け講座の一環として、micro:bit を用いるプログラミングの体験講座を計画している。この体験講座において、小学生か

ら中学生までの生徒ら約 10 名の開発履歴を収集し、分析を行う予定である。体験講座の内容としては、micro:bit の使用方法および、ブロックプログラミングの基礎を説明した後、生徒らに共通の課題に取り組んでもらうことを考えている。

5. まとめと議論の種

本稿では、コンピューショナル・シンキングスキルを評価するにあたり、ブロックプログラミングの開発履歴が活用可能かどうか検討するためのリサーチクエスチョンおよび、履歴の収集方法について整理した。今後は、実際に生徒らの開発履歴を収集し、最終成果物から求められる CT スキルと、時系列データを用いた CT スキルの分析結果の違いについて考察を行う予定である。ワークショップにおいては、micro:bit における、最終成果物からの CT スキルの評価手法や、時系列データを用いた CT スキルの評価手法の詳細について議論を行いたいと考える。

謝辞 本研究は JSPS 科研費 19K14337 の助成を受けた。

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き（第三版）（2020）。
- [2] Wing, J. M.: Computational Thinking, *Communications of the ACM*, Vol. 49, No. 3, pp. 33–35 (2006).
- [3] 磯辺秀司, 小泉英介, 静谷啓樹, 早川美徳: コンピュータショナル・シンキング, 共立出版 (2016).
- [4] Moreno-León, J., Robles, G. and Román-González, M.: Dr. Scratch: Automatic Analysis of Scratch Projects to Assess and Foster Computational Thinking, *Distance Educational Journal*, No. 46, pp. 1–23 (2015).
- [5] 山川祥輝, 藤原賢二: ブロックプログラミング演習におけるプログラミング行動履歴収集システムの試作, 第 28 回ソフトウェア工学の基礎ワークショップ (FOSE2021) (ポスター発表) (2021).
- [6] 安東亮汰, 伊原彰紀: コンピュータショナル・シンキング・スコアに基づく Scratch ユーザの習熟度到達予測, 情報処理学会論文誌, Vol. 63, No. 4, pp. 928–937 (2022).
- [7] 榎原絵里奈, 米田浩崇, 小野景子: オンライン Scratch プログラミング演習支援にむけたコードメトリクス可視化ツールの提案および評価, 情報処理学会論文誌教育とコンピュータ (TCE), Vol. 8, No. 2, pp. 37–50 (2022).