

Scratch プログラミング学習における コンピューティショナル・シンキングスキルの習熟過程の分析

安東 亮汰^{1,a)} 伊原 彰紀^{1,b)}

概要：Scratch を用いたプログラミング学習では、学習者が自由に作品制作を進める過程でプログラム実装に必要なスキルを学ぶため、学習者の習熟度を客観的に評価することは容易ではない。本論文では、Scratch に蓄積される作品データを対象に、学習者が Scratch を用いて容易に習熟する CT スキル概念を明らかにし、学習者の習熟段階および習熟順序を明確にする方法を検討する。

1. はじめに

プログラミング学習環境 Scratch は、プログラム個々の命令処理をブロックで視覚的に表現することで、学習者の直感的な作品制作を実現している。学習者は作品制作を通して、プログラム実装に必要な思考（コンピューティショナル・シンキング (CT) スキル [1]) を習熟する。学習者は自身が習熟した CT スキルを把握するために、作品評価ツール Dr.Scratch [2] を用いることで、制作した作品に使用された CT スキルの 7 概念をそれぞれ 0 点から 3 点で評価することができる。表 1 は CT スキル 7 概念とそれぞれの評価基準を示す。

数学や英語のような明確に学習順序が示されているカリキュラムを有する学習は公式学習に分類される一方で、Scratch のような学習者が自由に作品制作を進める過程で習熟する学習は非公式学習に分類される。非公式学習は、明確なカリキュラムを持たないが故、学習者が習熟した内容を客観評価することは容易でない。従来研究では学習者が Scratch における習熟過程を明らかにするための研究が進められている。Aivaloglou らは、Scratch 作品 231,050 件を収集し、Dr.Scratch を用いて各作品の評価結果を調査した結果、概念の一つであるフロー制御、ユーザ対話性は容易に高い点数を獲得できる一方で、抽象化、論理は高い点数の獲得が困難であることを明らかにした [3]。また、Troiano らは、学習者が単一作品を制作していく過程で、作品に使用される CT スキルの点数の移り変わりを調査している。その結果、作品制作の時間の経過と共に、並列、論理、同期は点数が向上していく一方で、抽象化は点数の

向上が困難であることを明らかにした [4]。Dr.Scratch を用いることで、ある一時点において学習者が作品制作に要した概念を把握することができるが、学習者が CT スキル概念を習熟する過程は明らかにされていない。

本論文では、Scratch を用いたプログラミング学習における CT スキル概念の習熟段階および習熟順序を明らかにする方法を検討する。具体的には、Scratch に蓄積される作品データを対象に、作品制作に使用されることの多い概念 (対) を調査することで、Scratch で制作された作品に頻繁に使用される概念を明らかにする。その後、CT スキル概念の習熟段階および各習熟段階を達成するために事前に習熟する必要のある概念を明らかにする方法を検討する。

2. Scratch プログラミングにおいて習熟する CT スキル概念の分析

本論文では、学習者が過去に制作した作品に使用した CT スキル概念を明らかにするために、Scratch3.0 がリリースした 2019 年 1 月 3 日以降に Scratch を使用し、20 以上の作品を制作した学習者 6,323 人の作品 126,460 件を分析対象とする。紙面の都合上、分析対象作品の統計量を示すことはできないが、分析対象の作品における CT スキル概念の点数分布は、従来研究の Aivaloglou らが公開する作品の点数分布と類似しているため、20 以上の作品を制作した学習者を対象としたことによる、作品の特徴に偏りはないと考える。

2.1 作品制作で習熟する CT スキル概念の理解

Scratch を用いたプログラミング学習において習熟する CT スキル概念を明らかにするために、本節では Scratch における作品制作において学習者が頻繁に使用する概念 (対) を分析する。学習者が頻繁に使用する単一の概念に

¹ 和歌山大学
Wakayama University, Wakayama 640-8510, Japan

a) s216009@wakayama-u.ac.jp

b) ihara@wakayama-u.ac.jp

表 1 CT スキル 7 概念の評価基準

概念	抽象化 (Abstraction)	並列 (Parallelism)	論理 (Logic)	同期 (Synchronization)	フロー制御 (Flow Control)	ユーザ対話性 (User Interactivity)	データ表現 (Data Representation)
評価基準	機能単位に分割	並列処理の使用	論理制御の使用	待機処理の使用	反復処理の使用	ユーザの入力動作の使用	変数の使用

については、1 章で述べた通り Aivaloglou らが既に調査しているため、本分析では概念対についてのみ分析する。

作品制作に使用される概念対を分析するために、本論文ではアソシエーション分析を用いる。アソシエーション分析は、データ内である事象 {X} が発生するときに同時に発生する事象 {Y} を抽出する手法であり、評価指標として支持度、信頼度、リフトの 3 つを使用する。

- **支持度**：事象 {X} と {Y} がデータ全体で出現する割合
- **信頼度**：事象 {X} が出現する中で、事象 {Y} が同時に出現する割合
- **リフト**：信頼度を事象 {Y} が出現する割合で割った値 (1 以上である場合、事象 {X} と {Y} の関連性は高い)

アソシエーション分析には、R の `arules` パッケージを用いた。作品 126,460 件それぞれの 7 概念の点数を入力として与えることで、同一作品に使用される概念対のルールを抽出した。使用頻度の少ない概念対のルールを除去するために支持度は 0.05、信頼度は 0.05、リフトは 1 を最小値と設定した。また、冗長なルールは支持度は低い一方で、リフトが高くなるという結果になるため、1 つのルールに含まれる概念の数が 4 以上のルールは除去した。図 1 は、抽出したルールのうち、リフト値が高い上位 50 件の結果をネットワーク図で示す。ノードの大きさは支持度を示し、ノードの色が濃いほどリフト値が高い。また、ノードに入出力するエッジは、作品に頻繁に使用される概念名とその評価点数を示す。例えば“Abst.3”は抽象化で 3 点を獲得することを意味する。概念名の距離が近いほどそれらが同一作品に使用される概念であることを示す。

図 1 の左側に各概念の 0 点を多数含むルールのネットワークが示されるため、同一作品で 0 点の概念と 3 点の概念が使用される頻度は少ないことがわかる。また、図 1 の右側に着目すると、論理で 3 点やユーザ対話性で 3 点を獲得するとき、データ表現を除いた他の概念においても 3 点を獲得するが、抽象化やフロー制御で 3 点を獲得する作品は、並列や同期では同時に 3 点を獲得していないことが分かる。したがって、Scratch 作品において使用する CT スキルには、同時に使用される概念対と、同時に使用されることが少ない概念対が存在する。本分析結果に基づき、学習者が繰り返し作品を制作する過程で使用する概念 (対) の変化を分析可能となる。

2.2 CT スキル概念の習熟段階

前節において明らかにした制作に必要な CT スキル概念

ノードの大きさ：支持度 (0.051 - 0.075)
 ノードの色濃さ：リフト (4.161 - 15.285)
 Abst: 抽象化, Para: 並列, Logi: 論理, Sync: 同期
 Flow: フロー制御, User: ユーザ対話性, Data: データ表現



図 1 アソシエーション分析の結果のうち、リフト値が高い上位 50 件のルール

対に基づき、新たに制作する作品までに使用した概念と将来使用する概念を分析することで、CT スキルの習熟順序を明らかにする方法を検討している。

3. まとめ

本論文では、Scratch を用いたプログラミング学習において作品制作に頻繁に使用される CT スキル概念対を明らかにした。今後は、習熟段階を明らかにすることで可能となる CT スキル習熟度を評価する方法を検討する。

謝辞 本研究は、JSPS 科研費 18KT0013 の助成を受けたものです。

参考文献

- [1] Wing, J. M.: Computational thinking and thinking about computing, *Journal of PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY*, Vol. 366, pp. 3717–3725 (2018).
- [2] Moreno-León, J., Robles, G. and Román-González, M.: Dr. Scratch: Automatic analysis of scratch projects to assess and foster computational thinking, *RED. Revista de Educación a Distancia*, No. 46, pp. 1–23 (2015).
- [3] Aivaloglou, E., Hermans, F., Moreno-León, J. and Robles, G.: A Dataset of Scratch Programs: Scraped, Shaped and Scored, *Proceedings of the 14th International Conference on Mining Software Repositories (MSR'17)*, pp. 511–514 (2017).
- [4] Troiano, G., Snodgrass, S., Argimak, E., Robles, G., Smith, G., Cassidy, M., Tucker-Raymond, E., Puttick, G. and Harteveld, C.: Is My Game OK Dr. Scratch?: Exploring Programming and Computational Thinking Development via Metrics in Student-Designed Serious Games for STEM, *Proceedings of the 18th ACM International Conference on Interaction Design and Children (2019)*.