

日常の活動を題材とした小学生向け プログラミング的思考育成ツールの開発

内田早紀子^{†1} 松村敦^{†2}

概要：2020年度から小学校でプログラミング教育が必修化される。暗記型のプログラミング教育では、小学生がプログラミング的思考を習得することは困難とされている。そこで、小学生にとって身近な日常の活動を題材としたプログラミング的思考育成ツールを開発した。小学校2校で利用実験と評価を行ったところ、プログラミング的思考の向上の効果が異なった。これは、プログラミング的思考の評価方法の違いが大きく影響している可能性がある。また、ファシリテーターの教え方に差があったため統制された評価になっていないことも一因と考えられる。アンケートでは、参加者の約8割の子供から楽しく学習でき、ツールは使いやすかったと回答があり、本ツールは小学生が利用するのに適している事がわかったが、約3割が難しいと感じていた。子供たちの理解に合わせた動きや事象と問題のレベルの設定については、再度検討する余地がある。

キーワード：情報教育、小学校、学習、教育支援

Development of programming thinking training tool for elementary school students based on daily activities

SAKIKO UCHIDA^{†1} ATSUSHI MATSUMURA^{†2}

1. はじめに

2020年度からの学習指導要領において、小学校からのプログラミング教育が必修化される。小学校のプログラミング教育とは、子供たちにコンピュータに意図した処理を行うよう指示することができるということを体験させながら、「プログラミング的思考」などを育むことであり、コーディングを覚えることが目的ではないとされている[1]。プログラミング思考とは、「自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力[1]である。そこで小学生向けのワークショップや実際の授業でプログラミングを実践し、講義ベースのアプローチよりも簡単で効果的であるという事例が多くみられるようになった[2]。さらに、小学校の低学年段階においてプログラミングへの興味の喚起とともに、他者と協力して物事を進める力の育成にも効果的であることも分かった[3]。

しかし、尾花らは、従来の暗記型のプログラミング教育では、小学生がプログラミング的思考を習得することは困

難であると指摘し、新たに小・中学生向けのビジュアルプログラミング言語を用いプログラミング学習のアプリケーションを開発、提案している[4]。これによりプログラミング構造の基礎となる「順次処理」、「分岐処理」、「繰り返し処理」について学習することができた。さらに、川本らは、フローチャートを用いた学習システムを開発し、使用した小学生の論理的文章を書く力が向上した事例がある[5]。小学生がプログラミング的思考を学習するためには、アプリケーションを新たに開発する必要があると言える。

また、コンピュータを使わないアンプラグドプログラミング教育を小学校で実践し、成果をあげている事例がある。国語科において、「朝ごはんを食べる」と言った身近な課題を取り込むことにより、簡単に順序を考えることができた[6]。さらに、理科の実験に使った身の回りのものを分けていき、条件分岐の考えをおさえることができたという事例もある[6]。別の先行研究では、給食の配膳や掃除の手順を可視化し、フローチャートを用いる事によって、順次処理、条件分岐、繰り返しを学習し、成果を上げている[7]。

これらを踏まえ、小学生にとって身近な題材をアプリケーションとすることで、より効果のある学習支援ができるのではないかと考えた。そこで本研究では、日常の活動を題材に用い、プログラミング的思考を育成するツールを開

^{†1} 筑波大学大学院 図書館情報メディア研究科
Graduate school of Library, Information and Media Studies, University of Tsukuba

^{†2} 筑波大学 図書館情報メディア系
Faculty of Library, Information and Media Science, University of Tsukuba

発した。このツールを小学生が用いることで、プログラミング的思考力が向上することを目指した。

2. 手法

2.1 ツールの題材

身近な課題を取り組むことで、物事の順番を簡単に考えることができたという先行研究[6][7]をもとに、ツールの題材を日常の活動の題材とした。さらに、「プログラミングで育成する資質・能力の評価規準」[8]では、「身近な生活でコンピュータが活用されていることに気が付くこと」が目標とされている。小学生の身近な生活の活動をプログラミングすることで、コンピュータが活用されていることに気づくことが示唆される。これらの理由より、小学生が日常的に行っている片付け、お手伝いや学校の準備を題材に選定した。今回開発したツールでは、「片付け」、「洋服の選択」、「出かける準備」、「肩揉み」、「料理の手伝い」、「犬の散歩」、「じゃんけん」、「ドッジボール」を題材とした。

2.2 プログラミング的思考の評価

プログラミング的思考の評価は、「プログラミングで育成する資質・能力の評価規準」[8]を参考に表1にまとめた。この基準では、「問題解決のためにコンピュータに指示を出すには必要な手順があることに気付く」とある。低学年は「順序がある場面があることを知ること」、中学年は「条件を満たすまで動作を続ける場面があることを知ること」、高学年は「条件により動作が変化する場面があることを知ること」となっている。そこで、「必要な手順があることに気付く」に着目し、「順次処理」、「繰り返し処理」、「条件分岐処理」を実装するツールの開発を行うこととした。

表 1 プログラミングで育成する資質・能力の評価規準[8]

プログラミング教育を通じて目指す育成すべき資質・能力(文部科学省)	身近な生活でコンピュータが活用されていることや、問題の解決には必要な手順があることに気付くこと。
目標	問題解決のためにコンピュータに指示を出すには必要な手順があることに気付く。
低学年(1・2年生)	1. コンピュータには明確な手順を命令する必要があることを知ること。 2. 順序がある場面があることを知ること。(順次処理)
中学年(3・4年生)	1. 条件を満たすまで動作を続ける場面があることを知ること。(繰り返し処理) 2. 順次処理や繰り返し処理を組み合わせ、コンピュータに意図した処理を行うための
高学年(5・6年生)	1. 条件により動作が変化する場面があることを知ること。(条件分岐処理) 2. 順次処理や繰り返し処理、条件分岐処理を組み合わせ、コンピュータに意図した処理を行うための指示を出す体験をすること。 3. センサーの存在を知り、センサーが身近な生活で活用されていることに気付くこと

2.3 ツール開発

2.3.1 ツール概要

小学生の低学年でも操作しやすいタブレット端末 (iPad) を選定した。iPad 用のアプリケーションを作成するため Xcode を用いて開発を行った。

プログラミング的思考を育成する要素[8]として、「目的に合わせてより良い手順を作ること」と定義されている。そのため、目的に合わせてよりよい手順に事象を並び替える仕様とした。作成したツールの画面例を図1に示す。



図 1 ツールの画面

2.3.2 ステージの内容

ステージの内容は、表2にまとめた。ステージは7つの学習用ステージ、各3つの事前テスト・事後テスト用のステージと3つのチャレンジ問題のステージの全16ステージを作成した。題材の一連の動きを分割し、事象を正しい手順に並び替える内容とした。加えて、「順次処理」、「繰り返し処理」、「条件分岐処理」を実装した。

各ステージを実施後、正解の場合には「クリア！やったね！」、不正解の場合には、「ちょっと違うよ！もう一回やってみて！」と正解か不正解かがわかるようなメッセージを表示した。正解の場合でも不正解の場合でも、何度でも並び替えを行えるような仕様とした。

以下では、各処理の具体的な内容を説明する。

(1) 順次処理

「片付け」を題材に、「鉛筆を筆箱に入れて、ランドセルに入れる」という一連の動きを、「鉛筆を筆箱に入れる」「筆箱のふたをしめる」「ランドセルに筆箱を入れる」に分割した事象を順不同に配置した。実験参加者は、正しい手順を考えて、事象を並び替える事ができる。

表 2 ツールのステージ内容

ステージ	内容	順次	条件分岐	繰り返し
1	片付け	3		
2	洋服の選択		1	
3	出かける準備	2	1	
4	洋服の選択		2	
5	腰揉み			1
6	料理のお手伝い	5		
7	犬の散歩	1	1	1
8	事前テスト 明日の学校の準備	4	2	
9	事前テスト じゃんけん	3	3	
10	事前テスト ドッジボール	2	2	1
11	事後テスト 明日の学校の準備	4	2	
12	事後テスト じゃんけん	3	3	
13	事後テスト ドッジボール	2	2	1
14	チャレンジ問題 明日の学校の準備	4	1	
15	チャレンジ問題 じゃんけん	3	3	
16	チャレンジ問題 ドッジボール	4	2	1

(2) 「繰り返し処理」

「犬の散歩」を題材に、「ポチが喜ぶまでボールで遊んであげる」という一連の動きを、「ポチが喜ぶまで繰り返す」「ボールで遊んであげる」「↑」に分割した。「～まで繰り返す」と「↑」の事象を分かりやすくするために、オレンジ色に配色した。「順次処理」と同様に正しい手順に並び替える事ができる。最後の上矢印は、繰り返しの条件が上矢印まで有効であることを意味する。

(3) 「条件分岐処理」

「出かける準備」を題材に、「明日、国語の授業があったなら、国語の教科書をランドセルに入れる」という一連の動きを、「国語の授業があったなら」「国語の教科書をランドセルに入れる」「←」に分割した。「～なら」と「←」の事象を分かりやすくするために、青色に配色した。「順次処理」と同様に正しい手順に並び替える事が出来る。最後の左矢印は、分岐の条件が左矢印まで有効であることを意味する。

● 事前テスト・事後テスト

「明日の学校の準備」、「じゃんけんの手順」、「ドッジボールの手順」の一連の動きに、「順次処理」、「繰り返し処理」、「条件分岐処理」を含んだ文章の事象を、正しく並び替える課題を出した。文章の並び替えを目的とするため、学習用ステージの「繰り返し処理」と「条件分岐処理」の矢印や配色は用いない。

● チャレンジ問題

事前テスト・事後テストと同様の課題とし、学習用ステージの矢印や配色を用いた。

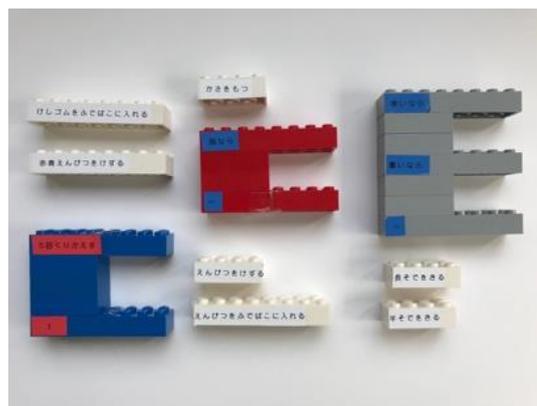


図 2 使用したレゴブロック

表 3 参加者の内訳

	1年生	2年生	3年生	4年生	5年生	6年生	合計
A 小学校	16人	10人	13人	10人	9人	7人	53人
B 小学校	13人	12人	13人	9人	4人	2人	64人

3. 評価実験

3.1 実験参加者

実験参加者は、2校の小学校にて、夏休みのプログラミング教室のイベントとして参加した117名(男子73名、女子44名)である(表3)。保護者向けにメールにて参加者の募集を行い、A小学校(2018年8月7日に実施)は64名(男子45名、女子19名)、B小学校(2018年8月31日に実施)は53名(男子28名、女子25名)が参加した。実験時間は1時間とし、低学年(1, 2年生)、中学年(3, 4年生)、高学年(5, 6年生)の3クラスに分けて行った。

3.2 実験手順

- 導入として、テレビ、ゲームやスマートフォンの例などを示しながら、プログラミングは子供達の身近にあることを説明した。次に、レゴブロックを用い、「雨なら傘をさす」「寒いなら長袖を着る、暑いなら半袖を着る」「鉛筆を削って、筆箱に入れるを5回繰り返す」を例として順次処理、繰り返し処理、条件分岐処理の説明を行った(図2)。
- 事前テストは、A小学校では「ドッジボールのルール」の文章を書く課題、B小学校ではツール上で行う課題を出した。
- プログラミングツールを用い、順次処理、繰り返し処理、条件分岐処理の学習を行った。
- 事後テストは、事前テストと同様の課題を行った。
- 事後テストの他に、チャレンジ問題を行った。
- ツールの楽しさ、使いやすさ、難しさのアンケートを参加者に行った。数日後、学習後の家庭での様子について、アンケートを保護者に行った。

4. 結果と考察

4.1 事前テストと事後テストの結果 (A 小学校)

A 小学校での事前テストと事後テストの結果を表 4 に示す。川本らの小学生を対象とした学習支援システムの評価実験では、「じゃんけんのルール」を説明する文章を参加者に作成させ、整理した文章の論理性を評価した結果、学習後の方が論理的文章を書けるようになったという成果がある[5]。この先行研究では、「じゃんけん」といった本研究の着目した課題である小学生に身近な題材を評価の課題に採用しているため、A 小学校の事前テストは文章を書く課題とした。

作成してもらった文章から、プログラミング的思考の評価項目である「順次処理」、「繰り返し処理」、「条件分岐処理」を使用した回数を算出した。回数の算出方法は、「順次処理」は、まず一連の活動を事象に 2 つ以上分解されていること、その事象の組み合わせが正しい順序かどうかを判断し、正しい場合には組み合わせごとに 1 点を加算する。複数の正しい組み合わせがある場合は、組み合わせごとに 1 点を加算する。「繰り返し処理」は、「～まで～する」という語彙が含まれていたら 1 点を加算し、それが複数回含まれていたら、回数分得点を加算する。「条件分岐処理」は、「(もし)～なら」という語彙が含まれていたら 1 点を加算し、それが複数回含まれていたら、回数分得点を加算する。その結果を表 4 に示す。学年別の平均点は、低学年 2.88 点、中学年 3.35 点、高学年 6.73 点であった。

一方、事後テストでは事前テストと同様の課題、算出方法で行った。その結果、低学年 1.00 点、中学年 2.78 点、高学年 6.47 点であった。事前テストと事後テストを比較した結果、事前テストの点数の方が高かった。

本実験では、川本ら[5]の実験のように、プログラミング的思考向上の結果を示すことはできなかった。先行研究の実験参加者が理数に関する試験を受け、選抜された小学 3～6 年を対象であるのに対して、本実験の参加者は、小学校での必修化を考慮し、一般の小学生を対象としたため、異なる結果になってしまったのではないと思われる。さらに、事前テストと事後テストで同じ課題を出したため、1 度書いた課題をもう一度書く事への飽きや、すでに十分書かれていた回答にさらに追加する手順を思いつく事ができなかったように思われる。これらの結果を踏まえて、B 小学校では文章を書く課題ではなく、ツールを使用した評価に変更し、操作ログが採取できるようにツールを改良した。

表 4 A 小学校の学年別テスト結果(平均)

学年	事前テスト	事後テスト	比較
低学年	2.88	1.00	➡
中学年	3.35	2.78	➡
高学年	6.73	6.47	➡

4.2 事前テストと事後テストの結果 (B 小学校)

B 小学校での事前テストと事後テストの結果を表 5 に示す。事前テストは、文章を書く課題ではなくツール上で「順次処理」、「条件分岐処理」、「繰り返し処理」を用いた課題とした。テストは何度でも実行可能であるが、1 回目に行った回答を対象に、並び替えた結果が正しい場合 10 点、不正解の場合 0 点とした。その結果を表 5 に示す。学年別の平均点数は、低学年 5.60 点、中学年 11.36 点、高学年 15.00 点であった。

事後テストは、事前テストと同様の課題、算出方法で行った。その結果、低学年 8.00 点、中学年 13.18 点、高学年 20.00 点となり、事前テストより高い結果となった。プログラミング的思考の向上に一定の効果があつたことが示唆される。しかし、ワークショップを行った 3 人のファシリテーターが参加者との接し方が異なったため、ツールを使用した効果なのか、ファシリテーターの教え方がよかつたのかが、判断できなかった。今後は、参加者に対する接し方を、ファシリテーター内で統一し、ツールを使用した効果を正確に測定することが重要な課題となる。

表 5 B 小学校の学年別テスト結果(平均)

学年	事前テスト	事後テスト	比較
低学年	5.60	8.00	➡
中学年	11.36	13.18	➡
高学年	15.00	20.00	➡

4.3 全実行回数と正解までの実行回数 (B 小学校)

プログラミング的思考を育成する要素[8]として、「目的に対して、必要十分な評価の観点を考え、実行したことが、意図した活動に近づいているかどうか評価すること」と定義されている。そこで、参加者一人当たりの各ステージにおける全実行回数と正解までの実行回数それぞれの平均を図 3 にまとめた。ステージ番号 1 から 7 は学習用のステージ、11 から 13 は事後テスト、14 から 16 はチャレンジ問題である。なお、ステージ 8 から 10 は事前テストであり、実行回数を 1 回に制限しているため、ここでは対象外とした。

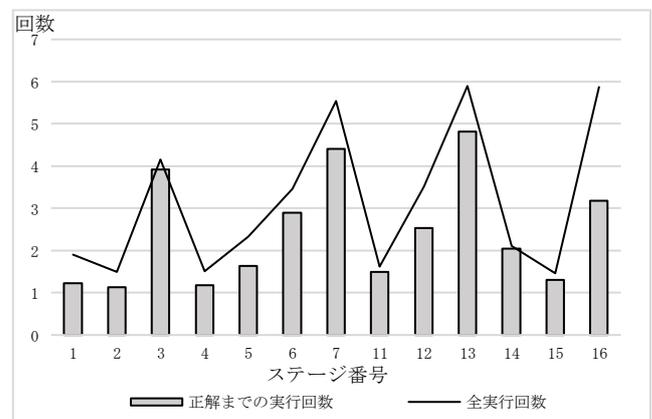


図 3 全実行回数と正解までの実行回数 (平均)

ステージ番号 7 と 13 は、「順次処理」、「条件分岐処理」、「繰り返し処理」の 3 つを含むステージとなったため、正解までの回数が多く、実行回数も多くなった。またステージ番号 3 は「順次処理」と「条件分岐」のみのステージだが、初めて複数の処理の組み合わせを体験するステージであったことが、回数が多い原因であると考えられる。

ステージ番号 16 は、実行回数は多いが、正解までの回数が少なかった。これは、実行は何度も行っているものの、正解にたどり着けなかった子供がいたためである。題材とした「ドッジボール」は、全てを表現しようとする動きが複雑で難しくなってしまうため、ツールではボールを投げる自分と相手だけの動きを事象にした。その結果、子供たちが行っているドッジボールと異なってしまう、正解に達する事ができなかった可能性が高い。子供たちが理解できる動きや事象と問題のレベルについては、今後の課題として検討する必要がある。

4.4 アンケート結果

アンケートの結果を表 6～8 に示す。82.0%の子供が「とても楽しかった／楽しかった」と答えており、概ね好評だったと言える。また、78.6%の子供が「とても使いやすかった／使いやすかった」と答えており、使いやすさの面でもツールの完成度は高かったと言える。しかし、難易度に関しては、「とても簡単だった／簡単だった」と回答している子供が 40.1%いる一方で、「とても難しかった／難しかった」と回答している子供が 29.0%いた。難易度に関しては、子供のやる気や理解に影響を与える大きな要因であるため、それぞれの子供に合わせたステージの設定が重要な課題となる。この課題の解決には、多様な難易度のステージの構築と子供のレベルに合わせる方法の検討が必要となる。

表 6 ツールは楽しかったか

回答	割合
とても楽しかった	58.1%
楽しかった	23.9%
普通	14.5%
つまらなかった	0.9%
とてもつまらなかった	0.0%
未回答	2.6%

表 7 ツールは使いやすかったか

回答	割合
とても使いやすかった	43.6%
使いやすかった	35.0%
普通	12.9%
使いにくかった	1.7%
とても使いにくかった	3.4%
未回答	3.4%

表 8 ツールは難しかったか

回答	割合
とても難しかった	8.5%
難しかった	20.5%
普通	28.2%
簡単だった	8.5%
とても簡単だった	31.6%
未回答	2.7%

5. まとめ

本研究では、プログラミング的思考の向上を目的としたツールの開発を行い、その有用性を検証した。

実際に、小学校 2 校で利用実験と評価を行ったところ、1 校目ではプログラミング的思考の向上に効果が見られなかったが、2 校目では効果が見られた。これは、プログラミング的思考の評価方法の違いが大きく影響している可能性がある。また、ファシリテーターの教え方に差があったため統制された評価になっていないことも一因と考えられる。今後は、参加者に対する接し方を、ファシリテーター内で統一し、ツールを使用した効果を正確に測定することが重要な課題となる。

また、アンケートでは、参加者の約 8 割の子供から楽しく学習でき、ツールは使いやすかったと回答があり、本ツールは小学生が利用するのに適している事がわかった。しかし、約 3 割が難しいと感じており、子供たちの理解に合わせた動きや事象と問題のレベルの設定については、再度検討する余地がある。

謝辞 実験に参加して頂いた生徒、保護者方、協力して頂いた先生方に、謹んで感謝の意を表します。

参考文献

- [1] 文部科学省. “小学校段階におけるプログラミング教育の在り方について (議論の取りまとめ) 平成 28 年 6 月 16 日”. http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.htm (参照 2018-01-15).
- [2] Vidushi Chaudhary, Agrawal Pragya Sureka, Ashish Sureka. An Experience Report on Teaching Programming and Computational Thinking to Elementary Level Children using Lego Robotics Education Kit, 2016 IEEE Eighth International Conference on Technology for Education, p.38-41.
- [3] 萱津理佳, 矢澤星奈. 初等中等教育段階におけるプログラミング教育の考察: プログラミング体験教室の実践から [研究ノート], 長野県短期大学紀要 71 巻, p.13-22 (2016).
- [4] 尾花拓海, 鈴木龍成, 吉村明人, 臼田莉菜, 半澤魁士, 佐久間拓也, 川合康央, 池辺正典. “小・中学生を対象にしたプログラミング学習を支援するアプリケーション開発プロジェクト”, 情報教育シンポジウム」2017 年 8 月, Vol.2017, No.32, p.202-205(2017).
- [5] 川本佳代, 出口直輝, 林雄介ほか. 論理的思考力育成を指向したフローチャート活用学習システムと小学校児童による実験的評価, 教育システム情報学会誌, Vol.32, No.3, p.214-219(2015).
- [6] 小林祐紀, 兼宗進. “コンピューターを使わない小学校プログラミング教育 “ルビィのぼうけん” で育む論理的思考”, 翔泳社(2017/03/31).
- [7] 豊田充崇. 小学校プログラミング授業の推進における実践上の課題, 和歌山大学教職大学院紀要. 学校教育実践研究, Vol.2, p.83-90(2018).
- [8] NPO 法人 CANVAS. “プログラミングで育成する資質・能力の評価規準(試行版)”. <https://beneprog.com/wp-content/uploads/2018/08/ver2.0.0.pdf> (参照 2018-10-17).