

プログラミングの考え方を学ぶドリル教材の提案

島袋舞子^{1,a)} 井戸坂幸男² 兼宗 進¹

概要：本研究では、プログラムの処理の流れや考え方を、小学生が紙の上で問題を解くことによって学ぶことができるドリル教材を提案する。ドリル教材はプログラミングに関する学習項目を検討し、難易度や発達段階に応じた順に並べて出題することで、段階的に学ぶことができるようにした。本稿では、開発したドリル教材の概要と小学生を対象としたイベントで実践した結果を報告する。

キーワード：プログラミング教育、プログラミング的思考、アンブラグド、ドリル教材

1. はじめに

2020年度から小学校でプログラミング教育が導入される。小学校段階のプログラミング教育では、情報活用能力の1つとして各教科の中でプログラミングの体験を通して、プログラミングの基本的な考え方を身につけることを目的としている [1]。児童はプログラミングを体験することで、コンピュータやプログラミングの考え方を学ぶことができる。一方で、すぐにプログラムを修正し、実行することができるため、プログラムの動きをあまり考えずに試行錯誤をくり返すことで、プログラムの考え方の理解が不十分であっても、プログラムが偶然意図したとおりに動作してしまう場合があることが報告されている [2]。

そこで、プログラミングの考え方を学ばせることを目的とし、紙の上で問題を解くことを通してプログラミングの考え方が学べるドリル教材を開発した。開発したドリル教材の概要と小学生を対象としたイベントで実践した結果を報告する。

2. 小学校段階におけるプログラミング教育

小学校の次期学習指導要領解説 [3] では、総合的な学習の時間の「プログラミングを体験しながら論理的思考力を身に付けるための学習活動」、5年生の算数科「B 図形 (1) 平面図形の性質」の「(ウ) 正多角形」、6年生の理科「(4) 電気の利用」の「ア (ウ) 身の回りには、電気の性質や動きを利用した道具があること」にプログラミング活動が例

【教科】
総合的な学習の時間 (9) プログラミングを体験しながら論理的思考を身につけるための学習活動
【例示部分の抜粋】
プログラミングを体験しながら、生活を便利にしている様々なアプリケーションソフトはもとより、目には見えない部分で様々な製品や社会のシステムなどがプログラムにより動いていることを体験的に理解するようにすることが考えられる。
【教科】
算数 第5学年 B 図形 (1) 平面図形の性質 (ウ) 正多角形
【例示部分の抜粋】
正多角形の作図を行う学習に関連して、正確な繰り返しを行う必要があり、更に一部を変えることでいろいろな正多角形を同様に考えることができる場面などで取り扱う。
【教科】
理科 第6学年 (4) 電気の利用 ア (ウ) 身の回りには、電気の性質や動きを利用した道具があること
【例示部分の抜粋】
実際に目的に合わせてセンサーを使い、モーターの動きや発光ダイオードの点灯を制御するなどといったプログラミングを体験的に学習するといったことが考えられる。

図1 プログラミング学習活動の例示部分の抜粋

示されている。例示部分の抜粋を図1に示す。

学習指導要領に例示された内容や教科以外でも、児童・生徒に負担の無い範囲でプログラミング教育を取り入れることが推奨されており、小学校段階のプログラミングに関する学習活動の分類 [1] の範囲で様々な教科で授業実践が進められている [1][4][5][6]。

また、コンピュータを使わずに紙や鉛筆、カードなどの

¹ 大阪電気通信大学
Osaka Electro-Communication University, Shijonawate, Osaka 575-0063, Japan

² 三重県多気郡大台町立宮川小学校
Miyagawa Elementary School

^{a)} shimabuku.m@gmail.com

教具を使った活動を通してプログラムの動きを考えさせる学習法も存在する。また、計算手順や図形の分類などをフローチャートで表すことで、コンピュータを使わずにプログラミングの考え方を教科の学習活動に取り入れている実践 [7] も存在する。

このように様々な学年でプログラミング言語や教材を利用した実践が進められているが、学習する内容の難易度や発達段階に応じて学習内容をまとめた教材は多くない [8]。

本研究では、ドリル教材を作成するにあたり、プログラミングの視点から学習内容を検討し、発達段階に応じた学習ができるように設計した。以下に、ドリル教材の設計についての考え方を示す。

3. プログラミングの考え方を学ぶドリル教材

3.1 ドリル教材で扱う内容の検討

小学校段階でのプログラミング教育では、どの学年のどの科目で、どの内容を学ぶといったカリキュラムは各小学校に委ねられており、学年別の学習内容は定められていない。そのためドリル教材で扱う学習項目を複数の体系的なカリキュラム案から検討する。

小学校の次期学習指導要領で例示された内容 (図 1) では、5年生では反復処理、6年生では分岐処理を教科の中で扱う。文部科学省が公開している「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) [9]」では、小学校段階におけるプログラミング教育で育成する知識・技能として「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づくこと」[9]と定義している。この定義に基づき作成された評価規準も存在する [10]。その評価規準の中では、1,2年生で順次処理、3,4年生で反復処理、5,6年生で分岐処理を知識・技能として身につけることを提案している。また、実践研究をふまえて作成された情報活用能力の体系表例も存在する [11]。この体系表例ではステップごとに項目が設けられている。小学校段階に相当するステップ 1, 2, 3 でプログラミングに関連する学習内容としては、「基本的な問題解決の手順」「単純なくり返し・条件分岐・データや変数などを含んだプログラムの作成、評価、改善」「図示 (フローチャート) などによる単純な手順 (アルゴリズム) の表現方法」[11]がある。また、久野ら [12] は、1,2年生で手順実行、3,4年生で条件分岐、5,6年生で反復、反復と条件分岐を組み合わせた手順を段階的に学ばせることを提案している。

次期学習指導要領における例示と 3つの提案を表 1 にまとめた。いずれの提案も 1~6 年を 3つの学年区分にわけており、1,2年生で手順実行 (順次処理) を扱っている。

そのことから、今回提案するドリル教材では、次の内容を扱うことにした。

- 基本的な制御構造 (順次、反復、分岐)

表 2 各学年区分で扱う内容

学習項目	1,2年生	3,4年生	5,6年生
順次処理	○	○	○
反復処理	○	○	○
分岐処理	○	○	○
変数	○	○	○
配列	-	-	○
関数定義	-	○	○
情報科学	○	○	○
アルゴリズム	○	○	○
データ活用	-	-	○
移り変わり図	-	-	○

- フローチャート
- 変数
- 配列
- 関数定義
- 情報科学
- アルゴリズム

順次処理、反復、条件分岐の中では、フローチャートも扱うことにする。学年を 1,2年生、3,4年生、5,6年生の 3つの区分にわけ、それぞれの学年区分において扱う内容を検討することにした。表 1 には含まれていないが、プログラム作成時に扱う考えである配列、関数定義、情報科学についても扱うことにした。配列については、変数の応用との位置づけで要素の値を参照、代入をすることを扱うことが可能と考えた。関数定義については、単純な手順実行をまとめたものであれば扱うことが可能だと考えた。情報科学では、二進法や真値等について扱う。アルゴリズムでは、値の交換や探索、整列などの動きを問題を通して体験することを目的として扱う。

3.2 各学年区分で扱う学習項目の検討

新学習指導要領の例示より、児童は 5年生の算数で反復処理、6年生の理科で分岐処理や変数を含んだプログラムを作成すると思われる。そのため 3,4年生までにある程度、基本的なプログラムの考え方を身につけておくことが必要と考え、1,2年生から順次処理、反復処理、分岐処理、変数を扱うことにする。このとき、順次処理、反復処理、分岐処理については子どもたちがつまづかないように具体的な指示から、フローチャートへと段階的に抽象度を上げていくことに留意する。また、変数については多くの要素を含んでいるため、学習する内容を階層的に考えることにした [13][14]。図 2 に変数の理解構造を示す。下の内容が上の内容を学ぶための前提条件となる。配列と関数定義についても、変数と同様に考えることにした。それぞれの学年区分で扱う内容 (学習項目) を表 2 に示す。

1,2年生では、基本的な考え方を身につけるため「順次」「反復」「分岐」を中心に扱う。身近なものを題材にした問

表 1 プログラミング学習活動の例示部分の抜粋

学年	学習指導要領の例示 [3]	資質・能力の評価規準 [10]	情報活用能力の体系例 [11]	久野 [12]
1, 2 年生	—	手順実行	手順実行	手順実行
3, 4 年生	—	繰り返し	繰り返し 条件分岐 変数	条件分岐
5, 6 年生	繰り返し (5 年算数) 条件分岐 (6 年理科)	条件分岐	アルゴリズムの改良 フローチャートによる手順の表現	繰り返し 繰り返しと分岐の組み合わせ

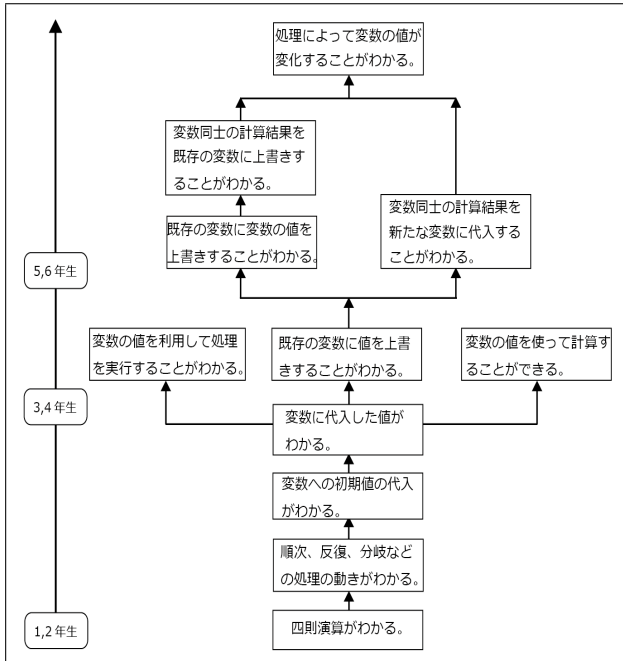


図 2 変数の理解構造

題で考え方を身につけたのち、フローチャートで図示された処理内容の実行結果を考える。最終的にはアイコン型のプログラミング環境で記述されたプログラムが読めるように段階的に学べるようにする。変数や情報科学の内容については、身近なものを題材にした問題で考え方を学習する。

3, 4 年生では、1, 2 年生のドリルと同様に「順次」「反復」「分岐」を扱う。最終的にはブロック型のプログラミング環境で記述されたプログラムが読めるように段階的に学べるようにする。また、新たに「関数定義」と「アルゴリズム」を扱う。関数定義では、1つの引数を取る関数の利用と引数のない関数定義について扱う。

5, 6 年生では、「反復や分岐が入れ子になったプログラムの動き」や「変数の値の変化によって、異なる処理をする」といった応用的な内容を扱う。また、テキスト型のプログラミング環境で記述した変数値への代入や計算式、関数についても扱う。最終的にはテキスト型のプログラミング環境で記述されたプログラムが読めるように段階的に学べるようにする。また、中学校への接続を意識し、5, 6 年生でプログラムの振る舞いの図示方法を考える「移り変わり図」として、状態遷移図やシーケンス図、アクティビティ図などを小学生向けにアレンジして扱うことにする。

上記の学習項目には、小学生には高度な内容も含まれるが、出題方法によって理解が可能になると考えた。

3.3 問題の出題手法の検討

以下の点に留意して、問題を作成した。

- (1) 児童に親しみやすく生活に身近なものを題材にする。
- (2) プログラムの記述方法を問うのではなく、学習項目の考え方について問うものとする。
- (3) プログラミングを体験したことがなくても思考することで解くことができるようにする。
- (4) 出題する学年区分に応じた難易度にする。

(1) については、カードやトランプ、お掃除ロボットなど、子どもたちの生活に身近なものを題材にすることで、プログラムの動きをイメージしやすくなると考えた。イメージしやすい具体物で考え方が身についたところで、少し抽象度の高いフローチャートの問題に進むといったように段階的に抽象度をあげていくことで、児童のつまづきが少なくなるようにする。

(2) と (3) については、初めて扱う題材の1問目には考え方の例示を入れることでプログラミングを体験したことがなくても、考え方を理解し、思考することにより解くことができるようにする。

(4) については、同じ内容を問う問題でも発達段階によって難易度や文章の表現方法が異なるため、学年区分ごとにそれぞれの項目の概念について扱う問題を作成する。問題で扱う漢字は、学年別漢字配当表 [15] に準拠する。1, 2 年生では、文章の量は最低限に挿絵や図を多めにすることで、視覚的にわかりやすくする。問題の解答は言葉ではなく、絵や図を選ぶものを多めにする。3, 4 年生では、文字で説明できる内容については文章で説明し、必要に応じて挿絵や図を使用することで視覚的にわかりやすくする。問題の解答は、主に絵や図を選ぶものにするが、1, 2 年生よりも言葉を記入することを多めにする。5, 6 年生では、文字で説明できる内容については文章で説明する。また、問題の解答は主に言葉を記入することにする。

これらの点をふまえて、それぞれの学年区分に応じた問題を作成した。

3.4 作成した問題

前節の内容をふまえて、問題を作成した。作成した問題

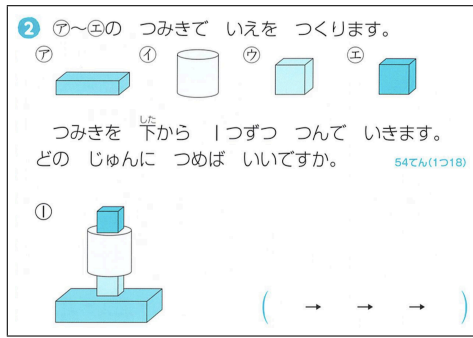


図 3 順次処理の概念を扱う問題 ([16] より一部抜粋)

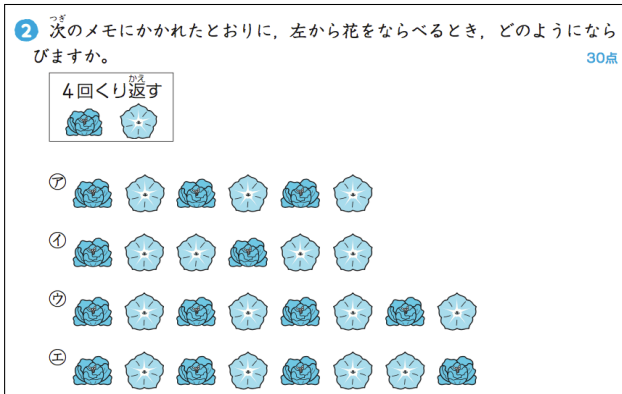


図 4 反復処理の概念を扱う問題 ([17] より一部抜粋)

は、プログラミングの基本的な考え方を問題の解き方やルールとして取り入れている。ここでは、3つの学年区分に共通する順次処理、反復処理、分岐処理、変数の問題を紹介する。

3.4.1 順次処理

図 3 は 1, 2 年生向けに作成した問題である。この問題は「4種類のつみきを1つずつ積んでいく」というルールに従って完成例と同じようにつみきを積むためには、どの手順でつみきを積んでいけばよいかを問う。学習者は、つみきという具体物を積んでいく手順を考えることをとおして、「コンピュータは命令した手順のとおり命令を実行する」という順次処理の考え方を学ぶ。

3.4.2 反復処理

図 4 は 3, 4 年生向けに作成した問題である。この問題は、メモのとおり花を並べたとき、どのような順に花が並べられるかを考える。メモには花の並びとそれを繰り返す回数が書かれている。メモに書かれたとおりに花を並べるときをくり返したとき、どのように花が並ぶのかを考えることをとおして、「コンピュータは1つの命令のまとまりを指定された回数だけ繰り返し実行する」という反復処理の考え方を学ぶ。

3.4.3 分岐処理

図 5 は 5, 6 年生向けに作成した問題である。この問題は、ベルトコンベアから流れてくるパンの種類を2色の旗を使って知らせたとき、挙げた旗によってどのパンが流れ

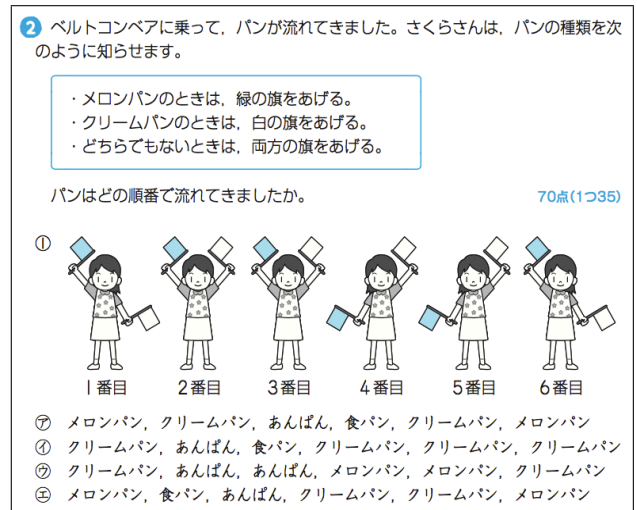


図 5 分岐処理の概念を扱う問題 ([18] より抜粋)

てきたのかを考える。旗はメロンパンが流れたときは緑色、クリームパンのときは白色、どちらでもないときは両方の旗をあげる。あがった旗から、どのパンが流れてきたのかを考えることをとおして「コンピュータは条件によって実行する命令を変える」という分岐処理の考え方を学ぶ。

3.4.4 変数

図 6 は 1, 2 年生向けに作成した問題である。変数では「一つの変数には一つの値のみ保存できる」性質を「最後に伝えたかった言葉や数のみを記憶するロボット」に置き換えて出題した。ロボットが覚えている言葉を他のロボットへ伝えることで「変数値の上書き」、新しいロボットに新しい言葉を伝えることで「初期値の代入」、ロボットが何を覚えているか問うことで「変数値の参照」を扱うことができる。このようにロボットに置き換えることで、1, 2 年生から変数の考え方を扱うことができると考えた。

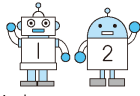
4. 小学生向けイベントでの実践

4.1 イベントの概要

小学生を対象にしたイベントで作成した問題を使用し、問題を解く児童の様子を観察した。イベントは2部構成になっており、第1部では問題に取り組む。第2部では教育用プログラミング言語「ドリトル」[19]を利用してプログラミングを体験する。イベントの流れを図 10 に示す。参加者は1年生から5年生までの21名である。

イベントで使用する問題は、1, 2 年生向けのドリトル教材 [16] から、どの学年でも楽しめる問題を「順次」、「反復」、「分岐」、「変数」の4つの学習項目から抽出した。問題はそれぞれ4ページずつで「順次」では、図 3 のようにつみきを上から積む問題、つみきを上から取っていく問題を選んだ。「反復」では、メモに書かれた順にお花を並べるときを指定された回数だけ繰り返す問題(図 7)、繰り返す1つのまとまりを考える問題を選んだ。「分岐」では、タイルに描く絵をタイルの形に応じて描き分ける問題を選んだ

① ロボットは さいごに つたえられた ことばを おぼえます。あおいさんは つぎのように ロボット1ごうと ロボット2ごうに ことばを つたえました。



40てん(1つ10)

ロボット1ごう←“こんにちは”
 ロボット2ごう←“ハロー”

① ロボット1ごうと ロボット2ごうが おぼえている ことばを ()に かきましよう。

1ごう ()

2ごう ()

② ゆうまさんは ロボット2ごうに ロボット1ごうが おぼえている ことばを おぼえるように めいれいしました。

ロボット2ごう←ロボット1ごう

ロボット1ごうと ロボット2ごうが おぼえている ことばを ()に かきましよう。

1ごう ()

2ごう ()

ロボット2ごうは、
 ロボット1ごうが
 おぼえて いる
 ことばを おぼえるよ。

71

図 6 変数の概念を扱う問題 ([16] より抜粋)

(図 8)。変数では、図 6 のようにロボットに伝えた言葉を考える問題、ロボットが覚えている言葉を他のロボットに伝えた時、それぞれのロボットがどの言葉を思っているかを考える問題を選んだ。イベントで使用したドリル教材の問題は付録に収録する。

また、フローチャートや経路探索を題材にしたチャレンジ問題を用意した(図 9)。チャレンジ問題は、ドリル教材の問題に近い問題を4問、フローチャートを扱った問題を2問、キャラクタを動かしてスタートからゴールまでのプログラムを考える経路探索的な問題を4問、それぞれの学習項目ごとに用意した。


児童はドリル教材の問題を解き、解き終えたらチャレンジ問題を解く。これを「順次」、「反復」、「分岐」、「変数」の順に実施した。このとき、児童は自分のペースで問題を解く。その後、第2部でタブレット端末を使ってプログラミングを体験した。キャラクターがアイテムを拾うゲームづくりを通して、解いた問題と実際のプログラミングとのむすびつきを確認した。

4.2 イベント中の児童の様子

第1部の問題を解く時間では、1,2年生の児童はわからない部分を保護者に聞いて問題に取り組んでいた(図 11)。問題の文章や挿絵から考え方を読み取ることができない児童は保護者が追加の説明をすると理解することができていた。3,4,5年生の児童は自分で解いていた。早く終わった子は近くの席の困っている子に教えている姿も見られた。

第2部のプログラミングを体験する時間では、児童は楽


② メモに かかれた じゅんばんで ^{ひだり}左から ^{はな}花を ならべます。




2かい くりかえすと どんな じゅんばんに なりますか。

20てん

ア



イ



ウ


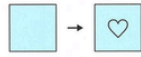


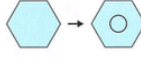
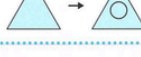
図 7 反復処理の概念を扱う問題 ([16] より一部抜粋)

① あおいさんは タイルに えを かきます。

しかくい タイルには ♡を かきます。




しかくては ない タイルには ○を かきます。


つぎの タイルで ^{ただ}正しく かかれて いるのは どれですか。

50てん


ア



イ



ウ



エ




図 8 分岐処理を扱う問題 ([16] より抜粋)

しそうに取り組んでいた(図 12)。また、保護者のサポートを受けながら、自分なりにプログラムを変更する姿も見られた。

4.3 児童が解いた問題の分析

児童が取り組んだ問題から、どの程度取り組むことができたかを確認する。学年別にそれぞれの問題に取り組んだ人数を表 3 に示す。表 3 より、「反復」まで全員が取り組むことができたといえる。1年生はドリル教材の「反復」から「分岐」に取り組むことができなかったのが3名、「分岐」から「変数」に取り組むことができなかったのが2名いることから、反復処理や分岐処理の考え方の理解に時間がかかるといえる。

2年生以上の学年では、ドリル教材の「変数」までほぼ全員が取り組むことができた。その一方で、ドリル教材の「変数」にたどりつけなかった7名中、そのうち6名が1,2年生だったことから、1,2年生は反復や分岐の考えを理解するのは容易ではなかったことが考えられる。

ドリル教材の問題の正答率を図 13 に示す。小学生では学年だけでなく生まれた月による発達段階の影響も考えら

2 ロボットは次の命令の図の順で動きます。1マス進むの命令でロボットは矢印の方向に1マス分進みます。

(1) ロボットはどのマスで止まるでしょうか。記号で答えましょう。

答え:

※保護者メモ：お子さんの理解度は(A・B・C)です。

3 カメのロボットに命じて赤い城を目指します。ロボットは黄色の道を進みません。このとき、壁(水色)にあたってはいけません。

(1) ロボットに次の命令をしました。ロボットは城に着きますか？「はい」か「いいえ」で答えましょう。

前に進む
前に進む
前に進む

答え:

※保護者メモ：お子さんの理解度は(A・B・C)です。

図9 順次処理を扱うチャレンジ問題(一部抜粋)

構成	時間(分)	内容
第1部	5	導入
	45	自らのペースで問題に取り組む
	15	休憩
	45	自らのペースで問題に取り組む
	10	休憩
第2部	2	導入
	50	プログラミング体験
	3	まとめ、保護者へのアンケート

図10 イベントの流れ



図11 ドリル教材の問題を解く様子

れるため、前後の2学年で2項移動平均を求めた(図14)。ただし、1,2年生の児童の多くは保護者と一緒に問題に取り組んだため、3,4,5年生と分けて考えることにした。

1,2年生の正答率を見ると、「分岐1」以降で正答率が下がっている。これは問題に取り組む時間が足りず、問



図12 プログラミングを体験する様子

表3 問題に取り組んだ人数(学年別)

学年	参加人数	順次	反復	分岐	変数
1	9	9	9	6	4
2	6	6	6	6	5
3	3	3	3	3	2
4	1	1	1	1	1
5	2	2	2	2	2

題を最後まで取り組めなかった児童が存在するからだと考える。問題に取り組んだ児童のみの正答率は、「分岐1」が100%(12名)、「分岐2」が97%(12名)、「変数1」が97%(9名)、「変数2」が93%(9名)であることから、取り組んだ児童は問題を解くことができたことがわかる。

3,4,5年生の正答率を見ると、「順次2」「反復1」「変数」以外の項目は90%以上の正答率だった。一方で「順次2」や「反復1」は、3年生が3名、4年生が1名、5年生が2名と分析できるデータが少ないため、正答率が低くなってしまっていると考えられる。学習項目別に学年間の正答率に差があるかを確認するため2項移動平均した値でMann-WhitneyのU検定をおこなったところ、学年間に有意な差は見られなかった(表4)。

また、正答率が最も低い「変数2」の児童の解答を詳しく見ると、児童は同一の箇所、同様の誤答で間違えていることがわかった。変数2は図6のように、ロボットに伝えた言葉を答える問題である。この問題では、変数への初期値の代入と変数値の上書きを扱う。誤答の箇所を確認すると、図6の(2)のようなロボット1号が覚えている言葉をロボット2号へ伝える問題で間違えていた。このとき、2台ともロボット1号が覚えている言葉を覚えるが、多く児童の誤答はロボット1号が「ハロー」、ロボット2号が「こんにちは」のように2台のロボットが覚えている言葉が入り替わっていた。これらの要因は、ロボットは最後に伝えた言葉を覚えること(変数への初期値の代入)やロボット1号がロボット2号へ言葉を伝えるとロボット1号が覚えている言葉をロボット2号が覚えることができることは理解しているが、伝えた側のロボット1号が覚えている言葉

学年	人数	順次1	順次2	反復1	反復2	分岐1	分岐2	変数1	変数2
1	9	100	89	100	98	67	63	44	42
2	6	100	100	100	98	100	100	78	77
3	3	100	67	100	100	100	96	67	58
4	1	100	100	60	100	100	100	100	100
5	2	100	100	100	100	100	100	83	88

図 13 学年別の正答率 (1, 2 年は保護者と回答)

学年	人数	順次1	順次2	反復1	反復2	分岐1	分岐2	変数1	変数2
1,2	15	100	94	100	98	83	81	61	59
学年	人数	順次1	順次2	反復1	反復2	分岐1	分岐2	変数1	変数2
3,4	4	100	83	80	100	100	98	83	79
4,5	3	100	100	80	100	100	100	92	94

図 14 正答率の2項移動平均 (1, 2 年は保護者と回答)

	正答率 (%)		p 値
	3, 4 年 (N=4)	4, 5 年 (N=3)	
順次	90	100	1
反復	92	92	1
分岐	98	98	1
変数	92	95	0.8286

* 0.01 ≤ p < 0.05 ** 0.001 ≤ p < 0.01 *** p < 0.001

がどうなるのか、わからなくなってしまったためだと考えられる。

4.4 考察

イベント時に児童が問題を解く様子を目視で観察し、児童が解いた問題を分析した。1, 2 年生は保護者と一緒に問題に取り組む様子が多く見られた。1, 2 年生では、時間内に問題をすべて解くことができなかつた児童が多く見られたため、「分岐1」から「変数2」まで徐々に正答率が下がっている。その一方で、問題に取り組んだ児童の「分岐」「変数」の正答率は90%以上であったことから、1, 2 年生でも時間をかければ問題を理解し、解くことができると考えられる。また、保護者はプログラミング未経験の場合が多いがドリル教材の問題を理解し児童に教えていたことから、成人であればドリル教材の問題を通して、初見でプログラミングの概念を理解できることが示唆される。

3 年生以上は、ほとんどの児童が時間内にドリル教材の問題を解くことができた。また、3, 4 年生と 4, 5 年生で学習項目別の正答率に有意な差がなかったことから、学年間に理解の差はほとんどないと考える。

5. おわりに

小学生を対象にプログラミングの考え方を段階的に学ぶことができるドリル教材の提案をした。今後は教育現場で実践し、教育効果について検証を進めていきたい。

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2版), <http://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1403162.htm> (参照 2018-11-24).
- [2] 本郷健, 山下成明: プログラム作成プロトコルの分析による思考過程の考察, 教育情報研究, Vol.9, No.3, pp50-59 (1994).
- [3] 文部科学省: 小学校学習指導要領解説 (2017).
- [4] 森秀樹, 杉澤学, 張海, 前迫孝憲: Scratch を用いた小学校プログラミング授業の実践: 小学生を対象としたプログラミング教育の再考, 日本教育工学会論文誌, Vol.34, No.4, pp.387-394 (2011).
- [5] 三井一希: 学習者の相互作用を軸とした小学校低学年におけるプログラミング教育の実践, コンピュータ&エデュケーション, Vol.40, pp.61-66 (2016).
- [6] 三井一希: 小学校国語科の「書く活動」へのプログラミング導入による学習効果, 教育システム情報学会誌, Vol.34, No.1, pp.60-65 (2017).
- [7] 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).
- [8] 太田剛: 系統的学習を考慮した小学校プログラミング教育用教材の試作, 日本 STEM 教育学会, 第 1 回年次大会, pp.10-13 (2018)
- [9] 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), <http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.htm> (参照 2019-05-21).
- [10] ベネッセコーポレーション: 第 2 版「プログラミングで育成する資質・能力の評価規準(試行版)」Ver.2.0.0 (2018/8/21 版), <<https://beneprog.com/wp-content/uploads/2018/08/ver2.0.0.pdf>> (参照 2019-05-24).
- [11] 文部科学省: 次世代の教育情報化推進事業「情報教育の推進等に関する調査研究」成果報告書, <http://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1400796.htm> (参照 2018-11-24).
- [12] 久野靖, 和田勉, 中山泰一: 初等中等教育を通じた情報教育の必要性和カリキュラム体系の提案, 情報処理学会論文誌「教育とコンピュータ」, Vol.1, No.3, pp.48-61 (2015).
- [13] 保福やよい: なぜプログラミングは難しいのか? 繰り返しの理解構造と C の教科書分析からのアプローチ, 情報処理, Vol.54, No.3, pp.252-255 (2013).
- [14] 鈴木克明: 教材設計マニュアル-独学を支援するために-, 北大路書房 (2002).
- [15] 文部科学省: 学年別漢字配当表, <http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/011/siryo/attach/1400158.htm> (参照 2019-5-20).
- [16] 島袋舞子, 兼宗進: ドリルの王様 1, 2 年生のたのしいプログラミング, 新興出版社啓林館 (2019).
- [17] 島袋舞子, 兼宗進: ドリルの王様 3, 4 年生の楽しいプログラミング, 新興出版社啓林館 (2019).
- [18] 島袋舞子, 兼宗進: ドリルの王様 5, 6 年生の楽しいプログラミング, 新興出版社啓林館 (2019).
- [19] 清水健吾, 本多佑希, 島袋舞子, 兼宗進: ドリトル言語をベースとしたビジュアルプログラミング環境の提案. 日本産業技術教育学会, 第 34 回情報分科会 (2019).