

小学校における「プログラミング的思考」の育成

佐伯市立切畑小学校 波多野 美代子

要旨

本研究は、プログラミングの経験の少ない小学校高学年の児童に向けた授業モデルを作成し、プログラミング的思考を育むことを目的とした。

授業モデルではプログラミングの基礎を学び、地域や学校に関する身近なテーマをもとにプログラミングの作品を制作する学習活動を行った。制作の始めにゲームの構成を表した「流れ図」を作成し、伝えたいことやキャラクターの動きについて意見交換しながら、作品制作を行った。「分解」※1「抽象化」※2「一般化」※3する学習活動を取り入れた授業モデルをもとに、実践授業を行ったことで、プログラミング的思考が高まることが確認された。

〈キーワード〉 プログラミング的思考 Scratch 流れ図 プログラミング教育

※1「分解」とは、プログラミング的思考において「大きな事象や動きを解決可能な小さなものに分けること」を表す。

※2「抽象化」とは、プログラミング的思考において「分解した事象や動きの中から適切なものを選び出し、他の部分を除くこと」を表す。

※3「一般化」とは、プログラミング的思考において「類似性や関係性を見だし、別の場合でも利用できる内容にすること」を表す。

I 研究の背景と目的

1 研究の背景

(1) 現状

平成29年7月に告示された小学校学習指導要領解説の総則編において、「将来どのような職業に就くとしても時代を超えて普遍的に求められる『プログラミング的思考』を育むため、小学校においては、児童がプログラミングを体験しながら、コンピュータに意図した処理を行わせるために必要な論理的思考力を身に付けるための学習活動を実施すること」とされている(注1)。

大分県では「大分県学校教育情報化推進計画『ICT活用教育推進プラン2022』」の取り組みにおいて、ICTを活用した情報活用能力の育成を目指す中で、プログラミング教育の充実を図っている。佐伯市においても、小・中学校にICT支援員を配置し学校のICT環境や指導体制の整備に努め、プログラミング教育の研修会を実施して教育の情報化を進めている。

所属校では、ICT支援員と連携を図りながらプログラミング教育への取り組みを進めている。しかし、教育課程で扱う回数は少なく、プログラミング教育への取り組みは順調とは言えない。児童がプログラミングを学ぶ機会は、ほとんど無いのが現状である。教員もプログラミングの経験は少なく、何をどのように教えればよいか、指導への不安は大きい。児童の実態を把握し、プログラミング的思考を育成するための授業モデルの作成が必要であると考ええる。

(2) 「プログラミング的思考」について

小学校プログラミング教育の手引き(第三版)において、プログラミング的思考は「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み合わせが必要で

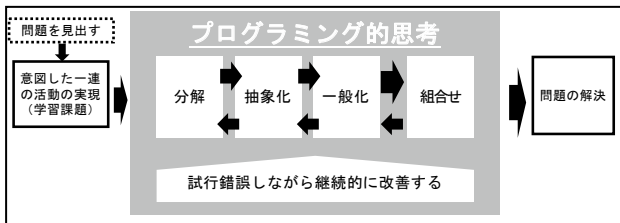
あり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組み合わせをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力」と定義されている(注2)。

ベネッセコーポレーション(2018)では、文部科学省の定義をもとにプログラミング的思考を整理している。ベネッセは、プログラミング的思考の4つの要素として、「分解」「抽象化」「一般化」「組合せ」を挙げている。「分解」とは、大きな事象や動きを解決可能な小さなものに分けることとされている。単純に分けるだけでなく、問題解決に適した大きさに「分解」することが必要となる。「抽象化」とは、分解した事象や動きの中から適切なものを選び出し、他の部分を除くこととされている。「一般化」とは、類似性や関係性を見だし、別の場合でも利用できる内容にすることとされている。「組合せ」とは、目的に合わせてより良い手順を創ることとされている。

プログラミング的思考では、「分解」「抽象化」「一般化」したものを、「組合せ」することで問題解決を図っていくことが望ましいと考える。そのため、文部科学省(2020)のプログラミング的思考を働かせるイメージと、ベネッセの4つの要素を取り入れて、プログラミング的思考のイメージを作成した(資料1)。

私のこれまでのプログラミングの実践を振り返ると、所属校の児童において、プログラミング的思考の重要な要素となる「分解」「抽象化」「一般化」の部分に課題を感じていた。学校生活でも、自分の考えを伝えるとき、相手や場面に応じて伝える内容や伝え方を考えることが苦手な様子などがみられた。

<資料1> プログラミング的思考を働かせるイメージ



(3) 課題

このような現状をふまえ、児童の苦手を改善し「プログラミング的思考」を育成することを課題とする。

2 研究の目的

児童の苦手を改善する授業モデルをもとにした実践授業を行い、児童のプログラミング的思考を高めることを目的とする。

II 仮説

児童の困りに応じた授業モデルをもとにした実践授業を行うことで、プログラミング的思考が高められるだろう。

III 研究の方法と内容

1 児童のプログラミング的思考に関する実態調査

児童のプログラミング的思考に関する実態を把握するため、実態調査を実施した。実態調査の概要については、以下(資料2-①)に示す。

<資料2-①> 実態調査の概要

目的	対象児童の「プログラミング的思考」に関する実態把握のため
対象	佐伯市立切畑小学校 第5学年(16名) 第6学年(15名)
時期	(事前) 令和4年9月1日(木) (事後) 令和4年10月31日(月)・11月1日(火)
調査方法	ペーパーテスト・アンケート(選択式)

(1) 「プログラミング的思考」に関するテスト

プログラミング的思考のテストは(資料2-②)に示した内容で作成した。テストは3問設定した。問1と問3は、事前調査と事後調査において、できるだけ近い形になるよう問題を作成した。問2は、事前調査で記述した自分の解答を事後調査で振り返らせ、命令の改善点を書き加えさせる形とした。

問1はプログラミング的思考全般を見取るために、エラーが出ないように命令させる問題である。問2は主にプログラミング的思考における「分解」「抽象化」する

ことについて見取くことをねらいとし、できるだけ詳しく命令を書き出させる問題である。問3では、主に「抽象化」「一般化」することについて見取るため、スライドの類似性や関係性をとらえて必要な内容が何かを考えさせる問題である。

<資料2-②> 「プログラミング的思考」に関するテストの内容

問1 プログラミング的思考全般の力を測る
車のロボットをゴール地点まで動かすために必要な命令を書き出す問題

問2 主に「分解」「抽象化」の力を測る
意図した動きをさせるために必要な命令を書き出す問題(事前・事後共通)

問3 主に「抽象化」「一般化」の力を測る
発表内容全体の構成と、それぞれのスライドの役割を理解し必要な内容を考える問題

3. ゆうきさんは近所の〇〇川に住む生き物について、調べたことをクラスで発表します。そこで、発表の時にみんなに見せるパネルを①～⑥の順に6まいつくりました。

① 〇〇川の生き物しらべ ② 問題 〇〇川にはどんな生き物が住んでいるでしょう。 ③ 解説 ④ ドンコははーばせ科の魚、きれいな水に住んでいる。 ⑤ 生き物が減った原因 ⑥ 未来のために

④ しかし、最近では... 〇〇川の生き物の数はかなり減っている! ⑤ ・はんらんしないための工事
・汚れた水
・ゴミ捨てなどのゴミ ⑥ 自分ができることをして、〇〇川に住む生き物を大切にしましょう。

あなたは、ゆうきさんの発表をより分かりやすくするために「パネルをもっと増やしてほしいよ」とアドバイスすることにしました。

(事後)

① 題字	盲導犬について
② 話題提示	問題「盲導犬はどんなことができるでしょう。」 答え「曲がり角を教えること、段差を教えることなど」
③ 解説	できること①「曲がり角にきたら、動きで教えてくれる。」
④ 問題1	数は...「日本国内で900頭くらい。年々減っている。」
⑤ 問題2	困っていること「犬に話しかけたりエサをあげたりする人がいること、病院や店に入れないことがあること」
⑥ まとめ	わたしたちは...「目に障がいのある人のために気を付けて過ごしましょう。」

(2) 「プログラミング的思考」に関するアンケート

プログラミング的思考に関する自己評価アンケートは、ベネッセの教師向けの評価規準を参考にし、高学年児童向けに、自己評価するものを作成した(資料2-③)。プログラミング的思考の育成には、「知識及び技能」「思考力・判断力・表現力等」「学びに向かう力、人間性等」それぞれの項目が必要となる。本研究では、特にプログラミング的思考を育む上で重要である「思考力・判断力・表現力等」の8項目を中心に分析した。

<資料2-③> 「プログラミング的思考」に関する自己評価アンケートの内容

＜知識及び技能＞	
身近なコンピュータ活用	a(コンピュータの仕組みを考える) 「コンピュータ」がどのようなところで使われていて、それがどのような仕組みなのかを考えることができますか。
	b(プログラムの仕組みを説明する) 「プログラム」がどのようなところでどのように使われているか、説明することができますか。
コンピュータに指示を出す手順	c(条件分岐の使い方がわかる) コンピュータに命令するとき、「～をしたとき」「もし△△なら」のような場合分け(分岐)の使い方がわかりますか。
	d(順次、反復、条件分岐に気を付けるとよいことがわかる) コンピュータに命令するとき、順番、くり返し、場合分け(分岐)に気を付けるとよいことがわかりますか。
	e(センサーがどのように活用されているかわかる) 身近な生活の中で、「センサー」がどのように活用されているか、わかりますか。
＜思考力、判断力、表現力等＞	
論理的に考えを進めること	f(目標を達成するために計画を立てる) 目標を達成するために自分が何をすればよいのか、計画を立てることができますか。
	g(理由やつながりを見つけ説明する) ものごとの理由や、物と物とのつながりを見つけ、説明することができますか。
動きに分ける 「分解」	h(大きな目標を小さな目標に分ける) 難しいことに取り組みむときに、大きな目標をいくつかの小さな目標に分けることができますか。
適切な大きさに分ける 「分解」「抽象化」	i(内容や量を工夫する) ものごとを伝えるとき、場面に応じて内容や量を工夫することができますか。
記号にする 「抽象化」	j(どのような方法が必要なかを考える) 目標を達成するために、どのような方法が必要なかを考えることができますか。
一連の活動にする 「一般化」	k(似ているところや関係するところを当てはめる) 新しいこととすでに知っていることや、いくつかのものごとの似ているところや関係するところを見つけ、別のことにも当てはめることができますか。
組み合わせる 「組合せ」	l(より良い順番を考える) 目標を達成するための方法の、より良い順番を考えることができますか。
振り返ること	m(どんな問題があるのかを考え改善する) 考えた方法や順番に、どんな問題があるのかを考え、改善することができますか。
＜学びに向かう力、人間性等＞	
コンピュータを生かそうとする態度	n(最後まで計画的にやりとげようとする) 目標ややり方を考えて、最後まで計画的にやりとげようとしていますか。
	o(コンピュータの便利な使い方を考える) 身近な生活の中で、どのようなところにコンピュータを使うと便利なかを、考えることがありますか。

(3) 事前調査から見た児童の実態

テストでは、問1のプログラミング的思考全般をみる問題において、エラーとなる解答をしていた児童は26%であった。命令の順序の間違いは無く、全て命令の不足による間違いだった。「組合せ」ではなく、「分解」「抽象化」「一般化」する部分に主なつまづきがあるのではないかと推察された。問2の、主に「分解」「抽象化」をみることをねらいとした問題においては、何らかの動きを命令することができた児童の割合は94%であった。しかし、表情についての命令ができた児童の割合が48%、腕の高さや向きについての命令ができた児童が全体の13%であった(資料3)。このことから、全体への漠然とした命令はできるが、細かな部分へのより詳細な命令はできない児童が多いと考えられる(「分解」「抽象化」ができていない)。

<資料3> 人型ロボットへの命令ができた児童の割合(n=31)

人型ロボットの動きについて何らかの命令ができた児童	94%
表情についての命令ができた児童	48%
腕の高さや向きについての命令ができた児童	13%

これらのことから、授業モデルにおいて、「分解」「抽象化」「一般化」することを中心とした学習活動を意識的に取り入れることで、児童のプログラミング的思考が向上すると考え、検証授業を実施した。

2 検証授業の実施

検証授業の概要については、以下(資料4-①)に示す。

<資料4-①> 検証授業の概要

検証内容	「分解」「抽象化」「一般化」する学習活動を取り入れた授業モデルを実施しプログラミング的思考を高める。
対象	佐伯市立切畑小学校 第5学年(16名) 第6学年(15名)
期間	令和4年10月14日(金)～10月28日(金)
分析の視点	児童のプログラミング的思考の高まりがみられたか。

<資料4-②> 授業モデルの内容

全5時間(5・6年共通)

時	「本時のねらい」・主な学習内容
	「プログラミングの基本処理と、プログラミング的思考について知ることができる。」
1	・授業モデルのゴール(3・4年生に、切畑や弥生のことを伝えるゲームを作ろう。)を知る。 ・プログラミングの基本処理(順次・反復・条件分岐)について知る。 ・何について紹介するか(大テーマ)を班で決める。
	「ゲームの構成やその内容を考えて、流れ図を作ることができる。」
2	・班ごとにどんなゲームにするか話し合い、小テーマをKJ法で整理して決める。 ・話し合ったことをもとに、ゲームの流れ図(場面の案)を作成する。 ・流れ図が低学年の友達にわかりやすいものになっているか、班で振り返る。
	「流れ図に沿った動きになるよう、Scratchでプログラミングすることができる。」
3	・班で協力してScratchで流れ図に合わせて動くようにプログラミングする。 ・困りを学級で共有する。 低学年の友達にわかりやすいものになっているか、班でプレイし合い振り返る。
	「もらった意見や話し合ったことをもとに、プログラミングすることができる。」
4	・3・4年生からもらった意見をもとに班で自分たちのゲームを振り返り、優先順位を付けて今日の計画を立てる。 ・必要であれば流れ図とプログラムを修正する。
	「『3・4年生にしっかり伝える』ゲームが作れているか振り返ることができる。」
5	・完成したゲームを、流れ図、修正の過程の説明を含めて学級で発表する。 ・お互いにプレイし合い、感想を伝え合う。 ・プログラミングの学習、プログラミング的思考の振り返りをする。

授業モデルは、黒田(2020)のアジャイル開発型協働学習における学習過程を参考に、計画・実行・評価・修正の小さなサイクルを繰り返す中で、自分たちの作品を何度も振り返り、修正することができるような授業を設定した。何度も振り返り改善しながら制作することで、プログラミング的思考が高まりやすいと考えたからである。また、大まかな計画をもとに、制作しながら細かな修正を行っていくので、プログラミング経験の少ない所属校の児童にとっても、取り組みやすくなる考えた。また、プログラミングのソフトには、プログラムを流

れ図と対応させやすいことと、小学校で広く使用されていることから、Scratchを使用した。

第1時でプログラミングの基礎を学び、第2時から地域や学校の良さを伝えるゲームを制作した。始めにゲームの構成を表した流れ図を作成した。流れ図をもとに、伝えたいことやキャラクターの動きについて意見交換しながら作品制作を行った。各時で、作品を振り返って次の計画を立てる時間を設定した(資料4-②)。

IV 結果

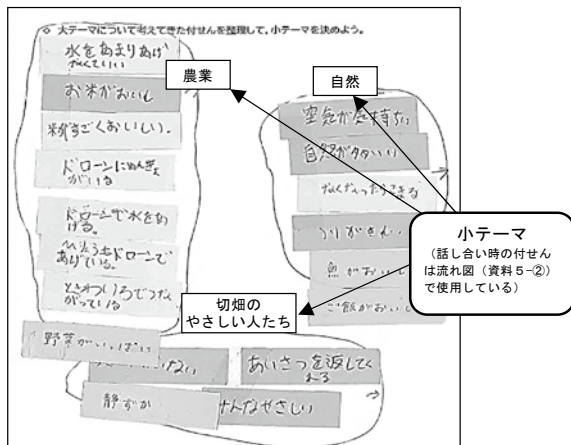
1 検証授業から

検証授業では、プログラミングの計画を立てる流れ図を作成する場面と、Scratchを使用しプログラミングする場面それぞれでプログラミング的思考を働かせることを想定した学習活動を設定した。

(1) 流れ図を作成する場面(主として「分解」「抽象化」する学習活動)

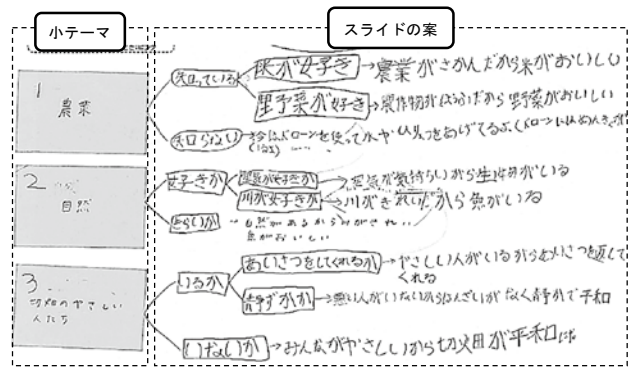
まず3・4年生にわかりやすく伝えるためにはどんな内容がよいか、出し合ったアイデアを整理しながら、小テーマとゲームの構成を考えて流れ図を作る活動を行った(資料5-①、5-②)。大テーマに決めた伝えたいことを「分解」し、流れ図に使用する小テーマへと「抽象化」していく活動である。

<資料5-①> 「分解」「抽象化」しているワークシートの例



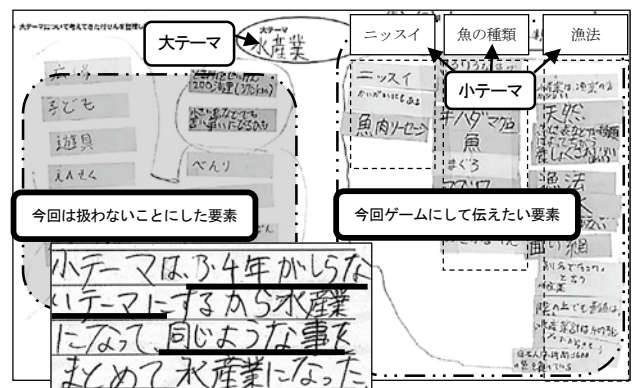
児童は始めに伝えたいことのアイデアを付せんに出していった(「分解」の学習活動)。その後、班でワークシート上に合わせていき、「これとこれは、まとめて〇〇とするといいね。」「これを伝えるためには〇〇もあるね。」「今回はこれはやめておこう。」というように、付せんを見ながら、いくつかの要素をまとめたり必要なものを選んだり(「抽象化」の学習活動)しながら、流れ図を作っていく姿が見られた。

<資料5-②> 流れ図のワークシートの例



3・4年生にゲームとして伝えることを考え、難しすぎるものやすでに知っているようなこと、ゲームの形で伝えにくいものなどを除いて、流れ図を作成した班もあった(「抽象化」の学習活動)(資料5-③)。

<資料5-③> 「抽象化」しているワークシートの例



(2) Scratchを使用しプログラミングする場面(プログラミング的思考を働かせる学習活動)

流れ図を作成した後、流れ図をもとにScratchでのプログラミングを行った。児童は、意図したゲームの場面を「背景」や「キャラクター」、「セリフなどの文字」、「それらの動き」といった構成要素でとらえていた。それぞれの動きを「分解」「抽象化」「一般化」し、「組合せ」をしてプログラミングしていった。

授業の中では、流れ図や制作物を自分たちで振り返ったり、3・4年生に感想をもらったりしながら、意図したものになっているか考える時間を設けた(「分解」「抽象化」の学習活動)。より3・4年生にわかりやすい形を考えて、流れ図やプログラムを改善する姿が見られた。以下(資料6-①)に示した児童の班では、ゲームの最後の場面において、地域のよさを伝えるために背景画像の利用が効果的であることに気が付き、メッセージを書き込み改善していた(資料6-②)。

<資料6-①> 「分解」「抽象化」に関わる振り返りの様子

(3・4年生の感想)

「背景の写真を見て
質問に答えることができて
楽しかったです。」

「答えるとき難しかったです。」

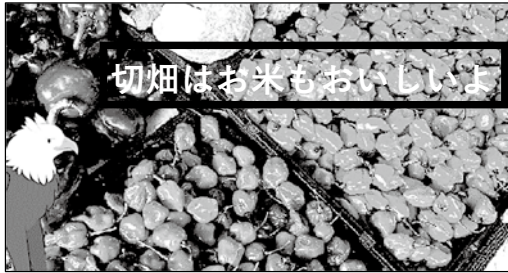
(6年児童の振り返り)

今日、四年生と三年生の感想を見てその感想を参考に
うにして少し作りなおした
たのでよか。たであ、
あと写真の背景に字を
打ちこんで分かりやすく
して字の色も考えられた。

するプログラムなので、回答が無数にある状態では、その後のプログラムの処理が止まってしまうエラーが出ていた。セリフを変更し回答を指定したことで、次の処理に進むプログラムを作成することができるようになった。

(資料6-④)に示した児童の班も、3・4年生の感想からセリフを見直し、より伝わりやすく改善することができた例の一つである。問題解決に適した大きさを考え「抽象化」することができていた。

<資料6-②> 「分解」「抽象化」し改善した制作物の例



また、3・4年生の「答えるとき難しかった」という感想を受けて、どうすれば答えやすくなるか話し合い、セリフの言葉を変更していた(「抽象化」の学習活動)(資料6-③)。

<資料6-③> 「抽象化」し改善した制作物の例



漠然とした質問だと何と答えればよいのかわかりにくかったが、回答を指定することで、回答者にとってより答えやすい質問となった。加えて、回答によって枝分かれ

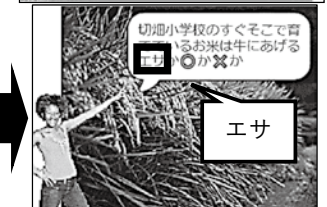
<資料6-④> 「分解」「抽象化」し改善した制作物の例

(3・4年生の感想)

「おもしろかったけど、
字が読めないのもあった。」

(プログラムを改善した児童の振り返り)

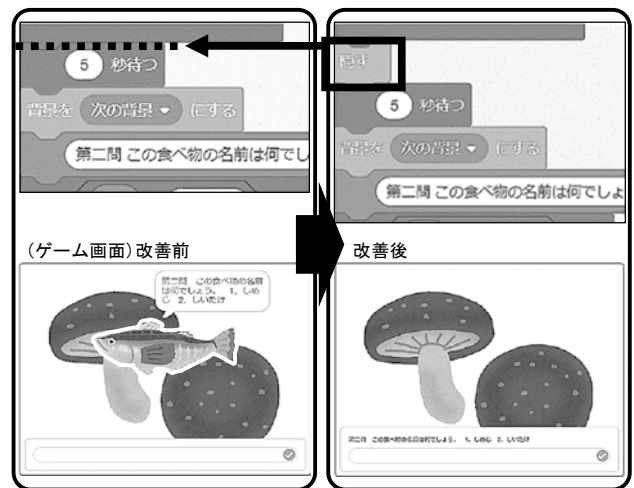
3・4年生が漢字だとわからな
いから、ひらがなに変えたり、
答えるときに正確なのをまちが
っているようになってしまったと
ころを少し変えてわかりやすく
答えられるように班の人で
話し合ってきたのでよか
たです。



<資料6-⑤> 「抽象化」「一般化」し改善した別の班の例

(プログラム)改善前

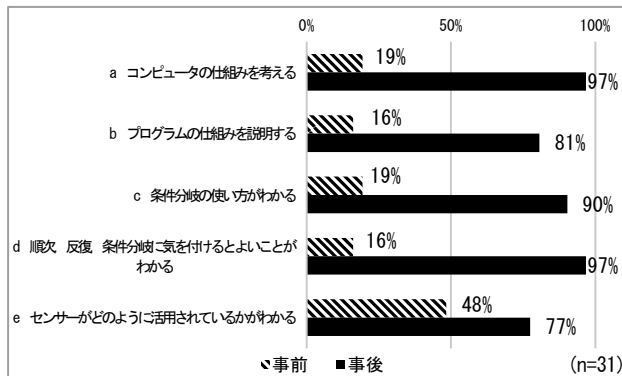
改善後



また、児童は作成したプログラムをその場で実行し、意図した動きになっているかどうかを確認していた。(資料6-⑤)に示した班では、前の場面で表示した魚のキャラクターを隠さないまま、次の場面に進むプログラムを作成していた。より意図したゲームの場面に近づけるために、初め「抽象化」「一般化」したときには選択していなかった、魚のキャラクターを隠すプログラムを追加

ポイントと大幅に増加した(資料13)。

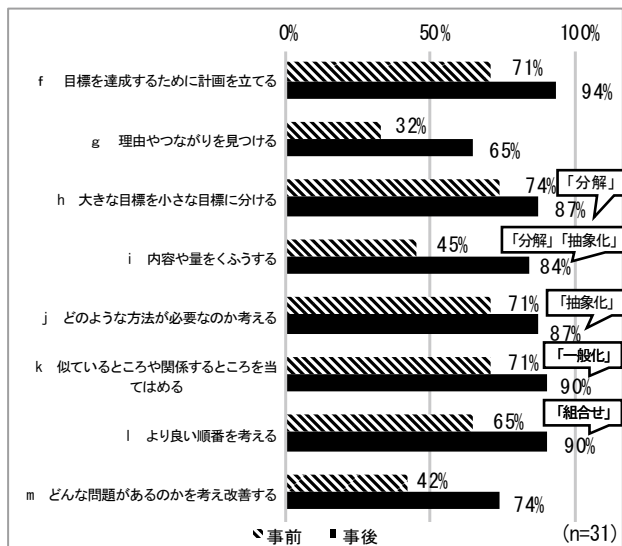
<資料13> 「知識及び技能」に係る設問における肯定的回答の割合



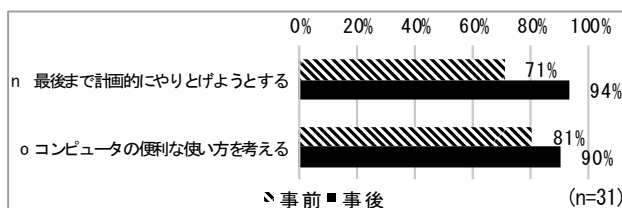
「思考力・判断力・表現力等」に係る設問における肯定的回答の割合では、全項目に伸びが見られた。

「分解」についての設問「h 大きな目標を小さな目標に分ける」は13ポイント、「分解」「抽象化」についての設問「i 内容や量をくふうする」は39ポイント、「抽象化」についての設問「j どのような方法が必要なのか考える」は16ポイント、「一般化」についての設問である「k 似ているところや関係するところを当てはめる」は19ポイント増加した(資料14)。

<資料14> 「思考力・判断力・表現力等」に係る設問における肯定的回答の割合



<資料15> 「学びに向かう力・人間性等」に係る設問における肯定的回答の割合



「学びに向かう力・人間性等」に係る設問では、「n 最後まで計画的にやりとげようとする」についての肯定的回答が23ポイント増加した(資料15)。

V 考察

1 検証授業から

児童は流れ図を作成する場面において、ゲームにして伝えたい内容を班で出し合い、意見を整理する学習活動に取り組んだ。ワークシートや振り返りの記述から、3・4年生により良く伝えるために必要な内容や構成を話し合う中で、プログラミング的思考が高まっていたと考えられる。

Scratchでのプログラミングの場面では、構成要素ごとに「分解」「抽象化」「一般化」した命令を「組合せ」しながらプログラミングすることができていた。また、プログラム作成後に、意図したゲームの場面が実現しないときには、流れ図や制作物を自分たちで振り返ったり、3・4年生に感想を参考にしながら、改善することができていた(資料6-②, 6-③, 6-④, 6-⑤)。児童の制作物、振り返りの記述から、セリフや説明文のわかりにくい点、プログラムの不足や選択ミスなどに気付き改善することができていたことがわかる(主として「分解」「抽象化」すること)。自分だけで気が付かなかったことにも、3・4年生の感想によって気が付くことができていた。このことから、振り返りの場面の設定がプログラミング的思考の向上に有効であったと考えられる。特に、自分で振り返るだけでなく、他者の意見を取り入れることも有効であったと判断できる。

一方で、振り返りにはScratchでのプログラミングの場面で「うまくいかなかった」「思い通りにならなかった」という記述が、数名に見られた。このことから、コンピュータ上におけるプログラミングの際、流れ図を具体的なプログラムに置き換えることが、うまくできなかった児童もいたことがわかる。意図したゲームの動きを、Scratchのプログラムに対応した動きに置き換えることを支援するための手立てが必要であったと考えられる。

2 事前調査・事後調査の比較から

(1) 「プログラミング的思考」に関するテストについて

「プログラミング的思考」に関するテストの結果から、プログラミング的思考の向上がうかがえる。それは、問1の、正答率の向上や最も効率の良い手順を解答していた児童の割合の増加(資料7)から推察できる。また、問2の結果で示したエラーとなる解答の減少や命令数の増加(資料8)からもプログラミング的思考が向上している

と判断できる。問3の結果(資料11)からは、全体のストーリーや他のスライドとの関係性を捉え、適切なスライドを解答できるようになったことがわかる。作成されたプログラムを振り返り、改善する力の伸びがうかがえる。

(2) 「プログラミング的思考」に関する自己評価アンケートについて

「プログラミング的思考」に関する自己評価アンケートにおける「知識及び技能」に係る設問の結果(資料13)から、全項目で大きく伸びが見られた。このことから、児童はこれまでコンピュータやプログラミングについて学習する機会が少なかったことがわかる。今後、プログラミングの学習機会を、発達段階に応じて計画的に学習できるよう位置付けていく必要がある。

また、プログラミング的思考にあたる「思考力・判断力・表現力等」の結果(資料14)より、全項目において伸びが見られ、プログラミング的思考が向上していることがうかがえる。「分解」「抽象化」「一般化」を中心とした、プログラミング的思考を働かせることを想定した学習活動を設定したことが、有効であったと推察する。

また、児童は、ゲームにして伝えたい内容を整理するときや流れ図を具体的なプログラムに置き換えるとき、要素同士の関係性や全体の構成について考えることができていた。このことが、「理由やつながりを見つける」ことができる児童の割合が大きく伸びたことにつながったと考えられる。また、振り返りを繰り返しながらプログラミングを進め、より効率のよいプログラムを作成できるようになったことが、「どんな問題があるのかを考え改善する」ことの伸びにつながったと考えられる。加えて、試行錯誤しながら継続的に改善に取り組むことができたことから、「学びに向かう力・人間性等」における「最後まで計画的にやりとげようとする」ことについての自己評価も高まったと推察する。

VI 研究のまとめ

1 成果

本研究では、作成した授業モデルを実施したことで、児童のプログラミング的思考が向上することが示された。授業モデルにおいて、3・4年生にわかりやすく伝えるというゴールを意識しながら、内容の整理をし、流れ図やプログラムを作成する活動や、作成したプログラムを繰り返し振り返って改善する活動を設定した。

流れ図を作成する場面では、主に「分解」「抽象化」する学習活動としてアイデアを出し合い、まとめたり

必要なものを選んだりする活動を行った。プログラムを作成する場面では、「分解」「抽象化」「一般化」「組合せ」の学習活動として、意図したゲームの場面をどのように実現すればよいかを、話し合いながら具体的なプログラムを作成していった。また、振り返って改善する場面では、「分解」「抽象化」「一般化」の学習活動として、作成したプログラムが3・4年生にわかりやすく伝わるものかどうか考え、必要に応じて改善する活動を行った。

それぞれの場面において、以上のような「分解」「抽象化」「一般化」を中心とした学習活動を意識的に取り入れた授業モデルが、プログラミング的思考の育成に有効であることがわかった。

2 課題

本研究では、プログラミング的思考を4つの要素でとらえ、授業モデルを作成した。しかし、それぞれの明確な分類が難しく、テストの調査結果や設定した学習活動においてあいまいな部分が残ってしまった。今後、研究を続けていきたい。

また、制作物が未完成となってしまった班が多く、授業モデルにおいて想定していた時間が大きく不足していたことがわかった。振り返り改善するための時間を、しっかりと確保し、計画・実行・評価・修正の小さなサイクルを、より多く繰り返せるように授業モデルを見直す必要がある。

本研究の対象児童は、プログラミングの学習の経験が少なく、機器やソフトの扱いにも練習が必要な状態であった。児童のコンピュータへの不慣れさを補う指導を充実させる必要がある。朝活動などの短い時間の活用や、他教科との連携が考えられる。

文部科学省は、プログラミング的思考を働かせるイメージにおいて、プログラミング的思考は発達の段階ごとに「繰り返し学習することで高次に育つ」としている。学校全体で計画的に児童のプログラミング的思考を育成していけるように、所属校において意識の共有を図り、本研究での成果を還元していきたい。

<引用文献・参考文献>

- (注1)文部科学省 小学校学習指導要領解説総則編 第3章第3の1の(3) (平成29年7月)
- (注2)文部科学省 小学校プログラミング教育の手引き(第三版) p.13 (令和2年2月)
 - ・ 文部科学省 小学校プログラミング教育の手引き(第三版)「図4 プログラミング的思考を働かせるイメージ」 p.16 (令和2年2月)
 - ・ 株式会社ベネッセコーポレーション「プログラミング教育」(2018) <https://benesse.jp/programming/beneprog/> (最終確認2022年12月9日)
 - ・ 黒田昌克・福井昌則・掛川淳一・森山潤「アジャイル開発型協働学習を取り入れた小学校プログラミング教育の実践による児童の創造的態度の変容」(2020年2月)