

# 論理的思考力を育む プログラミングの体験の在り方に関する研究

## ー小学校算数科・理科の指導を通してー

### 【研究の概要】

新学習指導要領から導入される小学校プログラミング教育について、4年生の算数科と6年生の理科の授業実践を通して教科への導入の仕方を考察する。プログラミング的思考のとらえ方を「コンピューテーショナル・シンキング」の観点から見直し、使用教材の選定、児童の論理的思考力の育成の見とり方などを工夫し、指導案を作成し、今後のプログラミング教育の一助となるような方向性を示したい。

キーワード：プログラミング教育 プログラミング的思考 コンピューテーショナル・シンキング

平成 30 年 3 月  
岩手県総合教育センター  
長期研修生  
所属校 盛岡市立向中野小学校  
及 川 良 紀

## 目次

I	研究主題	1
II	主題設定の理由	1
III	研究の目的	1
IV	研究の目標	1
V	研究の見通し	2
VI	研究構想	2
1	研究についての基本的な考え方	2
(1)	諸外国のプログラミング教育の実態	2
(2)	国の動向から	4
(3)	本研究におけるプログラミング教育及びプログラミング的思考の定義	4
2	研究の見通しで示した方法（手立て）	6
(1)	意識調査	6
(2)	プログラミング的思考の具体化	8
(3)	プログラミングの体験の配置	8
(4)	コンピュータサイエンスアンプラグドの活動	10
(5)	プログラミング教材の選定	10
(6)	使用機器の選定	12
(7)	学習形態の工夫	12
3	検証計画	12
(1)	事前・事後アンケートの活用	12
(2)	授業記録，ワークシートの活用	12
(3)	授業後の振り返り	13
4	研究構想図	14
VII	授業実践と授業の考察	15
1	授業実践計画	15
(1)	授業実践①	15
(2)	授業実践②	15
2	授業実践の実際と考察	15
(1)	授業実践①（4年生 算数「垂直・平行と四角形」）	15
(2)	授業実践②（6年生 理科「電気とわたしたちの暮らし」）	18
(3)	授業の検証	20
VIII	研究のまとめ	35
1	全体考察	35
2	研究の成果	36
3	今後の課題	36
	<おわりに>	36
IX	引用文献，参考文献，参考 Web ページ	36

## I 研究主題

論理的思考力を育むプログラミングの体験の在り方に関する研究  
—小学校算数科・理科の指導を通して—

## II 主題設定の理由

平成 29 年 3 月に示された小学校学習指導要領総則には、主体的・対話的で深い学びの実現に向けた授業改善の手段の一つとして、情報活用能力の育成を図るため、「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計画的に実施すること」とある。これらのことは、「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）」（以下「議論の取りまとめ」）の「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力としての『プログラミング的思考』などを育むプログラミング教育の実施を子供たちの生活や教科等の学習と関連付けつつ、発達段階に応じて位置付けていくことが求められている」を受けてのものである。

既にプログラミングの体験を導入した研究校の実践では、「学習意欲が高まった」「課題に対して粘り強く取り組む力がついた」「協働して課題解決に向かう場面が増加した」などの成果があげられる一方で、「教科のねらいを達成するためにプログラミングの体験が適切であったか」等、指導者自身が悩みながら指導しているという課題も報告されている。これは教科や単元の目標とプログラミングの体験の目標が合致していないため有効に活用することができなかつたことが原因と考える。本県では、プログラミングの体験に関する研究や実践例は少なく、学校現場においても、「プログラミング言語を使った難しい指導を行わなければならないのではないか」「特別な専門の知識が必要なのではないか」等の不安を抱えている教員の声も聞かれる。また、文部科学省や民間企業などが提案しているプログラミング教材も様々公開されているが、教科での活用事例や授業実践の資料が少ないのが現状である。

小学校段階でプログラミングを体験させるねらいは、プログラミングスキルの習得ではなく、体験を通して論理的思考力を身に付けさせることであり、新学習指導要領では算数科・理科・総合的な学習の時間で取り扱うことが明記されている。教科で取り扱う場合、教科や単元、1 単位時間の目標やねらいに則し、プログラミングの体験を行う必要がある。そのためには、プログラミングの体験の導入の仕方を追究し、教科との関連性を重視した学習展開例の提示が必要と考え、本研究主題を設定した。

## III 研究の目的

小学校算数科・理科の学習指導における、論理的思考力を育むプログラミングの体験の在り方について明らかにする。

## IV 研究の目標

小学校算数科・理科の学習指導において、児童が論理的思考力を用いて課題解決に取り組めるように、プログラミングの体験を導入した学習展開例を作成し、実証的な授業実践を通してその在り方を明らかにする。

## V 研究の見通し

児童が論理的思考力を用いて課題解決に取り組むために、以下の5点について取り組むことにより、小学校算数科・理科の学習指導におけるプログラミングの体験の位置付けを明らかにする。

- ・所属校の教員への意識調査を行う。実践する上での問題点を把握し、プログラミングの体験を導入した学習展開例を作成する際の参考にする。
- ・教科や単元で身に付けさせたい力、プログラミングの体験で身に付けさせたい力を精査し、焦点化する。それを受け、児童が論理的思考力を用いて課題解決に取り組むためにどのようなプログラミングの体験がよいかを考案する。その際にプログラミングの体験が活かされていたかを見とるための判断基準についても検討する。
- ・教科・単元の目標に基づいたプログラミングの体験を取り入れた学習展開例を作成し、授業実践を行う。
- ・授業実践後、授業中の発言やワークシートへの記述、話合いの様子等を分析し、児童が論理的思考力を用いて課題解決に取り組めたかを検証する。
- ・授業に適したプログラミング教材やその使い方についての資料を作成する。

## VI 研究構想

### 1 研究についての基本的な考え方

#### (1) 諸外国のプログラミング教育の実態

諸外国では、日本に先立ってプログラミング教育が行われており、それぞれの国の教育課程に様々な形で位置付けられている。平成26年度の文部科学省による調査では23の国が対象となっているが、ここではプログラミング教育の先進国であるイギリスと、アジア圏内でコンピュータ関連分野において急速に力をつけている韓国を例にして比較したものを【表1】に示す。

どちらの国も、早い段階からプログラミング教育を取り入れているが、時代の要請もあり、ICTリテラシー偏重の時期があることや、最近になりプログラミング教育が見直されてきたという共通点がみられる。これはコンピュータの仕組みや概念がある程度確立され、開発されたソフトウェアを使って学ぶことから、開発の立場で学ぶというように、時代の流れや要請を受け、視点が変更されてきたものと考えられる。

日本では教科として「情報」が高等学校の課程に新設されたのが2003年であるが、その頃に既にイギリスでは初等教育で「IT（情報技術）」を学んでおり、韓国ではプログラミング教育の見直しが行われていたことになる。この点から見れば、日本はプログラミング教育の分野では諸外国に後れを取っていると言わざるを得ない。

韓国では国や企業が強力に先導する形で取り組まれており、ソフトウェア教育を強化する方針を打ち出す様子から、人材育成の傾向が強く表れているようにも見える。

プログラミング教育を実践する目的はその国の歴史や文化、教育制度等の社会的背景が大きく影響してくるため、論理的思考力育成の重視、人材育成の重視等、どの方針がよいかは一概に決められない。しかし、日本のプログラミング教育は後発であるがゆえにこれらの先進国の成果や課題を参考にすることができるという利点がある。

【表1】 諸外国（イギリス・韓国）のプログラミング教育実施状況

	イギリス	韓国
教育制度	<ul style="list-style-type: none"> <li>・初等教育（5～10歳までの6年間をプライマリースクール、前期2年間、後期4年間をそれぞれKey Stage1,2とする）</li> <li>・中等教育（11～17歳までの7年間をセカンダリースクール、前期3年間、中期2年間をそれぞれKey Stage3,4、後期2年間をシックスフォームとする）</li> <li>・高等教育（18～20歳までの3年間を大学とする）</li> </ul>	<ul style="list-style-type: none"> <li>・初等教育（6～11歳までの6年間）</li> <li>・中等教育（12～17歳までの6年間、前期3年間を中学校、後期3年間を高等学校とする）</li> <li>・高等教育（18～21歳までの4年間を大学とする）</li> </ul>
実施科目名	<ul style="list-style-type: none"> <li>・「Computing」（必修）</li> </ul>	<ul style="list-style-type: none"> <li>・初等教育の後期2年間「実科」（必修）</li> <li>・中等教育は「技術・家庭」（必修）、「情報」（選択）</li> <li>・高等教育は「情報」（深化選択）</li> </ul>
内容	<ul style="list-style-type: none"> <li>・「プログラミング教育」含む</li> </ul>	<ul style="list-style-type: none"> <li>・初等教育「実科」では「ICTリテラシー教育」中心</li> <li>・中等教育は「技術・家庭」で「ICTリテラシー教育」中心、「情報」で「プログラミング教育」を含む</li> <li>・高等学校は「深化選択」で「プログラミング教育」</li> </ul>
担任	<ul style="list-style-type: none"> <li>・初等教育は学級担任制，中学校以降は教科担任制</li> </ul>	
その他	<ul style="list-style-type: none"> <li>・「コンピューショナル・シンキング」を目標に掲げる。</li> <li>・カリキュラムの改定毎に「IT」「ICT」「Computing」と教科名が変わり、コンピュータサイエンスの内容を充実したのになっている。「IT」は1995年、「Computing」は2014年9月から導入。</li> <li>・「Computing」はCS(Computer Science), IT(Information Technology), DL(Digital Literacy)の3分野構成でプログラミングは主にCSで扱われる。</li> <li>・国により定められた教科書，ツール，キットはない。教材費用は各学校が負担。</li> <li>・グループワークが多くなるため，自己評価や相互評価の他にKWL表<sup>[1]</sup>を活用している。</li> <li>・課題は教員の知識・スキル不足，指導者・指導時間不足。</li> </ul>	<ul style="list-style-type: none"> <li>・1987年「学校コンピュータ教育強化法案」より初等・中学校へ情報教育導入。</li> <li>・1997年第7次教育課程改定でICTリテラシーの習得に目標が置かれプログラミングの内容が削除。</li> <li>・ICTリテラシー偏重への反省からプログラミング教育が再び重要視され，2007年の改定から復活。</li> <li>・教科「情報」の開講率は中学28%高等学校47%だが，受講率はそれぞれ8.1%，5.2%と低い（2012年）。</li> <li>・教科書がありコンピュータサイエンスの基礎的な内容を扱っている。</li> <li>・2017年から小学校の正規教育課程としてソフトウェア(SW)を教えることになる。</li> <li>・発達段階における体系的SW教育の不足，SW英才の早期発掘や育成支援体制の不足，大学入試に「情報教育」科目がない等の問題点。</li> </ul>

[1]教材を効果的に読ませるために考案された教育の手法。K列，W列，L列の3列からなる表の枠を作り，それぞれに知っていることは何か(What I Know)，知りたいことは何か(What I want to Know)知ったことは何か(What I learned)を書き込ませる。予習としてKとWを列挙して目的や興味を喚起し，復習としてLを列挙することで達成項目を明確にする。K, W, Lのうちいずれかが抜れたり，別の列が加わったりと，利用のしかたはさまざまである。

## (2) 国の動向から

平成 26 年 6 月に改定された「世界最先端 IT 国家創造宣言」において、初等・中等教育段階におけるプログラミング教育の重要性が示され、総務省においてもプログラミングに関する人材育成の重要性や論理的思考力の向上の可能性について言及されてきた。また、青少年の発達段階に応じたプログラミングに関する教育を通じて、将来の高度 ICT 人材としての素地の構築・資質の発掘を図るべくプログラミング人材育成の取組みが行われた。

文部科学省で協議された「議論の取りまとめ」の中には、人工知能の発達による「第 4 次産業革命<sup>[2]</sup>」や「IoT 社会<sup>[3]</sup>」等の影響を受け、今後、社会の在り方が大きく変化していくという予測が提言されている。また、今後 10～20 年程度で、半数近くの仕事が自動化される可能性や、子供たちの 65%は将来、現在存在していない職業に就く可能性、人工知能が新たな職業を生み出す可能性なども指摘されている。このような将来の予測が困難な時代の中で、プログラミング教育の必要性がうたわれた背景には「自動販売機やロボット掃除機など、身近な生活の中でもコンピュータとプログラミングの働きの恩恵を受けており、これらの便利な機械が「魔法の箱」ではなく、プログラミングを通じて人間の意図した処理を行わせることができるものであることを理解することは、時代の要請として受け止めていく必要がある」という一文からも読み取ることができる。これからの時代を生きていく子供たちには、コンピュータやプログラミングに対し、受け身ではなくむしろ積極的に関わりをもつことが大切であり、それによって育まれる「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力としてのプログラミング的思考」が必要とされている。

つまり、プログラミングを学ぶことは、時代のニーズに沿った必然的な学習であり、そこで育まれるプログラミング的思考は、様々な職種を越え、全ての子供たちに必要な力であると考えられる。

## (3) 本研究におけるプログラミング教育及びプログラミング的思考の定義

「プログラミング教育」と「プログラミング的思考」は本研究の中心を貫く重要な語句であるため、学習指導要領改訂に際して行われた「議論の取りまとめ」から 2 つの語句の意味するところを取り上げて記述する。

また、プログラミング的思考をより具体的にするために、考え方の基となった「コンピューショナル・シンキング」について考察する。

「議論の取りまとめ」によると「プログラミング教育」と「プログラミング的思考」については次のようにまとめられている。

### ア プログラミング教育について

#### 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に指示することができるということ**を体験させながら、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力としての「**プログラミング的思考**」などを育成するもの。（「議論の取りまとめ」より）

[2]Industry4.0。ドイツの産官共同プロジェクトが提唱。新たな産業高度化の概念。第 1 次…蒸気機関、第 2 次…電気機関、第 3 次…製造業の自由化、第 4 次…インターネットを通じてあらゆる機関が結びつく段階。

[3]Internet of Things（モノのインターネット）。自動車や家電など身の回りのあらゆるものがネットにつながるという考え方。機械の遠隔操作などができるようになる。



小学校におけるプログラミング教育が目指すのは、子供たちが、コンピュータに意図した処理を行うように指示する体験をしながら、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、各教科等で育まれる思考力を基盤としながら基本的な「プログラミング的思考」を身に付けること、コンピュータの働きを自分の生活に活かそうとする態度を身に付けることである。

## イ プログラミング的思考について

### プログラミング的思考とは

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**。  
(「議論の取りまとめ」より)

「プログラミング的思考」には、各教科等で育まれる論理的・創造的な思考力が大きく関係している。各教科で育む思考力を基盤としながら「プログラミング的思考」が生まれ、「プログラミング的思考」の育成により各教科等における思考の論理性も明確となっていくという関係を考え、アナログ感覚を大事にしていくことの重要性も踏まえながら、教育課程全体での位置付けを考えていく必要がある。

また、「議論の取りまとめ」には、プログラミング的思考は「コンピューターショナル・シンキング」の考え方を踏まえつつ、プログラミングと論理的思考との関係を整理しながら提言された定義であると記載されている。「コンピューターショナル・シンキング」とはウィング(2006)の「Computational Thinking」で定義されたものである。これによると「ある問題について、コンピュータによって効果的に実行されうる形式でその問題や解法を記述するときに関わってくる思考プロセス」であり、論理的思考やシステム思考と重なるものとされている。日本語訳は中島(2015)により「計算論的思考」として翻訳されているが、定訳にはなっていない。

「コンピューターショナル・シンキング」を教育の分野で取り入れているのがイギリスで、教科「コンピューティング」の目的や目標の中に示されている。

### 【資料1】イギリスの教科「コンピューティング」の目的及び目標

目的 プログラム、システム、コンテンツ制作というプログラミングの活動を通して、他教科に活用できる思考力(コンピューターショナル・シンキング)、創造性、洞察力、情報活用能力の育成を図る。

目標 プログラム作成等の実践的経験を通して、「コンピューターショナル・シンキング」を理解し、活用できるようにする。

さらに、「コンピューティング」で学ぶべき学習内容を段階的に示した「Computing Progression Pathways」の中では具体的な到達内容が記されており、それぞれ「抽象化」・「分解」・「アルゴリズム的思考」・「評価」・「一般化」の5つの能力に分類、対応させている。

【表2】 コンピューテーショナル・シンキングで育成する5つの能力

能力	概要
抽象化 (Abstraction) [AB]	問題を抽象化して理解する能力
分解 (Decomposition) [DE]	物事を分解して理解する能力
アルゴリズム的思考 (Algorithmic Thinking) [AL]	やるべきことを順序立てて考える能力
評価 (Evaluation) [EV]	最良の方法かどうかを評価・分析する能力
一般化 (Generalization) [GE]	方法を他に置き換えて一般化する能力

安藤 (2017) はこの5つの能力とプログラミング的思考の関連性を次のように述べている。

ウ プログラミング的思考とコンピューテーショナル・シンキングの対応

**プログラミング的思考とは**

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。

抽象化
分解

(符号化)
アルゴリズム的思考・一般化

(修正)
評価

プログラミング的思考=コンピューテーショナル・シンキングとはならないが、プログラミング的思考の定義はコンピューテーショナル・シンキングに包含されていると捉えることができる。言い換えると、コンピューテーショナル・シンキングは問題解決能力やコミュニケーション能力なども含んだ範囲の広い考え方であるのに対し、プログラミング的思考はその中から限定的な部分を抽出した考え方であるともいえる。

そこで、本研究ではこの5つの能力をプログラミング的思考で育成すべき能力とし、教科(算数科・理科)の中でどのように関係づけられ、育成していけばよいかを明確にしていくものとする。

## 2 研究の見通しで示した方法(手立て)

### (1) 教員の意識調査

所属校において教員の意識調査を実施した。プログラミングの体験についての認知度や21世紀型能力<sup>[4]</sup>に照らし合わせて、児童に身に付けさせたい資質・能力について調査した。

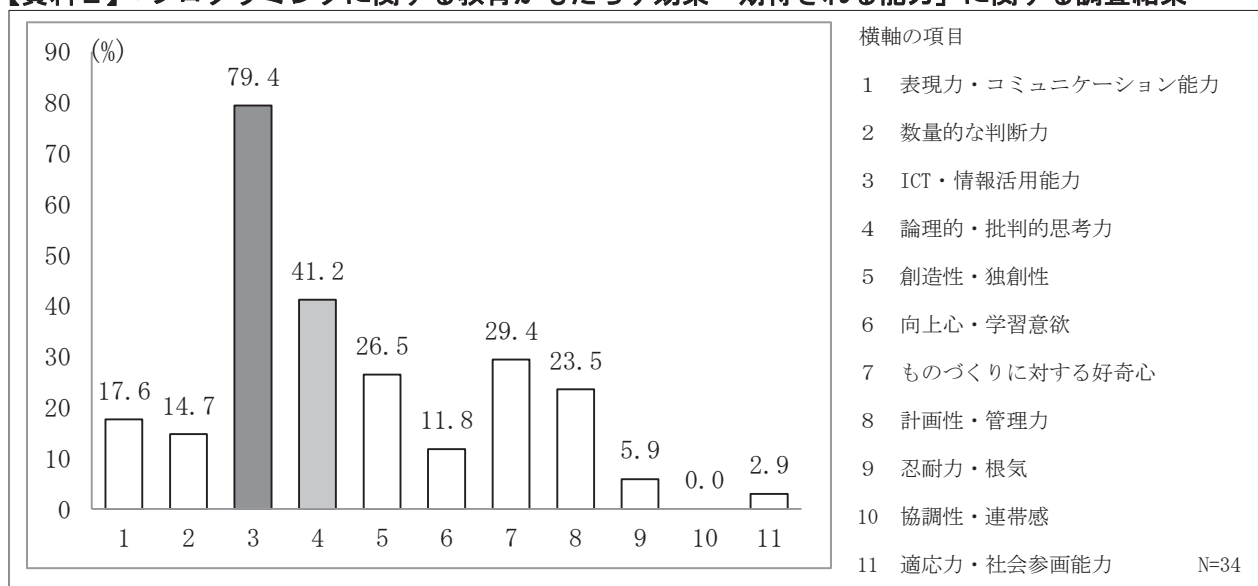
また、何に対して苦手意識をもっているのか、実践を行うにあたり喫緊の課題を調査結果から

[4] 国立教育政策研究所が提案した「生きる力」としての知・徳・体を構成する資質・能力から、教科・領域横断的に学習することが求められる能力を資質・能力として抽出し、これまで日本の学校教育が培ってきた資質・能力を踏まえつつ、それらを「基礎」「思考」「実践」の観点で再構成した日本型資質・能力の枠組み。「基礎力」には「言語スキル」「数量スキル」「情報スキル」が、「思考力」には「問題解決・発見・創造力」「論理的・批判的思考力」「メタ認知・適応的学習力」が、「実践力」には「自律的活動力」「人間関係形成力」「社会参画力」「持続可能な未来への責任」がそれぞれ位置付けられている。



明らかにし、授業実践の構想に反映させる。以下に調査結果の一部を記載する。

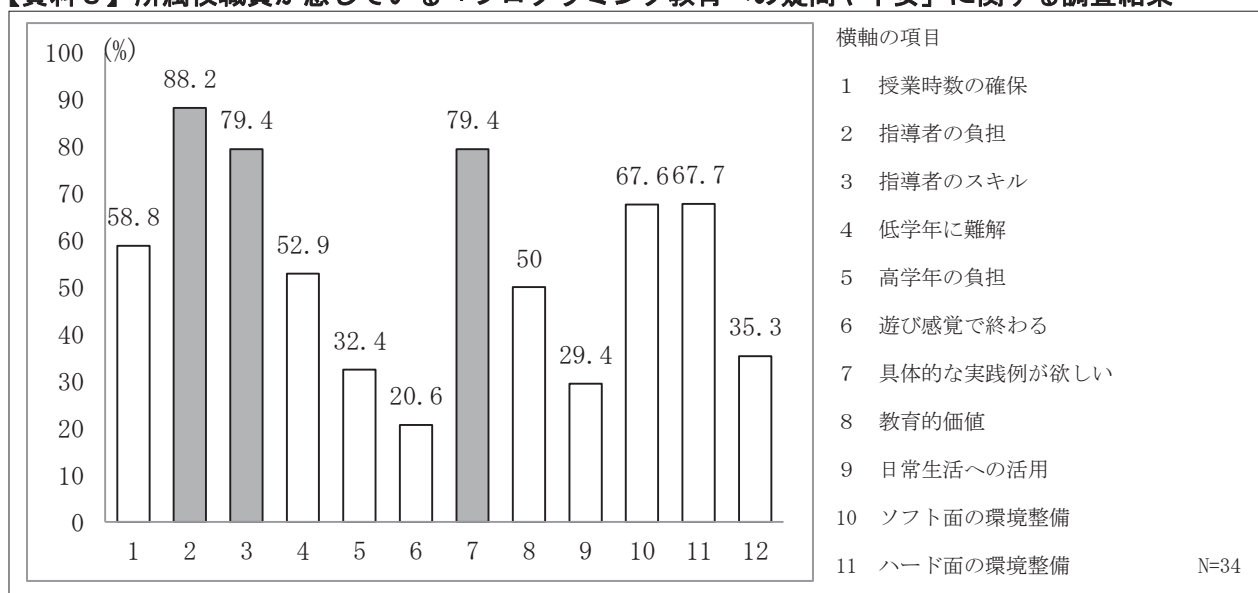
**【資料2】「プログラミングに関する教育がもたらす効果・期待される能力」に関する調査結果**



【資料2】は「プログラミング人材育成のあり方に関する調査研究（総務省）」の調査を参考に所属校で行った調査結果である。調査項目は21世紀型能力に照らし合わせたものである。

「3 情報活用能力」の項目は約80%と大きな値を示し、本研究で育成したい「4 論理的思考力」の項目は約40%となった。新学習指導要領が示されたとはいえ、所属校においては、調査を実施した6月初旬の段階では、プログラミング教育で何を育成するのか、何を指すのかといった部分の認識はまだ低い現状であることがうかがえる。このような実態を踏まえ、本研究ではプログラミング教育がねらうところの論理的思考力の育成を目指した授業実践例を提示することとする。

**【資料3】所属校職員が感じている「プログラミング教育への疑問や不安」に関する調査結果**



【資料3】からは「指導者の負担」、「指導者のスキル」、「具体的な実践例の必要性」の項目の値が高く表れた。これまで経験したことのない分野への取組みに対しての不安感が表れている。具体的

な実践例を提示することは、こうした不安感を取り除くことにもつながると思われる。より具体的に、授業の様子がイメージしやすい実践例の作成を心がけて提案することで、プログラミング教育に対する不安感を取り払い、教員の意識面のハードルを下げることができると思われる。そのためには、手軽に取り組めるものであること、授業のイメージがつかみやすいものであることなど、指導のポイントを明確にした実践例を提案する必要がある。

(2) プログラミング的思考の具体化

プログラミング的思考で育成する5つの能力について、教科における具体的な活動を【表3】に示す。さらにこの活動を算数科では「図形（平行四辺形）の作図」、理科では「電気の制御」に当てはめ、検証していくこととする。

**【表3】プログラミング的思考の具体的な活動**

能力	概要	具体的な活動
抽象化 (Abstraction) [AB]	問題を抽象化して理解する能力。	目的に合わせて対象から注目すべき要素を重点的に抜き出して問題を単純化する。
分解 (Decomposition) [DE]	物事を分解して理解する能力。	問題を理解・解決できるようにいくつかの部分に分解する。
アルゴリズム的思考 (Algorithmic Thinking) [AL]	やるべきことを順序立てて考える能力。	分解した要素を手順を考えながら構築する。
評価 (Evaluation) [EV]	最良の方法かどうかを評価・分析する能力。	間違いを修正したり、他の方法やより簡単な方法がないかを考えたりしながら、必要な修正を加え、確認する。
一般化 (Generalization) [GE]	方法を他に置き換えて一般化する能力。	類似性のある問題に置き換えて考える。

(3) プログラミングの体験の配置

プログラミングの体験を指導計画に配置するパターンをいくつか提示し、その効果やメリット、デメリットを考察する。

例えば単元構成が10時間、平成32年度から実施の学習指導要領に示されている算数科の「図形の作図」や理科の「電気エネルギー」の実験等を行うことができる時数が第6時に設定する場合で以下検討する。

教科や単元の構成にもよるが、単元内のどのタイミングでプログラミングの体験をさせると効果的であるのかをいくつかのパターンで実践し、検証する。

**パターン① 通常の授業展開の場合**

時数	1	2	3	4	5	6	7	8	9	10
内容						作図・実験				

**パターン② プログラミングの体験を単元の冒頭に配置する場合（プログラミングの体験に関する時数を便宜上0とする）**

時数	0	0	1	2	3	4	5	6	0	0	7	8	9	10
内容	プログラミング 教材の学習							作図・ 実験	教材を使 っての作 図・実験					

**パターン③ プログラミングの体験を途中で配置する場合（第6時と第0時は交換可能）**

時数	1	2	3	4	5	0	0	6	0	0	7	8	9	10
内容						プログラミング 教材の学習		作図・ 実験	教材を使 っての作 図・実験					

**パターン④ プログラミングの体験を終末に配置する場合**

時数	1	2	3	4	5	6	7	8	9	10	0	0	0	0
内容						作図・ 実験					プログラミング 教材の学習		教材を使っ ての作図・ 実験	

**【図1】プログラミングの体験の配置**

**【表4】それぞれの授業展開における予想されるメリット・デメリット**

パターン	メリット	デメリット
②	・プログラミング教材に触れてから学習することで、物事の手順や順序の大切さの意識を持続しながら臨むことができる。	・教材の学習から本時までの時数が空いてしまう。
③	・プログラミング教材の基本操作直後の指導なので興味・関心を持続することができる。 ・プログラミングスキルが高い状態で授業に臨むことができる。	・単元の流れが中断される。
④	・授業の流れを妨げることなく実践できる。 ・発展的な学習が望める。	・単元の終末までプログラミングの体験に触れることができないのでプログラミング的思考を学習に活かすにくい。
共通	・プログラミングの体験を行うことでプログラミング的思考を意識した学習展開ができる。	・いずれの展開でも導入初期の段階では、プログラミング教材の学習に充てる時数が発生してしまう。

(4) コンピュータサイエンスアンブラグド<sup>[5]</sup>の活動

マウスやキーボードなどの基本的なコンピュータスキルでも、児童は習熟度に大きな差があり、操作が苦手な児童にとってはハードルになりかねず、プログラミングを敬遠する要因ともなりうる可能性がある。プログラミングに初めて触れる今の段階では、期待と同時に不安を感じる児童も少なくない。

そこで、コンピュータを使わずに、体験を通してプログラミングの基礎を学ぶことができるよう、コンピュータサイエンスアンブラグドの活動を導入の段階で取り入れる。



それぞれの学年の発達段階や教科の特性を考慮し、4年生では「ルビィのぼうけん<sup>[6]</sup>」から歯磨きの手順やケーキを並べる際の手順など、日常生活を題材として「順次」や「修正」の考え方を、6年生では日常生活をフローチャート図<sup>[7]</sup>に表し、「順次」「分岐」「反復」というプログラミングの基本処理を学習する時間をそれぞれ配置する。このことにより、実際にコンピュータを操作するときに応用することができたり、学習活動が停止した際に振り返って考えさせたりすることにもつながる。全ての児童がプログラミングに興味をもって取り組めるように配慮する。

(5) プログラミング教材の選定

現在、教育に用いられているプログラミング教材は、コンピュータを使わないものからロボットに至るまで多岐にわたり、小学校で扱えるものも増えてきている。

今回の授業実践を行うにあたり、教材選定の際に検討した教材の一部を紹介する。直接教科に活用できるものの他に、発達段階に応じたもの、プログラミングの導入に有効なものなども含めて【表5】に表す。





【表5】プログラミング教材一覧



カテゴリー	教材名	特徴
アンブラグド	ルビィのぼうけん <a href="http://www.shoehisha.co.jp/book/rubynobouken/">http://www.shoehisha.co.jp/book/rubynobouken/</a>	リンダ・リウカス氏が創作した絵本「ルビィのぼうけん こんにちは！プログラミング」（翔泳社刊：鳥井雪 訳）。主人公のルビィが宝石集めを通してプログラムに必要な考え方に触れていく物語。コンピュータを使わずにプログラミング的思考の基本的な考え方を紹介している。 
ビジュアルプログラミング言語	Viscuit（ビスケット） <a href="https://www.viscuit.com/">https://www.viscuit.com/</a>	2003年に「誰でもプログラミングを体験してコンピュータの本質が理解できる」をコンセプトとして開発されたビジュアルプログラミング言語。自分で描いた絵がプログラムの部品となり、動かすことができる。 

[5]ニュージーランドで開発された学習コンテンツ。コンピュータを利用せずに、カードや筆記用具などを用いたゲームやパズルに取り組むことを通してコンピュータの仕組みや概念を学ぶことができる。

[6]リンダ・リウカス氏が創作した絵本。アンブラグドの考え方でプログラミング的思考の基本を紹介している。

[7]プロセスの各ステップを箱（ブロック）で表し、流れをそれらの箱の間の矢印で表すことでアルゴリズムやプロセスを表現する図のこと。

<p>ビジュアルプログラミング言語</p>	<p><b>Scratch (スクラッチ)</b>  <a href="https://scratch.mit.edu/">https://scratch.mit.edu/</a></p>	<p>マサチューセッツ工科大学メディアラボのミッチェル・レズニック氏により開発された、文字を書かずにプログラミングができるビジュアルプログラミング言語。</p> <p>2017年1月現在、150カ国以上、1,600万人以上のユーザーがいる。関連書籍も多く、テレビでも取り上げられているためプログラミング教室やワークショップなどで最も多く使用されている。</p> 
<p>プログラミング体験ツール</p>	<p><b>プログラミン</b>  <a href="http://www.mext.go.jp/">http://www.mext.go.jp/</a></p>	<p>文部科学省が提供するウェブサイトでプログラミングを体験できる。スクラッチを参考に作られた。動作や制御を表す「プログラミン」というキャラクターを使って指示することで対象の絵を動かしたり音を鳴らしたりすることができる。</p>
	<p><b>アルゴロジック</b>  <a href="http://home.jeita.or.jp/is/highschool/algo/">http://home.jeita.or.jp/is/highschool/algo/</a></p>	<p>JEITA (一般社団法人電子情報技術産業協会) が提供するアルゴリズム体験ゲームのウェブサイト。ゲームを通して順次処理、分岐処理、反復処理を体験できる。課題解決型のゲームを解くことで論理的思考を習得するのが目的である。</p>  <p>©2010 JEITA</p>
	<p><b>プログル</b>  <a href="https://proguru.jp/">https://proguru.jp/</a></p>	<p>日本の教科単元と密接にリンクさせたプログラミング教材。教科の目的とプログラミング的思考をドリル型で学ぶことができる。現在5年生の算数の「公倍数」の単元と「多角形と円」の単元のコースが提供されている。</p> 
	<p><b>Code Studio (コードスタジオ)</b>  <b>Hour of Code</b>  <a href="http://studio.code.org">http://studio.code.org</a>      ☆算数科(4年生)の授業実践で活用</p>	<p>2013年から米国の非営利活動法人「Code.org」が始めたプログラミングを普及させるための活動で、学習教材が無料で提供されている。現在180カ国以上、数千万人の子供たちが参加しており、米国では授業に取り入れている学校もある。“Code with Anna and Elsa”はスモールステップ、パズル形式のプログラミング教材で図形の作図に特化している。</p> 
<p>デバイス<sup>[8]</sup></p>	<p><b>教育版レゴ®マインドストーム®EV3</b>  <a href="http://education.lego.com/ja-jp/learn/middle-school/Mindstorms-ev3">http://education.lego.com/ja-jp/learn/middle-school/Mindstorms-ev3</a></p>	<p>レゴブロックを使って組み立てたロボットをプログラミングで制御できる教材。ロボット本体となるインテリジェントブロック、センサーやモーター、CPUなどの必要なハードウェアの他、専用ソフトウェアが含まれている。遊びながら学ぶ教材として小中学生向けのスクールや授業でも取り扱われている。</p>

	<p>Studuino (スタディーノ)</p> <p><a href="http://www.artec-kk.co.jp/studuino-2">http://www.artec-kk.co.jp/studuino-2</a></p>	<p>株式会社アーテックが販売するプログラミング教育用の制御基板。MIT メディアラボが開発した Scratch をカスタマイズしたソフトウェアでプログラミングをしそれをベースにロボットを操作することができる。</p>	
	<p>計測制御プログラマー</p> <p><a href="https://www.vstone.co.jp/products/mcprogrammer/">https://www.vstone.co.jp/products/mcprogrammer/</a></p> <p>☆理科（6年生）の授業実践で使用</p>	<p>ヴィストーン株式会社が販売する中学校技術家庭科の「計測と制御」を学習するためのプログラミング教材。本体と PC を USB で接続し、専用ソフトウェア「ビュートビルダーP」でプログラムを作成する。</p>	

#### (6) 使用機器の選定

どの学校でも多様に活用できる実践例を提示するため、本研究ではキーボードでの入力のしやすさ、マウスの操作性を考慮し、デスクトップパソコンを使用する授業を提案する。

#### (7) 学習形態の工夫

一斉指導が基本となるが、児童のコンピュータスキルの習熟度や学級の教科の進め方、発達段階などに応じて、グループ活動やペア学習を取り入れる。グループで考えることにより、協働して問題解決に向かう姿勢や、コンピュータ操作の苦手な児童のハードルを下げることを目的として行う。また一人では解けない問題でも、話合いの中からヒントを得たり、友達に自分の考えを伝えることで操作の手順を意識したりすることができると思う。

### 3 検証計画

論理的思考力を用いて課題解決している児童の様子を把握するために、以下の方法を使って検証する。

#### (1) 事前・事後アンケートの活用

児童の事前・事後の意識調査により、その変容を確認する。

#### (2) 授業記録、ワークシートの活用

プログラミングの体験により論理的思考力を用いて課題解決に取り組んでいるかを把握する。例えば算数では「図形を作図する際に手順を意識したプログラミングができていないか」、「他の図形に活用しようとしているか」、理科では「電気を制御するためにセンサーの働きを利用したプログラミングができていないか」などを児童が操作しているコンピュータ画面の記録やワークシートの記述から読み取る。その際にはプログラミング的思考の5つの能力に対応させ、どの力が用いられていたのかを確認する。データの採取方法については【表6】に記す。

[8]device。語意は「装置」。コンピュータではシステムを構成している独立したハードウェア（周辺機器）のことをいう。キーボードから各種メディアのドライブ、プリンタ、スキャナなどまで多種多様。



**【表6】データの採取方法**

採取方法	検証内容
ワークシート	児童それぞれの考えが筋道立てて記述されているか、コード <sup>[9]</sup> やブロックの並べ方、問題文から要素を抽出できているかなど、論理的思考につながる結果を読み取る。
ビデオカメラ (3台)	授業全体の様子や、教師の発問、児童の反応を記録する。また、児童の活動を個別に撮影することでより詳細な変容を記録する。 ビデオカメラ3台の用途は以下の通り。 ①授業全体（前方から撮影） ②授業全体（後方から撮影） ③個別（任意の児童の活動を撮影）
ボイスレコーダー (5台)	主に抽出した児童の話合いの様子を記録し、発言の中に論理的思考力につながる要素が含まれているかを読み取る。
スクリーンキャプチャ <sup>[10]</sup> (Interval ScreenCapture) <sup>[11]</sup>	児童のコンピュータ上の作業画面を一定間隔で画像として記録しておき、それを連続再生することで児童の思考過程を読み取る。さらにコードの並び方や時間の経過等から論理的な思考につながっているかを判断する。

(3) 授業後の振り返り

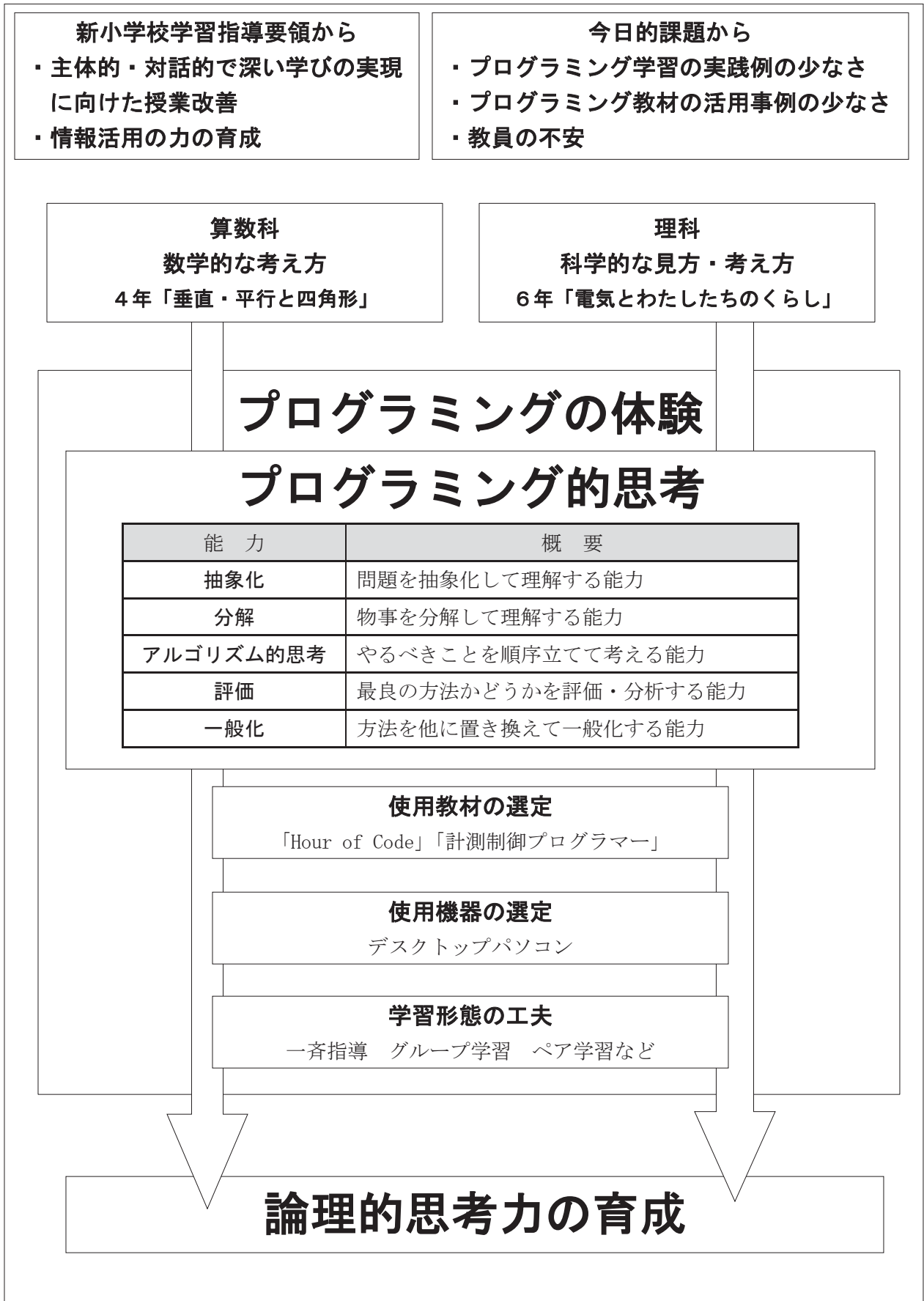
ワークシートの記述や事後アンケートなどの記述から、プログラミングの体験が論理的思考力の活用につながったかどうかを見取る。

[9]ソースコード (source code)。コンピュータプログラミング言語で書かれた文字列のこと。ここでは「前に進む100ピクセル」「右に曲がる90°」など一つの命令文のことを指して呼ぶ。

[10]スクリーンショット。コンピュータのモニタ上に表示された映像のこと。

[11]コンピュータ画面を指定した間隔で画像として自動保存する。画像保存形式は jpg。詳細やダウンロードについては岩手県立総合教育センターのホームページ ([http://www1.iwate-ed.jp/tantou/joho/material/web\\_interval\\_screen\\_capture/index.html](http://www1.iwate-ed.jp/tantou/joho/material/web_interval_screen_capture/index.html))。

4 研究構想図



【図2】研究構想図

## Ⅶ 授業実践と授業の考察

### 1 授業実践計画

#### (1) 授業実践①

実践校 盛岡市立向中野小学校

対象学年 第4学年 1組（男子15名女子17名計32名）

3組（男子15名女子17名計32名）

実施期間 平成29年8月21日（月）～平成29年9月15日（金）

※プログラミングに関する日程は「2 授業実践の実際と考察 (1)授業実践①」の項を参照。

実施内容 算数科「垂直・平行と四角形」（東京書籍 新しい算数 4年 pp.60～83）

※指導案は別添補助資料，ガイドブック参照。

#### (2) 授業実践②

実践校 盛岡市立向中野小学校

対象学年 第6学年 1組（男子16名女子19名計35名）

2組（男子14名女子18名計32名）

3組（男子15名女子20名計35名）

実施期間 平成29年10月16日（月）～10月19日（木）

※プログラミングに関する日程は「2 授業実践の実際と考察 (2)授業実践②」の項を参照。

実施内容 理科 「電気とわたしたちの暮らし」

（東京書籍 新しい理科 6年 pp.166～185）

※指導案は別添補助資料，ガイドブック参照。

### 2 授業実践の実際と考察

#### (1) 授業実践① [4年生 算数「垂直・平行と四角形」]

ア 【導入】プログラミングに関わる導入 [1組:8月23日（水），3組:9月13日（水）]

目標 プログラミング教材を使い，図形の作図の仕方について理解する。
-----------------------------------

#### (ア) コンピュータサイエンスアンプラグドの活動とプログラミング教材の関わり

児童のコンピュータスキルの優劣やプログラミングへの不安感を取り除くために，コンピュータサイエンスアンプラグドの活動を導入に取り入れた。プログラミング教材を使用する際に必要なアルゴリズムの基礎やパソコン画面の座標（方向）感覚を養うため，ゲーム的な要素を取り入れた展開とした。ここで身に付けた技能を，プログラミング教材を使った作図に活かすことができるよう配慮した。

#### (イ) コンピュータサイエンスアンプラグドの活動

アルゴリズムの基礎を身に付けるために，「ルビィのぼうけん」から，歯磨きの仕方やケーキの食べ方の手順についての問題など，日常生活の様子を例にして，物事の手順について考えさせた。また友達に発表する際にも順序を意識させた。さらにコンピュータ画面上

での座標（方向）感覚を養うために、宝探しゲームを行い、宝箱までの道筋を手順を考えながら説明させるとともに、2次元上での方向の表し方について考えさせた。

<p>歯みがきする順番は？ 空いている部分を考えてみましょう。</p> <ol style="list-style-type: none"> <li>洗面所に行く。</li> <li>歯ブラシをもつ。</li> <li>歯みがきをする。</li> <li>みがく。</li> <li>うがいをする。</li> <li>洗面所を出る。</li> </ol>	<p>ケーキを食べる順番は？ ちゃんと食べられるかな？おかしなところはどこでしょう。</p> <ol style="list-style-type: none"> <li>お皿をならべる。</li> <li>フォークをならべる。</li> <li>ケーキをのせる。</li> <li>テーブルクロスをひく。</li> <li>ケーキを食べる。</li> </ol>
---	--

【図3】歯磨き・ケーキを食べる手順（ワークシート）



【図4】宝探しゲームの記入例

【資料4】宝探しゲームの発表の場面（一部抜粋）

1:T（教師。以下T） 今いる場所はここ。宝箱はここ。この人を宝箱まで動かしてもらいたいです。「前に1歩進む」で1つの指示。目の前に石があるから次にどうしたらいい？

2:C（児童。以下C） 右を向く。

3:T そうですね。いろいろな道筋があるから先に矢印を書いてみるといいです。

4:T 左右が分からなくなりますね。紙を動かしてみるといいかも。できたら隣の人に説明してみましょう。  
～指でなぞったり椅子を回したりしながら考える～

5:T ではみんなで確認しましょう。

6:C 右を向く→前に4歩進む→左を向く→…

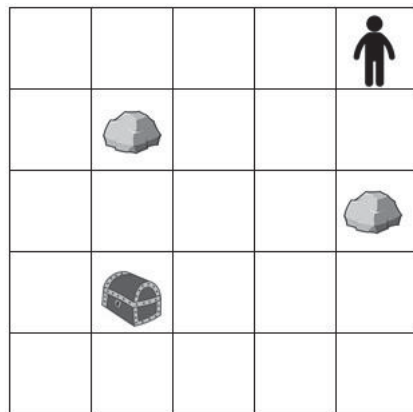
7:T たどり着きましたね。他のやり方はありませんか。

8:C 右を向く→前に2歩進む→前を向く→…

9:T 前を向いたらそのままだよ。

10:C あ、じゃあ左を向く→前に3歩進む→…→宝物をゲットする。

11:T 手順が正しければどの道を通っても宝箱には着きますね。だからお互いのやり方は違っていても正解なんです。次の時間はこういうことを実際にコンピュータを使ってやってみたいと思います。



(ウ) プログラミング教材について

本単元では「Hour of Code “Code with Anna and Elsa”」（みんなのコード）を選択して活用した。

画面上のキャラクターにコードを用いて命令すると、歩く軌跡が辺や角となって表示される。1本の直線を引くことから徐々に複雑な形へステップアップできる構成となっているため、体験を通し、教材の操作や方向感覚を楽しみながら身に付けることができる。



【図5】教材の作業画面

自由に作図できる機能もあるので、体得した技能を使って図形を組み合わせたり、円を作図したりすることもでき、発展的な学習へ展開が可能である。

プログラミングに初めて触れることになる児童には、できるだけ簡略化された教材が望ましい。授業実践では「普及率の高いスクラッチがベースであること」、「図形の作図に特化していること」、「複数の回答が許容されていること」などの理由から発達段階を考慮した上で選択した。

(エ) プログラミング教材を用いた活動

「Hour of Code “Code with Anna and Elsa”」を使用し、コンピュータで図形を作図するための操作の基本を学ばせた。コードの並べ方や数値の入力の仕方など、基本的な操作を理解することにより、率先して取り組んでいた。段階が進むと難易度も高くなるが、より難しい問題に挑戦しようと真剣に取り組んでいた。習得には同じ操作を繰り返すことも必要であるため、目標を設定し一斉指導を中心に進めることとした。

イ 【本時】 プログラミング教材を用いた平行四辺形の作図

[ 1組: 9月8日(金), 3組: 9月15日(金) ]

目標 平行四辺形を作図するには図形の特徴を使って作図することができることを理解する。

(ア) プログラミング教材を使用した平行四辺形の作図

1組は導入と本時の時数が空いたため、操作の仕方を忘れてしまった児童が多かった。そこで正方形の作図から復習し、その後平行四辺形の作図に取り組んだ。操作の仕方を思い出してからの活動はスムーズに行っていた。

グループでの話合いの場面を設け、課題解決につながる意見を出し合う中から、ヒントを得ることで、どの児童も作図を完成させることができるよう配慮した。



【図6】話合いの様子

【資料5】平行四辺形の作図の場面（一部抜粋）

1:C1 これ80じゃなくて100じゃない?	12:C1 これでいいかな?実行。いけるんじゃない?
2:C2 もうちょっと長くして。ここ120にしてみて。	13:C2 やったー。できたー。
3:C1 ここで「左に曲がる120°」。	14:C1 みんな納得した?
4:C2 「前に進む120pxでしょ」。	15:C したした。
5:C1 この次が多分「右に曲がる70°」。	16:T あれっ?これ反対じゃない?
6:C2 「右に曲がる110°」じゃない?70°の反対って言うってたでしょ。	17:C ん?
7:C1 ここまで実行してみよう。	18:C あれ?
8:C2 おお〜っ。できてる。	19:T 頂点Bから出発だよ。君たちののは頂点Aからになるてる。
9:C1 次が「前に進む120px」	20:C1 えー?じゃあどうするの?
10:C2 え、80pxでしょ。	21:T もう一度頑張りましょう。
11:C1 実行。あってる。これを繰り返せばいいんだ。	

内角を使って入力しようとするC1に対し、外角で入力することを理解したC2が軌道修正しながら進めていた。

話し合いの中ではキーワードとなるコードの順序や外角での入力に関する意見が活発に出され、個人の作業では完成できなかった児童も、友達の意見を参考にし、完成することができた。また、「反復（ループ）」のコードを使い、より簡潔に作図しようとする取組みも見られた。

(イ) ワークシートの活用

画面上で並べるコードが増えていくと、現在の操作がどの辺や角を表しているのかの判別がつかなくなる。そこで、画面と同じコードをワークシートに張り付け、入力したコードが実際の図形のどの辺や角に対応しているかを記述させた。コードと辺や角を対応させながら、手順を意識して発表させるアイテムとしても使用した。



【図7】ワークシート記入例

【資料6】作図した平行四辺形の発表場面（一部抜粋）

1:C1 まず点Dの方を向くために左に90° 曲がります。	7:C1 次にBに行くために右に70° 曲がります。
2:C1 次に、頂点Dに行くために120px 進みます。	8:C1 次に前に120px 進んで頂点Bにつきます。
3:C2 はい。	9:C2 最後に角Aを向くために右に110° 曲がります。
4:C1 頂点Cの方を向くために右に110° 曲がります。	10:C1 前に80px 進んで角Aについて平行四辺形ができます。
5:C1 前に80px 進んで頂点Cにつきます。	11:C2 はい。大丈夫です。できました。
6:C2 はい。	

始点が頂点Aからの場合の作図を隣同士で発表しあう活動を行った。

C1の児童は「頂点」「点」「角」など表現のしかたが統一されていなかったが、「まず」「次に」「最後に」など、接続語を使いながら順序を意識しながら発表することができていた。

この後、発表役を交代し、C2の児童が発表した。



(2) 授業実践② [6年生 理科「電気とわたしたちの暮らし」]

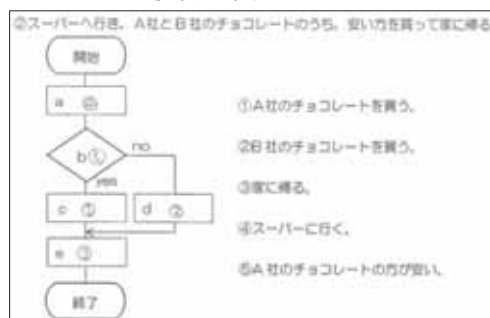
ア 【導入】プログラミングに関わる導入 [1組, 2組:10月16日(月), 3組:10月18日(水)]

目標 プログラミングを利用し電気を制御する活動を通して、電気が光や音、熱や運動に変換できることを理解する。

(ア) コンピュータサイエンスアンプラグドの活動とプログラミング教材の関わり

6年生では「順次」「分岐」「反復」といったアルゴリズムの基本処理を視覚的に捉えるため、フローチャート図を使用した。コンピュータサイエンスアンプラグドの活動では、日常生活を題材として取り上げたことで、抵抗感なく理解することができたようである。

また、使用する教材もフローチャート図を活用す



【図8】フローチャート図の記述



るため、コンピュータサイエンスアンプラグドの活動で得た知識を活かし、コンピュータを操作することができた。

(イ) コンピュータサイエンスアンプラグドの活動

「身の回りにはコンピュータが内蔵されている道具がたくさんあること」、「コンピュータに仕事をさせるために手順を書き込むことをプログラミングということ」、「手順をわかりやすくするためにフローチャート図を使うこと」とつなぎ、日常生活の様子をフローチャート図に表し、物事の手順について考えさせた。コンピュータに入力する際に必要な「順次」「分岐」「反復」の3つのパターンについて取り組んだ。

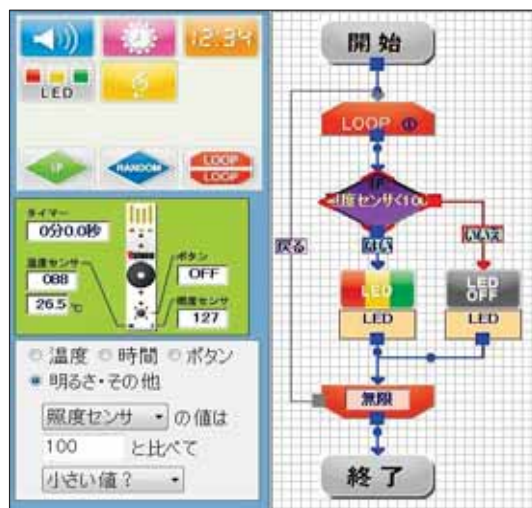
(ウ) プログラミング教材について

理科では、外部入出力装置として「計測制御プログラマー」を、使用ソフトウェアは「ビュートビルダーP」(ともにヴィストン株式会社)を選択した。「ビュートビルダーP」のソフトでプログラミングし、「計測制御プログラマー」を接続、データを送信することで本体のLEDを光らせたり音を鳴らしたりすることができるものである。自分の作成したプログラムが自分の意図した光や音に変換されることで、プログラミングをより体感することができる。本教材には「LED」、「ブザー」、「ボタン」、「照度センサー」、「温度センサー」が一つのパッケージの中に備えられており、すぐに操作・実行することができるという面で適切であると考えた。

また、6年生の発達段階を考慮し、フローチャート図により思考過程を視覚的にとらえることで論理的思考を意識させることをねらいとして選択した。さらに実際の機器を使つての操作を伴うので、「全員が触れられるもの」「体験を通しプログラミングを実感できるもの」「安価なもの」などの理由から選択した。

(エ) プログラミング教材を用いた制御の活動

光や音に変換した電気をプログラムによって制御する操作を行った。「光」や「音」、「分岐」などのブロックを並べ、矢印でつなぐことでフローチャート図を完成させる。この命令を「計測制御プログラマー」に送信し、動作させることになる。画面上の操作が実際の機器に反映されるため、自分で作成したプログラミングをより実感することができる。教材の操作は比較的短時間で身に付けることができ、練習問題以外にも積極的にプログラムを書き込む様子が見られた。



【図9】作業画面

イ 【本時】 センサーによる電気の制御

[1組:10月18日(水), 2組:10月17日(火), 3組:10月19日(木)]

目標 プログラミング教材を用いて電気を制御する活動を通し、センサーを利用し電気を有効活用することの大切さについて理解する。

(ア) センサーを利用した道具

センサーの働きについて説明し、身の回りにある様々なセンサーや道具、働きなどについて話し合わせた。児童は思いのほかたくさんのセンサーを見つけ、またそれを適切に分類することができていた。話合いを通し、自分たちがセンサーに囲まれて生活していることを実感することができた。

センサー	センサーを使った道具
光 センサー	太陽光発電、ライト、テレビ、計算器
温度 センサー	クーラー、ヒーター、れいぞうこ
距離 センサー	ロボットろいじき、自動車(自動ブレーキ)
人感 センサー	自動ドア、トイレ、防犯カメラ
タッチ センサー	改札口、水道、自動ドア
音 センサー	ポップパー、Siri
顔面 センサー	羽田空港の顔面認証、スマホ

【図10】ワークシート記入例「身の回りにおけるセンサーを使った道具」

【資料7】身の回りのセンサーについての話合いの場面（一部抜粋）

1:C1 自動ドア。	10:C1 人感と距離の両方に入れておこう。
2:C2 何センサー？	11:C3 トイレの手洗い場の水道もセンサーだ。
3:C1 人感センサー？	12:C1 人感？距離？
4:C3 スマホはタッチセンサー。	13:C2 両方ありそうだね。
5:C2 タブレットもタッチセンサー。	14:C1 入ると明かりがつくから照明もセンサーだ。
6:C1 トイレ。	15:C2 あとはルンバ（自動掃除機）。
7:C2 男子のトイレは人感センサーだよ。	16:C1 ルンバは何センサーだろう？距離？
8:C1 人を感知して水を流すよね。	17:C3 ぶつかると向きを変えるからタッチセンサー……う～ん？
9:C2 距離センサーじゃない？離れると水を流すよ。	18:C2 プリントにはないセンサーじゃない？

(イ) プログラミング教材を用いた制御の活動

「ビュートビルダーP」を使用し、照度センサーや温度センサーを利用した道具の仕組みについて学ぶためのプログラムの作成に取り組みさせた。児童は、条件によりLEDを光らせたたり音を鳴らしたりする活動を通してセンサーの働きを理解することができた。中には自分で条件を変えたプログラムを作成する児童もあり、発展的な学習につながった。

(3) 授業の検証

ア 授業実践前後の意識調査から

論理的思考力に関する児童の意識の変容を検証するために、授業実践前後で意識調査を行い $\chi^2$ 検定によって分析した。その結果、4年生、6年生ともに、「考えたり伝えたりするときに筋道や順番に気を付けているか」の項目で有意差が認められた【表7】。

これは4年生では図形を作図する過程で、6年生ではフローチャート図を使いアルゴリズムを構築していく中で、コンピュータに指示を出すためには正しい手順が必要なことを学び取り、発表やノートの記述の際に順序を意識しようという表れであると考えられる。プログラミングの体験を通して児童が筋道や順序の大切さに気付くことができたと思われる。

今後の課題として、この取組みが本時のみで終わることなく、他単元や他教科でも活かせるような手立てを考えていく必要がある。

【表7】論理的思考力に関わる意識の変容

4年生事前事後アンケートから（論理的思考力に関わる設問）

「何かを考えたり伝えたりするときに、筋道や順番に気を付けていますか。」 N=62

	事前	事後	合計	$\chi^2$ 検定
A	13	20	33	*9.77
B	31	37	68	
C	16	5	21	
D	2	0	2	
合計	62	62	124	

表中A とても気を付けている  
 B 気を付けている  
 C あまり気を付けていない  
 D 気を付けていない をそれぞれ表す。

6年生事前事後アンケートから（論理的思考力に関わる設問）

「問題を解くときに、筋道や順番に気を付けていますか。」 N=100

	事前	事後	合計	$\chi^2$ 検定
A	11	28	39	*15.32
B	67	64	131	
C	18	8	26	
D	4	0	4	
合計	100	100	200	

「自分の考えを伝えるときに、筋道や順番に気を付けていますか。」 N=100

	事前	事後	合計	$\chi^2$ 検定
A	8	25	33	*15.84
B	58	58	116	
C	33	15	48	
D	1	2	3	
合計	100	100	200	

- (注) 1 4年生の事前調査は8月21日(月)、事後調査は1組9月14日(木)、3組9月20日(水)にそれぞれ実施した。6年生の事前調査は10月10日(火)～13日(金)の期間に、事後調査は10月23日(月)～27日(金)の期間にそれぞれ実施した。
- 2 4年生のアンケート調査の結果を受けて、より詳細なデータを取るために、6年生では思考する場面と発表する場面に分けて実施した。
- 3 \*は有意水準5%で、有意差があることを示す。
- 4  $\chi^2$ 検定に用いた公式は下記に示す通りである。  

$$\chi^2 = \frac{(O1-E1)^2}{E1} + \frac{(O2-E2)^2}{E2} + \dots + \frac{(Ok-Ek)^2}{Ek}$$

イ スクリーンキャプチャによる思考過程の学習ログ分析から

村上ら(2017)は「数学の図形問題における解答に至る過程の可視化と分析」において、教師が適切な学習指導を行うための支援に活用することを目的として、タブレットPCへの解答記述における時間や筆跡といった記述の過程を表す時系列データの学習ログの分析を行っている。これは「書き直しが多い」「考え込んでいる時間が長い」などの最終的な答案には表れない情報を用いて思考過程を分析することを目指している。この研究を参考に、児童のコンピュータの画面上での操作の様子を可視化するために、コンピュータの画面を一定間隔ごとに画像として保存し、後日このデータを用いて、その操作の様子から読み取れる範囲で思考過程を分類した。

(ア) 4年生算数科の学習ログから

4年生算数科の第3時の授業の学習ログを【図11】に、第4時の授業の学習ログを【図12】に示す。

縦軸は児童、横軸は時間の経過（キャプチャ間隔は2秒毎、図中では1マス30秒ごとに記載）を表し、コンピュータを活用した実習時間のみを抽出して表記した。また、それぞれの学習時間内のプロセス1（思考が論理的かつ正しく働いている状態）の割合を算出し、この値の大きい順

に児童を並べ直したものである。ただし、第4時の後半のように、グループ活動の際に話合いの中心として使われた場合もあり、コンピュータを操作している時間は一定ではない。

コンピュータの操作中にプログラミング的思考の5つの能力「抽象化(AB)」・「分解(DE)」・「アルゴリズム的思考(AL)」・「評価(EV)」・「一般化(GE)」の、どの能力が活用されていたのかを図中に色別に示した。

完成度の欄には、1単位時間内の作図の完成度合をA(完成)、B(途中まで)、C(未完成)で評価し示した。第4時については、完成度Ⅰを自力解決での完成度、完成度Ⅱを話合いを経た完成度として示した。

第3時の前半(0:30~3:30)ではヒントを与えずに作図させたところ、全児童が完成できなかった。その後、内角と外角について指導し(4:00~17:30)、再び作図させた(完成15名/32名中)。

第4時の前半(0:30~5:30)では始点を頂点Bから頂点Aに変更して挑戦させた(完成15名/32名)。その後グループでの話合いの時間(6:30~11:30)を設定した。(完成20名/32名)。

完成、未完成に関わらず、プログラミング的思考の5つの能力のうち、活用した能力が一つでも増えたと認められた児童は緑のマス(ア)として表記した(9名/32名)。

コンピュータ操作のスキルが高ければ、コードも素早く入力でき、結果、思考時間は短く表れる。教材への理解力の高さ、教材ソフトの操作の慣れ具合などの要因を考慮すると、プロセス1が長く続いた児童だけが論理的思考力を用いて解決している訳ではないと思われる。

図形の作図では、「問題から抽出した数値(辺の長さや角度)(抽象化)」を「作図に必要なコードに当てはめ(分解)」、「手順に沿って並べ替える(アルゴリズム的思考)」の能力が必要となる。また、実行した際に、意図した通りに動かない場合、「間違っている部分を修正(評価)」する能力も必要である。これらの能力を使い、「他の問題(2問目)に置き換えて(一般化)」考えることで、論理的思考力につながっていく。

2問とも正答、または1問目から2問目にかけて伸びが見られた児童(65.6%:21名/32名中)は5つの能力を全て活用できているともいえる。

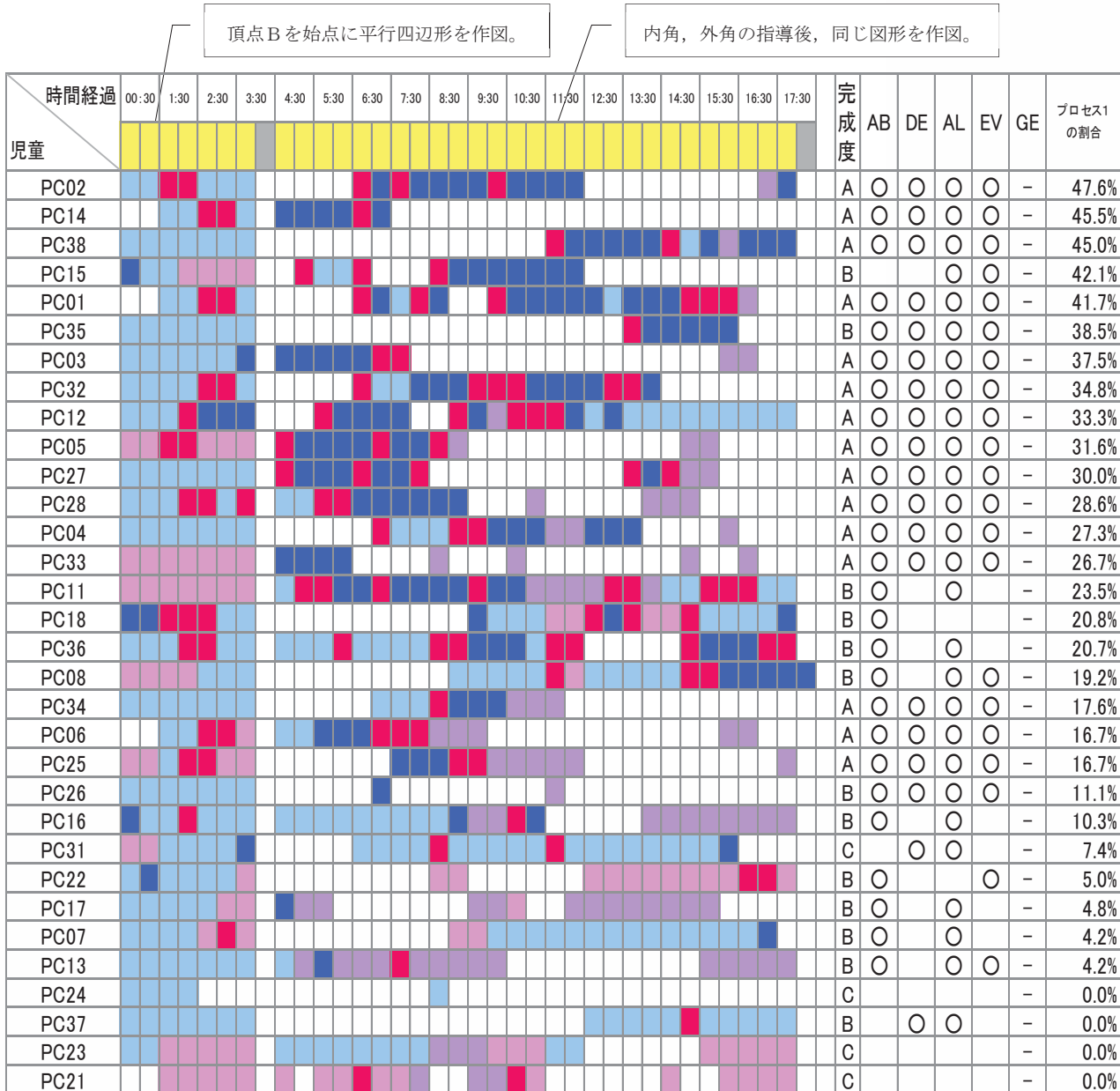
またプロセス1の割合が多い児童、プロセス3の割合が多い児童の画面の動きには次のような特徴が見られた【表8】。

**【表8】 プロセスの割合の違いによる操作の特徴**

プロセス1の割合が多い児童	プロセス3の割合が多い児童
<ul style="list-style-type: none"> <li>・手を止めている最中にコードを組み立ててから操作を始める。</li> <li>・その操作は正答であることが多い。</li> </ul>	<ul style="list-style-type: none"> <li>・同じコードを何度も入力する。</li> <li>・必要な数値の入力を行わない。</li> <li>・角度を1度ずつ変えて入力する。</li> </ul>
<ul style="list-style-type: none"> <li>・いずれもプロセス4(画面停止)後に表れることが多い。</li> </ul>	

プロセス1の割合が多い児童に見られる特徴として、画面停止の状態の後に正解を導き出していることが多い。これは問題に直面した際に「抽象化」、「分解」、「アルゴリズム的思考」などを頭の中である程度組み立ててから操作しているとみることができる。一方プロセス3の割合が多い児童の特徴は、一貫性のない操作が多く見られた。これらの操作は主にプロセス4後に表れてくることが多い。思考が停滞している際には、操作の仕方でも悩んでいるのか、図形の特徴を活かせず悩んでいるのかを確認し、適切に支援する必要がある。

### 第3時「頂点Bを始点として作図」(算数)



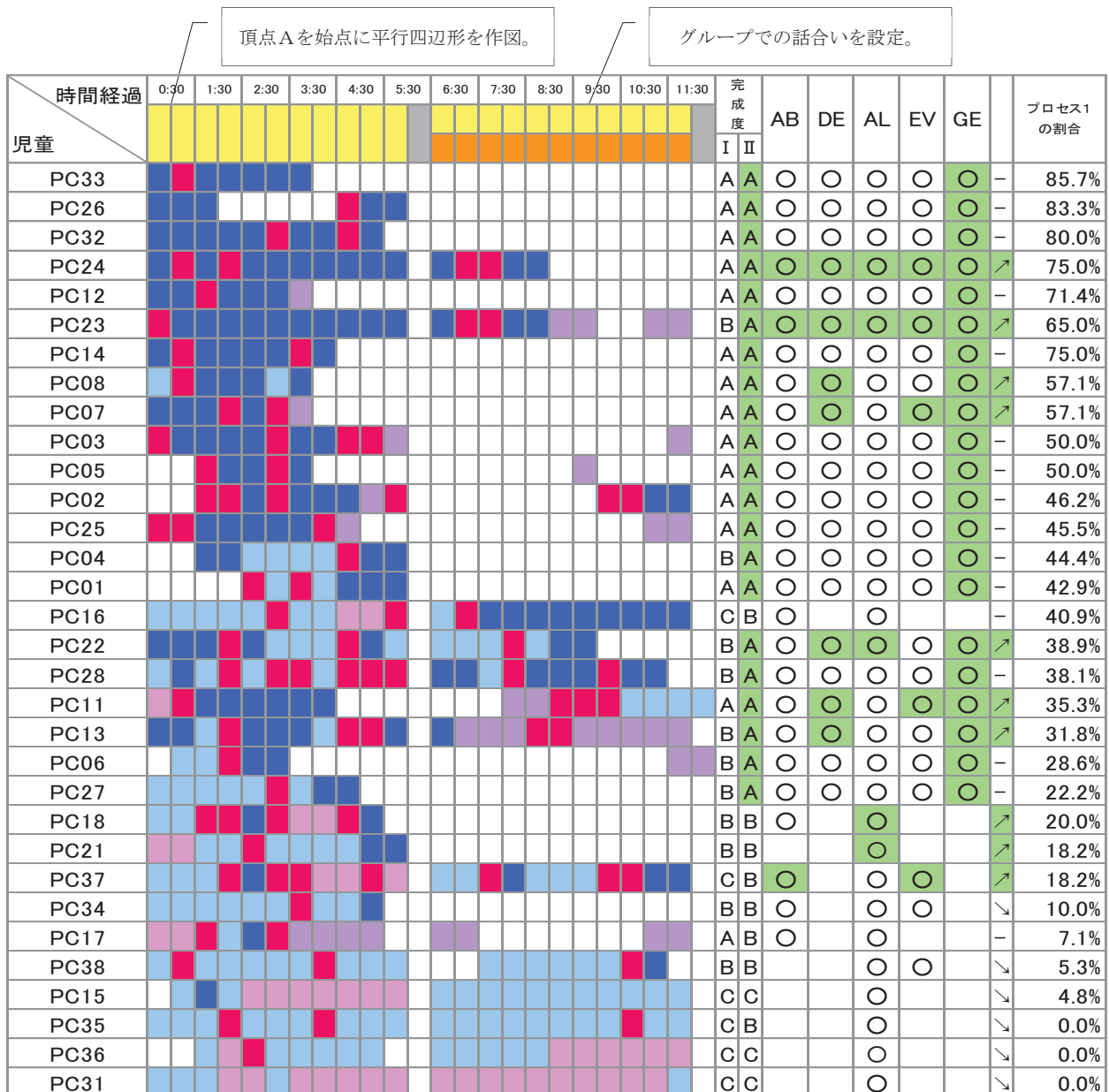
※縦軸は児童、横軸は時間の経過（キャプチャ間隔は2秒毎、図中では1マス30秒ごとに記載）

プロセス	色	思考過程の状態
プロセス1	青	思考が論理的かつ正しく働いている状態
プロセス2	水色	正答ではないが、思考が論理的に働いている状態
プロセス3	桃	思考しているが、論理的に働いていない状態
プロセス4	赤	画面が停止している状態（話し合いや操作が終了していることが明らかな場合は省略）
プロセス5	紫	その他の操作をしている状態
プロセス6	黄	主にコンピュータを操作している時間
プロセス7	橙	主に話し合いをしている時間
プロセス8	緑	本体と接続、本体の操作をしている状態

【図 11】スクリーンキャプチャによる思考過程の学習ログ①（4年生算数科）



### 第4時「頂点Aを始点として作図」(算数)



※縦軸は児童、横軸は時間の経過（キャプチャ間隔は2秒毎、図中では1マス30秒ごとに記載）

プロセス	色	思考過程の状態
プロセス1	青	思考が論理的かつ正しく働いている状態
プロセス2	水色	正答ではないが、思考が論理的に働いている状態
プロセス3	桃	思考しているが、論理的に働いていない状態
プロセス4	赤	画面が停止している状態（話し合いや操作が終了していることが明らかな場合は省略）
プロセス5	紫	その他の操作をしている状態
プロセス6	黄	主にコンピュータを操作している時間
プロセス7	橙	主に話し合いをしている時間
プロセス8	緑	本体と接続、本体の操作をしている状態

【図12】スクリーンキャプチャによる思考過程の学習ログ②（4年生算数科）



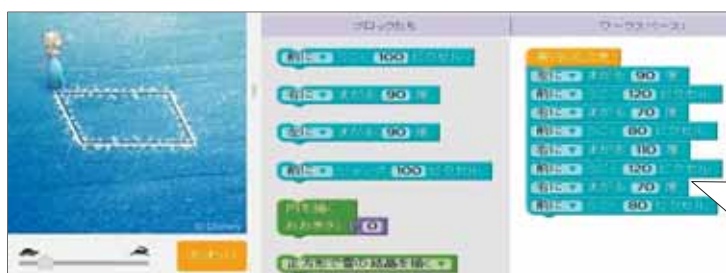
これらの児童に対しての支援として、操作で悩んでいる場合には、宝探しゲームで方向感覚を確認したり、正方形の作図の練習を重ねたりして操作に慣れさせることが必要である。図形の特徴を活かせずに悩んでいる場合は、三角定規やコンパスで作図した時と同様に、平行四辺形の定義（向かい合った2組の辺が平行）や特徴（向かい合った辺の長さや角の大きさが等しい）を活かして作図することを確認する必要がある。

論理的思考力は短時間で身につくものではないが、第4時で活用できる能力を増やした児童が32名中9名見られた。このことからプログラミングの体験が論理的思考力の活用につながっていることがうかがえる。



外角の考え方も理解でき、一度で作図することができた。「繰り返し（ループ）」のコードを使い簡潔な作図の仕方に挑戦する児童も見られた。

【図13】児童の作図①



角度を内角で入力したために平行四辺形の向きが逆になってしまった。この後間違いに気づき、コードを入れ替えることで修正することができた。

【図14】児童の作図②



角度が分らず連続して同じ角度を入力してしまっている。混乱することもあるので、このような場合は新規作成からの支援も必要である。

【図15】児童の作図③

(イ) 6年生理科の学習ログから

6年生理科の授業の学習ログの例を【図16】に示す。

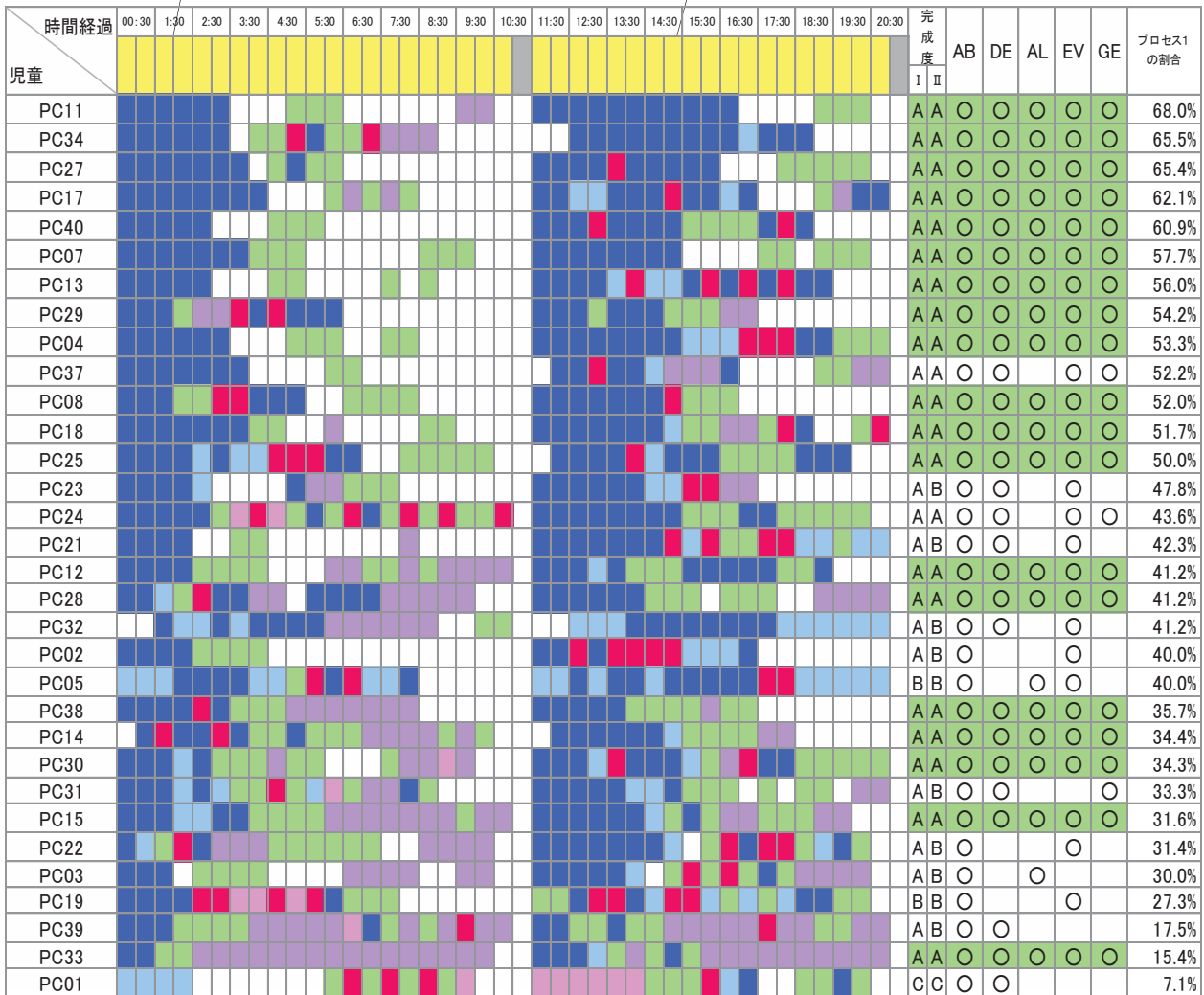
縦軸は児童、横軸は時間の経過（キャプチャ間隔は2秒毎、図中では1マス30秒ごとに記載）を表し、コンピュータを活用した実習時間のみを抽出して表記した。また、それぞれの学習時間内でプロセス1（思考が論理的かつ正しく働いている状態）の時間の割合を算出し、この値の大きい順に並べ直したものである。コンピュータの操作以外にもデータを送信し、教材本体でプログラムを実行する活動が伴うのでプロセス8（緑のマス）を設定している。

コンピュータの操作中にプログラミング的思考の5つの能力「抽象化(AB)」・「分解(DE)」・「アルゴリズム的思考(AL)」・「評価(EV)」・「一般化(GE)」の、どの能力が活用されているのかを図中に色別に示した。

### 第4時 「センサーを利用したプログラムの作成」(理科)

照度センサーの値が 100 より小さいとき  
LED が点灯する。

温度センサーの値が 30℃ より高いとき  
LED が点灯しドレミの音が鳴る。



※縦軸は児童、横軸は時間の経過 (キャプチャ間隔は2秒毎、図中では1マス30秒ごとに記載)

プロセス	色	思考過程の状態
プロセス1	青	思考が論理的かつ正しく働いている状態
プロセス2	水色	正答ではないが、思考が論理的に働いている状態
プロセス3	桃	思考しているが、論理的に働いていない状態
プロセス4	赤	画面が停止している状態 (話し合いや操作が終了していることが明らかな場合は省略)
プロセス5	紫	その他の操作をしている状態
プロセス6	黄	主にコンピュータを操作している時間
プロセス7	橙	主に話し合いをしている時間
プロセス8	緑	本体と接続、本体の操作をしている状態

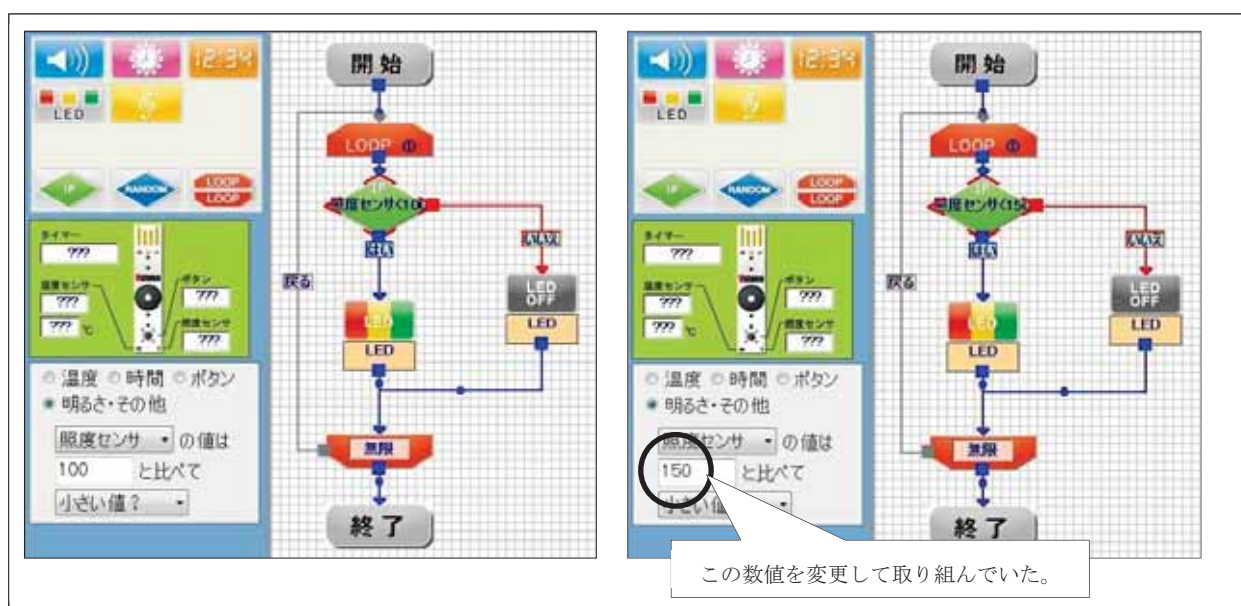
【図16】スクリーンキャプチャによる思考過程の学習ログ (6年生理科)

完成度の欄には1単位時間内のフローチャート図の完成度合をA（完成）、B（途中まで）、C（未完成）の評価で示した。完成度Ⅰを1問目、完成度Ⅱを2問目の完成度として示した。

コンピュータ操作のスキルが高ければ、フローチャート図も素早く入力でき、結果、思考時間は短く表れる。教材への理解力の高さ、教材ソフトの操作の慣れ具合などの要因を考慮すると、プロセス1が長く続いた児童だけが論理的思考力を用いて解決している訳ではないと思われる。

フローチャート図を作成するには、「問題から抽出した数値や語句などの要素（抽象化）」を「各ブロックに当てはめ（分解）」、「手順に沿って並べ替える（アルゴリズム的思考）」の能力が必要となる。また、本体と接続し動作を確認した際に、意図した通りに動かない場合、「間違っている部分を修正（評価）」する能力も必要である。これらの能力を使い、「他の問題（2問目）に置き換えて考える（一般化）」ことで、論理的思考力につながっていく。2問とも正答している児童が、5つの能力を有効に活用できているともいえる（59.4%：19名/32名中）。

6年生の特徴の一つとして、早い段階で正答に至った児童が、独自に操作を試みる場面がいくつか見られた。例えば「照度センサーの値が100より小さいとき、LEDが点灯する」という課題を解決した児童が「照度センサーの値を大きくしたり小さくしたりするとどうなるのか」、「条件による点灯と消灯を逆にしたらどうなるのか」など、数値や条件を変えながら取り組んでいた（【図16】の紫色のログ）。本来このような、自ら新しい課題を見つけ解決していこうとする学習の中にこそ、プログラミングの体験の本質があると感じる。



【図17】児童の作業画面の例

#### ウ ワークシートの記述から

児童の思考を可視化するもう一つの方法として、ワークシートを作成し記録を残すこととした。プログラミングの体験を通し、プログラミング的思考の5つの能力「抽象化(AB)」・「分解(DE)」・「アルゴリズム的思考(AL)」・「評価(EV)」・「一般化(GE)」のどの力が活用されたのかを確認するために、ワークシートの記述を検証した。



(ア) 算数科のワークシートから

ワークシートには画面と同じコードの縮小版を一人ひとりに用意し、画面上で完成させた平行四辺形の作図の手順を貼らせ、対応する辺や角についても書き込ませた。児童の記述の例を【図18】に示す。

児童Aの記述

児童Aの記述は、2つの手順で平行四辺形を作図する。手順1は頂点Bからスタートし、手順2は頂点Aからスタートする。各ステップで「実行したとき」のコード（左にまがる、前にうごく）と、それに対応する辺や角（辺BC、角C、辺CD、角D、辺DA、角A、辺AB、角B）を記述している。

児童Bの記述

児童Bの記述は、手順1（頂点Bからスタート）がほぼ完了しているが、手順2（頂点Aからスタート）のコードと対応関係の記述がほとんど空白である。

プログラミング的思考の5つの能力に関する評価結果

能力	概要	児童A	児童B
抽象化(AB)	必要な数値の抽出	○	
分解(DE)	作図に必要な辺や角のコードにあてはめる	○	
アルゴリズム的思考(AL)	作図の手順にコードを並べる	○	○
評価(EV)	間違いの修正, 確認	○	
一般化(GE)	実際の図形の辺や角に対応させる, 他の図形に置き換える。	○	

【図18】ワークシート記入例及び評価（4年生算数科）

児童Aの場合、「辺や角の数値の抽出（抽象化）」、「対応するコードにあてはめる（分解）」、「作図の手順に沿ってコードを並べる（アルゴリズム的思考）」、「間違いの修正や確認（評価）」、「実際の図形の辺や角にあてはめる、他の問題に置き換える（一般化）」など、全ての項目において活用で

初めは同じコードを繰り返し並べてしまっていた。声掛けをすることにより正しい順序に並べ替えることはできたが、数値の入力はできなかった。

【図19】児童Bの作業画面

きている様子うかがえる。一方、児童Bはコードを並べることはできているが、辺や角の数値を入力することができず、「抽象化」や「分解」の段階でのつまずきが見られることがわかる。

(イ) 理科のワークシートから

ワークシートには画面と同じフローチャート図を用意し、必要な数値や条件を記入できるようにした。児童の記述の例を【図 20】に示す。

**児童Cの記述**

数値や条件の抽出や記入、ブロックを線でつなげることができている。

**児童Dの記述**

条件の記入もれや○で囲んだ部分の接続が間違っているため 30℃より低い温度でも音になってしまう。

**プログラミング的思考の5つの能力に関する評価結果**

能力	概要	児童C	児童D
抽象化(AB)	必要な数値の抽出	○	○
分解(DE)	電気の制御に必要な数値や条件をあてはめる。	○	
アルゴリズム的思考(AL)	ブロック同士を正しくつなぎフローチャート図を完成させる。	○	
評価(EV)	間違いの修正、確認	○	
一般化(GE)	他の数値や条件に置き換えて実行してみる。	○	

【図 20】 ワークシート記入例及び評価（6年生理科）

児童Cの場合、「温度や音階などの抽出（抽象化）」、「対応するブロックにあてはめる（分解）」、「フローチャートの手順に沿ってブロックを並べる（アルゴリズム的思考）」、「間違いの修正や確認（評価）」など、全ての項目において力を活用できている様子うかがえる。一方、児童Dは条件を記入する欄が抜けていたり、ブロックをつなげる矢印の位置が違っていたりするため、プログラムを実行した際に正しい反応が得られなかった。

ワークシートは児童一人一人の思考を最終的な記録として残すことができるので、ワークシートの記述からでも思考の様子を読み取ることが十分可能である。ゆえにワークシートの作成にあたっては、何を見取りたいのかというねらいをはっきりとさせた上で作成することが必要であると考え。その際、児童の能力を見取るために、ワークシートには右記のような評価欄を設けておくと思われ【表 9】。

【表 9】ワークシートの評価欄（例）

①AB	②DE	③AL	④EV	⑤GE	計
					/
完成度		A	B	C	

エ 児童の振り返りの記述から

単位時間の振り返りの際に、自由記述でわかったことや感想を記述させた。得られた感想を大別すると「①プログラミング的思考に関する内容」、「②単元や本時の目標に関する内容」、「③コンピュータ操作や日常に還元される内容」となった。

そこで、児童の作図やフローチャート図の「完成度」、プログラミング的思考の「5つの能力」を点数化し、その合計点をもとに学級を上位、中位、下位の層に分割した際、それぞれの記述が各層にどの程度の割合で表れてくるのかを検証した【表 11】【表 12】。点数化に際しての基準は以下の通りである。

- ・完成度 I は、作図やフローチャートの完成度を「A:完成(2点)」、「B:途中まで(1点)」、「C:未完成(0点)」とした。
- ・完成度 II は、発展問題のため、配点率を高く設定し、「A:完成(3点)」、「B:途中まで(1.5点)」、「C:未完成(0点)」とした。
- ・プログラミング的思考の5つの能力「抽象化(AB)」「分解(DE)」「アルゴリズム的思考(AL)」「評価(EV)」「一般化(GE)」は各1点とした。
- ・上記の3つを合計し10点満点とし、上位と下位を10人ずつ、それ以外を中位とし3層に分け、各層内で①～③の記述の割合を求めた。
- ・同点数の場合は、前述のプロセス1の割合が多い児童から順に並べている。
- ・「①プログラミングに関する内容」については、それぞれ「プログラミング的思考の5つの能力」のどの能力について感想が出されたかを表中に色別（緑のマス）で示した。

児童の感想を分類するにあたり、記述の中に含まれる以下のようなキーワードに関連するものを参考に分類した【表 10】。

【表 10】感想の分類に関わるキーワード

分類	キーワード
①	算数：順序、順番、角度、長さ、指示、間違い、修正 理科：順序、順番、数値、指示、修正 等
②	算数：図形の特徴、特徴を活かす、特徴を使う 理科：制御、省エネ、エコ、節電、効率 等
③	算数：楽しかった、難しかった、またやりたい、 理科：身の回り、身近、センサー（の働き）、利用 等



【表 11】 児童の評価結果と授業後の感想（4年生算数科）

範囲	児童	完 成 度		5つの能力					合計 (点)	プロセス 1 の割合	プロセス 2 の割合	分 類	感想	割合	
		I	II	AB	DE	AL	EV	GE							
上位	PC33	A	A	○	○	○	○	○	10.0	85.7%	0.0%	①	平行四辺形をかくのが難しかったけど、角度のかき方を教えてもらって書くことができました。	① 60.0%	
	PC26	A	A	○	○	○	○	○	10.0	83.3%	0.0%	①	パソコンは消しゴムみたいのリセットできる場所がすごいなと思いました。少し難しかったけど楽しかったです。		
	PC32	A	A	○	○	○	○	○	10.0	80.0%	0.0%	①	中の角度ではなく外の角度でプログラミングすることが分かりました。とても楽しかったです。		
	PC24	A	A	○	○	○	○	○	10.0	75.0%	0.0%	①	プログラミングをすればコンピュータが指示通り動くことが分かりました。		
	PC12	A	A	○	○	○	○	○	10.0	71.4%	0.0%	③	平行四辺形をかけるなんて思っていませんでした。次はかけるようになりたいです。		② 0%
	PC14	A	A	○	○	○	○	○	10.0	62.5%	0.0%	①	辺や角度の数字を変えると平行四辺形もかけることが分かりました。		
	PC08	A	A	○	○	○	○	○	10.0	57.1%	28.6%	③	頂点Bからスタートする平行四辺形が難しかったです。		③ 40.0%
	PC07	A	A	○	○	○	○	○	10.0	57.1%	14.3%	③	いろんなことが勉強できてうれしかったです。またやってみたいです。		
	PC03	A	A	○	○	○	○	○	10.0	50.0%	0.0%	③	みんなと協力し合えたのでよかったです。家でもまたやりたいです。		
	PC05	A	A	○	○	○	○	○	10.0	50.0%	0.0%	①	三角定規ではやり直しに時間がかかったけど、コンピュータではすぐにかき直しができるとわかってよかったです。		
中位	PC02	A	A	○	○	○	○	○	10.0	46.2%	0.0%	③	少し難しかったけどとても楽しかったです。家でもやってみたいと思いました。	① 41.7%	
	PC25	A	A	○	○	○	○	○	10.0	45.5%	0.0%	①	指示を間違えると大変だということが分かりました。		
	PC01	A	A	○	○	○	○	○	10.0	42.9%	28.6%	②	コンピュータでは特徴を活かしてかくことができました。繰り返し、正確さ、速さなどがすごいと思いました。		
	PC11	A	A	○	○	○	○	○	10.0	35.3%	23.5%	①	順番が大切なんだということが分かった。		
	PC23	B	A	○	○	○	○	○	9.0	65.0%	0.0%	①	最初の問題が分からなかったけど次の問題が分かったのでよかったです。		
	PC04	B	A	○	○	○	○	○	9.0	44.4%	44.4%	①	プログラミングを使ってコンピュータに指示できることが分かりました。		② 25.0%
	PC22	B	A	○	○	○	○	○	9.0	38.9%	50.0%	②	平行四辺形の特徴を使って楽しくできました。何回もやり直したけど順序に気を付けてできました。		
	PC28	B	A	○	○	○	○	○	9.0	38.1%	23.8%	③	少し難しかったけど班の人と協力することができたのでよかったです。今日できなかったところもいつかできるといいなと思いました。		③ 33.3%
	PC13	B	A	○	○	○	○	○	9.0	31.8%	13.6%	③	少し難しかったけれどできたのでうれしかったです。家でもまたやりたいです。		
	PC06	B	A	○	○	○	○	○	9.0	28.6%	28.6%	②	コンピュータでも特徴を活かすと平行四辺形がかけるとことが分かりました。		
下位	PC17	A	B	○	○	○	○	○	5.5	7.1%	7.1%	②	コンピュータで作図するときは作るものの特徴を活かしてかけばいいことが分かりました。	① 30.0%	
	PC18	B	B	○	○	○	○	○	4.5	20.0%	20.0%	①	「前に100px進む」とか「右に90° 曲がる」とかが難しかったけど、楽しかったです。		
	PC37	C	B	○	○	○	○	○	4.5	18.2%	40.9%	③	コンピュータの勉強は面白いなと思いました。またやってみたいです。		
	PC38	B	B	○	○	○	○	○	4.5	5.3%	78.9%	③	最後まであきらめないでやることができよかったです。		
	PC16	C	B	○	○	○	○	○	3.5	40.9%	36.4%	①	プログラミングは正しくやらないとコンピュータがその通りに動いてしまうので気を付けようと思いました。		② 10.0%
	PC21	B	B	○	○	○	○	○	2.5	18.2%	54.5%	③	今日は平行四辺形をかくことができてよかったです。		
	PC35	C	B	○	○	○	○	○	2.5	0.0%	77.3%	③	とても難しかったけどとても楽しかったです。もう一度やりたいです。		③ 60.0%
	PC15	C	C	○	○	○	○	○	1.0	4.8%	61.9%	③	難しかったけど分かったことがたくさんありました。もっとやってみたいです。		
	PC31	C	C	○	○	○	○	○	1.0	0.0%	22.7%	①	コンピュータは自分で指示して動くことが分かりました。		
	PC36	C	C	○	○	○	○	○	0.0	0.0%	42.1%	③	コンピュータは算数などいろいろな教科でできることが分かりました。またやりたいです。		

※①プログラミング的思考に関する内容 ②単元や本時の目標に関する内容 ③コンピュータ操作や日常に還元される内容

※①の感想に関しては、5つの能力のうちどの能力について感想が出されたかを表中に色別で示した。

※割合の欄は各層内の①～③の感想を記述した人数を割合として示している。

N=32

【表 12】 児童の評価結果と授業後の感想（6年生理科）

範囲	児童	完成度		5つの能力					合計 (点)	プロセス1 の割合	プロセス2 の割合	分類	感想	割合
		I	II	AB	DE	AL	EV	GE						
上位	PC11	A	A	○	○	○	○	○	10.0	68.0%	0.0%	③	プログラミングされているものにはどんなものがあるのかが分かった。	① 40.0% ② 10.0% ③ 50.0%
	PC34	A	A	○	○	○	○	○	10.0	65.5%	3.4%	③	身の回りにはセンサーを使ったものがたくさんあることが分かった。	
	PC27	A	A	○	○	○	○	○	10.0	65.4%	0.0%	①	「○○のとき○○をする」というのを考えて作らないと失敗してしまうのでそこが面白かったです。	
	PC17	A	A	○	○	○	○	○	10.0	62.1%	10.3%	①	きちんとブロックを線ぞつないでできたのでよかったです。	
	PC40	A	A	○	○	○	○	○	10.0	60.9%	0.0%	③	身の回りにはコンピュータやセンサーで動いているものがたくさんあることが分かってよかった。	
	PC07	A	A	○	○	○	○	○	10.0	57.7%	0.0%	①	順序に気を付けることが大切だとわかった。	
	PC13	A	A	○	○	○	○	○	10.0	56.0%	12.0%	③	センサーを使った道具がたくさんあることが分かった。	
	PC29	A	A	○	○	○	○	○	10.0	54.2%	0.0%	③	センサーは身の回りでもよく使われていることが分かった。	
	PC04	A	A	○	○	○	○	○	10.0	53.3%	10.0%	②	電気はいろいろなものに変えていくことができることが分かった。センサーの働きも分った。	
	PC08	A	A	○	○	○	○	○	10.0	52.0%	0.0%	①	筋道や順序に気を付ければ、他の学習にも役立つことが分かった。	
中位	PC18	A	A	○	○	○	○	○	10.0	51.7%	3.4%	①	順序を確かめて初めから終わりまで一つずつしっかりとつなぐことが大切なことが分かった。	① 33.3% ② 50.0% ③ 16.7%
	PC25	A	A	○	○	○	○	○	10.0	50.0%	11.8%	②	制御をするといいいことがたくさんあることが分かった。	
	PC12	A	A	○	○	○	○	○	10.0	41.2%	2.9%	②	センサーを使うと省エネになるのですごいと思った。	
	PC28	A	A	○	○	○	○	○	10.0	41.2%	2.9%	②	センサーを利用すると人間が楽に生活できるし、エコにつながるものが分かった。	
	PC38	A	A	○	○	○	○	○	10.0	35.7%	0.0%	①	うまく動くともううれしいし、うまくいかないときにどうしてだろうと考えることができてもよかったです。	
	PC14	A	A	○	○	○	○	○	10.0	34.4%	3.1%	①	順序に気を付けて操作することができた。	
	PC30	A	A	○	○	○	○	○	10.0	34.3%	8.6%	②	センサーは無駄をなくしたり、感知して行動しているのがすごいと思いました。	
	PC15	A	A	○	○	○	○	○	10.0	31.6%	7.9%	②	電気を制御するときは、センサーを利用すると効率よく使うことができるということが分かった。	
	PC33	A	A	○	○	○	○	○	10.0	15.4%	2.6%	③	プログラミングの仕方が分ってよかったです。	
	PC37	A	A	○	○	○	○	○	9.0	52.2%	4.3%	③	パソコンとつなぐだけで音が鳴ったり光ったりしてすごなあと思った。	
	PC24	A	A	○	○	○	○	○	9.0	43.6%	0.0%	②	センサーを利用して電気を制御することで効率よく使うことができることが分かった。	
下位	PC21	A	B	○	○	○	○	○	6.5	42.3%	19.2%	②	センサーを使うことで使うエネルギーを減らして節電やエコなどにつながるものが分かりました。	① 30.0% ② 40.0% ③ 30.0%
	PC32	A	B	○	○	○	○	○	6.5	41.2%	35.3%	③	プログラミングのことを詳しく知ることができ、もっと知りたいと思った。	
	PC31	A	B	○	○	○	○	○	6.5	33.3%	15.2%	①	プログラミングをするときは順序などに気を付けなければならないことが分かった。	
	PC05	B	B	○	○	○	○	○	5.5	40.0%	45.7%	①	順序に気を付けてできたのでよかったです。	
	PC02	A	B	○	○	○	○	○	5.5	40.0%	15.0%	①	温度の数値を忘れていたのでもちんと最後まで見るようにして完璧にしたい。	
	PC22	A	B	○	○	○	○	○	5.5	31.4%	8.6%	②	センサーを使うと電気を無駄にしないでエコができることが分かった。	
	PC03	A	B	○	○	○	○	○	5.5	30.0%	3.3%	②	センサーを使うと電気を節電することができることが分かった。	
	PC39	A	B	○	○	○	○	○	5.5	17.5%	0.0%	②	センサーが省エネにつながっていることが分かった。	
	PC19	B	B	○	○	○	○	○	4.5	27.3%	12.1%	③	分らないときは友達と協力することができた。	
	PC01	C	C	○	○	○	○	○	2.0	7.1%	17.9%	③	センサーはプログラミングを使っていることが分かった。	

※①プログラミングに関する内容 ②単元や本時の目標に関する内容 ③学習の様子（話し合いなど）や日常に還元される内容

※①の感想に関しては、5つの能力のうちどの能力について感想が出されたかを表中に色別で示した。

※割合の欄は各層内の①～③の感想を記述した人数を割合として示している。

N=32

各学級、各層における①～③の記述の割合は以下のとおりである【表 13】。

【表 13】各層の記述の割合

教科	学級	上位			中位			下位		
		①	②	③	①	②	③	①	②	③
算数	4年1組	90.0%	0.0%	10.0%	41.7%	8.3%	50.0%	40.0%	0.0%	60.0%
	4年3組	60.0%	0.0%	40.0%	41.7%	25.0%	33.3%	30.0%	10.0%	60.0%
理科	6年1組	20.0%	20.0%	60.0%	28.6%	21.4%	50.0%	50.0%	0.0%	50.0%
	6年2組	40.0%	10.0%	50.0%	33.3%	50.0%	16.7%	30.0%	40.0%	30.0%
	6年3組	10.0%	60.0%	30.0%	7.7%	30.8%	61.5%	20.0%	40.0%	40.0%

各学級の児童の実態や記述力、授業展開の仕方等の差に左右される部分もあるが、得られた感想は前向きなものが多い。

「①プログラミング的思考に関する内容」に関して、「手順や順番が大切」と記述した児童は、コードやブロックの並べ方やつなぎ方、数値の入力などプログラミングの体験を通して論理的思考力を活用することができたと考える。また、他の学習にもこの考え方が活かされると発展的に考えることができている児童もいた。下位層の児童の中には、本時でその能力を発揮できなかった項目に関わる記述をしている児童も見られる。コンピュータの操作を通して、プログラミング的思考の大切さを感じ取っているように思える。継続して指導していく中で、徐々に様々な能力を活用できるようになっていくだろうと思われる。

「②単元や本時の目標に関する内容」では、「図形の特徴を活かして作図することが大切」や「センサーを利用すると省エネ、エコにつながる」等の記述をした児童は、プログラミングを学習のツールの一つであると捉え、本時の目標に迫ることができている。

「③コンピュータ操作や日常に還元される内容」の記述には、4年生と6年生に違いが見られた。4年生は「コンピュータ操作」に関わる内容が多く、「難しかった」が、「楽しかった」、「またやりたい」、「家でも続きをやってみたい」とプログラミングに興味を示していることがうかがえる。一方6年生は、「センサーを利用した道具が身近な生活の中にある」ことや「他のセンサーも探してみたい」等、学習したことを日常生活に関連付けて記述している内容が目立つ。これは理科がより日常生活につなげて考えやすいという教科の特性によるものであると考える。

また「プログラミングの学習が難しかった」という記述はいくつかあるものの、「分らない問題が解けた時がうれしかった」、「難しい問題に挑戦することが楽しい」など、肯定的に捉えられているものが多かった。

児童の振り返りや事後アンケートの記述を総合的にみると、授業実践を通してプログラミング的思考を意識して取り組んでいた児童が多いことがわかる。これは「ア 授業実践前後のアンケート調査から」の結果に通じるものである。児童が論理的思考力を活用しつつ学習を進めていくためには、他単元、他教科でのプログラミングの体験の活動場面を広げていくことが必要である。

#### オ プログラミングの体験の配置について

今回の授業実践では、4年1組（算数科）では導入と本時を離れたパターン②、4年3組（算数科）では導入と本時を続けて行うパターン③、6年生（理科）に関しては単元の構成上、単元の最後に行うパターン④でそれぞれプログラミングの体験を配置した【図 1】（再掲）。

パターン① 通常の授業展開の場合														
時数	1	2	3	4	5	6	7	8	9	10				
内容						作図・実験								

パターン② プログラミングの体験を単元の冒頭に配置する場合（プログラミングの体験に関する時数を便宜上0とする）														
時数	0	0	1	2	3	4	5	6	0	0	7	8	9	10
内容	プログラミング 教材の学習							作図・ 実験	教材を使 っての作 図・実験					

パターン③ プログラミングの体験を途中で配置する場合（第6時と第0時は交換可能）														
時数	1	2	3	4	5	0	0	6	0	0	7	8	9	10
内容						プログラミング 教材の学習		作図・ 実験	教材を使 っての作 図・実験					

パターン④ プログラミングの体験を終末に配置する場合														
時数	1	2	3	4	5	6	7	8	9	10	0	0	0	0
内容						作図・ 実験					プログラミング 教材の学習		教材を使っ ての作図・ 実験	

【図1】プログラミングの体験の配置（再掲）

導入と本時を離した場合、論理的思考の意識を継続しながら本時の学習を展開できるのではないかと考えて実践した。しかし、本時の授業を行った際には操作の仕方を忘れてしまっている児童が多く、復習の時間を取らざるを得なかった。一方パターン③やパターン④のように、導入と本時を連続して行った際には、前時の内容を活かして取り組むことができ、スムーズな展開で実施することができた。

前述の感想の欄にもあるが、「順序に気を付けて作図できた」と回答する児童はパターン②（4年1組）にもいたのだが、それがこの配置の工夫によるものなのかは判断できなかった。よって、授業の配置による効果については確認することができなかった。

ただし、4年生（算数科）では、コンピュータを使った作図を三角定規やコンパスを使っての作図の学習と関連させて行ったことで、コンピュータも作図する際に使用するツールの一つと捉えることができ、それぞれのよさを活かした作図方法を選択する必要性を理解することができたと思われる。

6年生（理科）についても、単元の最後にプログラミングの学習を位置付けることで、電気の「発電」・「蓄電」・「変換」・「制御」と一連の流れを崩さずに展開することができ、児童も発展的な学習

として抵抗なく受け入れることができていた。関連する学習内容と近接してプログラミングの体験を配置することに関しては効果があると思われる。

教科によっても扱うソフトや教材は異なり、それぞれのプログラミング教材に触れる時間は必要になってくる。また、児童がある程度プログラミングの知識を持ち、操作ができる状態で本時の授業を展開できることが望ましい。そのためには発達段階に合わせた教材を選択し、事前に操作に慣れさせておく必要がある。加えて学校全体の取組みとしては、年間指導計画の中に位置付けて計画的に実施することが必要である。

## カ 単元の目標との関連について

### (ア) 算数科の単元の目標との関連

単元の学習を通し、児童は平行四辺形の概念や特徴を学んできた。本時に至る前に、その概念や特徴を利用して三角定規やコンパスでの作図を行ってきた。プログラミング教材を利用した作図であっても、その本質は変わらない。図形の概念や特徴をコンピュータで作図する場合に置き換えて、作図に必要な辺の長さや角の大きさを見つけ、入力し、並べ替え、修正し、実行し、確かめることを取り入れてきた。プログラミングの一連の操作を通して、児童は平行四辺形の概念や特徴を使って作図する方法を自然と活用することができ、試行錯誤の操作の中にこそ論理的な思考が働いていたと考える。

しかし、コンピュータで作図する場合、「一筆書き」や「外角」で入力することが一般的であるため、これまで内角の学習に取り組んできた児童にとって、外角での作図に切り替えるのは難しかったようである。画面上での方向感覚に慣れるために、宝探しゲームなどで感覚を養ったり、角度の学習をする際に外角についても触れたりするなどの経験を積ませることが必要であったと感じた。

### (イ) 理科の単元の目標との関連

「買い物の手順」や「信号待ちの手順」など、日常生活の様子をフローチャート図を使ってプログラミングする活動を行ったことで、教材を使った光や音を制御する活動にスムーズに移ることができた。本時では自分で数値や条件を変え、新たなプログラムを作成する児童もみられ、より発展的な授業展開を行うことができる可能性を示すことができた。また、この自発的な活動こそプログラミングの体験の本質に迫るものと考えられる。今回このような児童は少数で、気付かれないうちに、こっそりと行っていた様子がうかがえるが、積極的に推奨することで次時のおもちゃ作りのアイデアなどにも発展させることができると考える。

プログラミングの体験で目指すもの、単元の目標に寄り添うことの両方の面で成果を出すことができた。

## Ⅷ 研究のまとめ

### 1 全体考察

本研究は小学校算数科・理科の学習指導における、論理的思考力を育むプログラミングの体験の在り方について明らかにすることであった。

論理的思考力は短時間で育成されるものではない。このような現状で「プログラミング的思考の5つの能力」の視点で算数科、理科における指導事例を示すことができたこと、学習ログの分



析やワークシートの活用の仕方等を示すことができたことは、これからプログラミング教育を行うための指針となったのではないだろうか。

さらに、児童から「順序に気を付けることが大切」、「筋道や順番に気を付ければ他の学習にも役立つ」という感想が得られたことは、児童が論理的思考を意識し、活用することができた表れではないかと考える。

## 2 研究の成果

- (1) 新学習指導要領に記載されている算数科、理科のプログラミングの体験において、授業の一例を示すことができた。
- (2) 「プログラミング的思考」を「コンピューショナル・シンキング」と関連付け、5つの能力に細分化したことにより、単位時間の中で「どの能力が必要なのか」が明確になり、活用すべき能力を具体的に示すことができた。
- (3) 「プログラミング的思考」の細分化は、活用する能力に合わせた指導計画の作成や、使用教材の選定、ワークシートの作成などに反映させることができた。
- (4) 学習ログは児童の思考過程を読み取るのに有効であることを示すことができた。しかし、学習ログの解析には膨大な時間を要するため、通常の授業での処理は難しいことから、学習ログの要素を包含したワークシートの活用の有効性へつなげることができた。

## 3 今後の課題

- (1) 「プログラミングの体験」に使用する教材の選定については、教科の目標や育成のねらいに沿ったものを選定するために実践例等の情報を共有することが必要である。
- (2) 学習活動が停滞している児童へは、学習内容がわからないのか、コンピュータの操作がわからないのかを確認した上で個別の指導が重要である。
- (3) 「プログラミングの体験」にあてる時数については、情報教育の一部として年間指導計画に位置付け、教科と関連付けながら計画的に実践する必要がある。

<おわりに>

長期研修の機会を与えて下さいました関係諸機関の各位並びに所属校の諸先生方と児童のみなさんに心から感謝申し上げます、結びの言葉といたします。

## IX 引用文献及び参考文献

### 【引用 Web ページ】

文部科学省 (2016), 『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](http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.htm)

文部科学省 (2014), 『諸外国におけるプログラミング教育に関する調査研究』

<http://www.educ.pref.fukuoka.jp/html/chosa/h22/guide-book.pdf>



文部科学省 (2017), 『小学校学習指導要領』

[http://www.mext.go.jp/component/a\\_menu/education/micro\\_detail/\\_icsFiles/afieldfile](http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile)

/2017/05/12/1384661\_4\_2.pdf

## 【参考文献】

- 安部和広 (2013), 『小学生からはじめるわくわくプログラミング』 日経 BP 社
- 安部和宏, 倉本大資(2016), 『小学生からはじめるわくわくプログラミング 2』 日経 BP 社
- 安藤明伸 (2017), 情報教育対応教員研修全国セミナー資料  
『どうする? 授業で取り組む「プログラミング」』
- 磯部秀司他 (2016), 『コンピュテーショナルシンキング』 共立出版
- 上松恵理子 (2016), 『小学校にプログラミングがやってきた! 超入門編』 三省堂
- エム・アール・アイ リサーチアソシエイツ株式会社 (2016),  
『「プログラミング教育」の実施状況に関する現状調査』
- 太田剛, 森本容介, 加藤浩 (2016), 『諸外国のプログラミング教育を含む情報カリキュラムに関する調査 -英国, オーストラリア, 米国を中心として-』
- 岡島裕史 (2015), 『こどもが楽しむ「プログラミン」入門』 技術評論社
- 風穴江, 神谷加代, 塩野禎隆, 合同会社デジタルポケット, 技術評論社編集部 (2015),  
『こどもプログラミング読本』 技術評論社
- 片桐茂男 (2004), 『数学的な考え方の具体化と指導』 明治図書
- (株) アンク (2016), 『プログラミングって何? 親子でゼロからわかる本』 秀和システム
- 株式会社バンタン 未来の仕事研究所 (2017),  
『まんがでわかる 親子で始めるプログラミング教育』 KADOKAWA
- 黒上晴夫・堀田龍也 (2017), 『導入の前に知っておきたい  
プログラミング教育 思考のアイデア』 小学館
- 国立教育政策研究所 (2013), 『社会の変化に対応する資質や能力を育成する  
教育課程編成の基本原理解』
- 小林祐紀 兼宗進 (2017), 『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” ではぐくむ論理的思考』 翔泳社
- 佐藤亮, 香山瑞恵, 國宗永佳, 伊藤一典, 橋本昌巳, 大谷真 (2013),  
『アルゴリズム的思考と基礎プログラミング力との関連性の検討』
- 杉野裕子 (2015), 『算数科におけるプログラミング活用授業のためにコンテンツ開発と改良』
- 杉野裕子 (2015), 『プログラミングを活用した図形概念形成についての研究』
- 総務省 (2015), 『プログラミング人材育成の在り方に関する調査研究報告書』
- 田中保成 (2008), 『消える学力, 消えない学力 算数で一生消えない論理的思考力を  
育てる方法』 ディスカバー・トゥエンティワン
- 田村学・黒上晴夫 (2013), 『考えるってこういうことか! 「思考ツール」の授業』 小学館
- 東洋館出版社編集部 (2017), 『平成 29 年度 小学校新学習指導要領ポイント総整理』  
東洋館出版社
- 利根川裕太・佐藤智 (2017), 『先生のための小学校プログラミング教育が  
よくわかる本』 株式会社シナノ

- 中嶋秀之：翻訳（2015），『計算論的思考』
- 中村好則（2016），『算数科におけるプログラミング的思考と数学的な見方・考え方の育成に関する考察-Sphero SPRK Edition を活用した「速さ」の指導事例を通して-』
- 新潟大学教育学部附属新潟小学校（2017），『ICT×思考ツールでつくる  
「主体的・対話的で深い学び」を促す授業』 小学館
- 西村陽介（2017），New Educational Expo 2017 資料  
『草津市のプログラミング教育の「これまで」と「これから」』
- 米田昌悟（2016），『プログラミング入門講座』 SBクリエイティブ株式会社
- 松田孝・吉田潤子他（2017），『小学校の「プログラミング授業」実況中継』 技術評論社
- 三好勝利・高橋純・堀田龍也・山西潤一（2015），  
『教科 Computing 実施後の英国の情報教育カリキュラムおよびテキストの特徴』
- 村上正行，Chengcheng HOU，飯山将晃，美濃導彦（2017），日本教育工学会第33回全国大会資料  
『数学の図形問題における解答に至る過程の可視化と分析』
- 安川要平（2016），『Scratchでつくる！たのしむ！プログラミング道場』 ソーテック社
- 吉田葵・安部和宏（2017），『はじめよう！プログラミング教育』 日本標準
- 渡邊茂一（2017），New Educational Expo 2017 資料  
『相模原市の教育課程における取組からの提案』
- リンダ・リウカス（2016），『ルビイのぼうけん こんにちはプログラミング』 翔泳社
- リンダ・リウカス（2016），『ルビイのぼうけん コンピューターの国のルビイ』 翔泳社
- Benesse Corporation(2017)，『プログラミングで育成する資質・能力の評価規準（試行版）』
- Jeannette M. Wing（2006），『Computational Thinking』
- Tim Bell, Ian H. Witten, Mike Fellows, 兼宗進[監訳]（2007），  
『コンピュータを使わない情報教育アンプラグドコンピュータサイエンス』 イーテキスト研究所

## 【参考 Web ページ】

- 意外と古い？ コンピュータの歴史【ルーツを含む】  
<https://matome.naver.jp/odai/2134094091635863401> (平成29年12月14日閲覧)
- 学習用翻訳「計算論的思考」  
<https://www.facebook.com/notes/1238337629547688> (平成29年12月14日閲覧)
- 韓国の情報教育 <https://www.ipsj.or.jp/magazine/9faeag000000gihw-att/5504-6.pdf>  
(平成29年12月14日閲覧)
- 【教職教養◆教育原理③】学習形態・指導法・学習課程  
<http://www.kyoushi.jp/entries/953> (平成29年12月14日閲覧)
- コンピューテーショナル・シンキング(computational thinking)とは  
<https://www.programming-edu.com/2017/05/02/computational-thinking/>  
(平成29年12月14日閲覧)
- 「コンピューテーショナル・シンキングとプログラミング的思考」再訪  
<https://blogs.yahoo.co.jp/toshishyun> (平成29年12月14日閲覧)
- コンピューテーショナル・シンキングについて  
<http://www.con3.com/rinlab/?p=1859> (平成29年12月14日閲覧)

コンピュータの歴史（年表）

[http://www.infonet.co.jp/ueyama/ip/history/history\\_ct.html#leg](http://www.infonet.co.jp/ueyama/ip/history/history_ct.html#leg)

（平成 29 年 12 月 14 日閲覧）

サイボウズ式 プログラミングって？

<https://cybozushiki.cybozu.co.jp/articles/m001226.html>

（平成 29 年 12 月 26 日閲覧）

品川区立京陽小学校 『プログラミング学習実践事例集』

<http://school.cts.ne.jp/data/open/cnt/3/956/1/keiyoprograming.pdf>

（平成 29 年 12 月 14 日閲覧）

小学校でプログラミングが必修化？教師や保護者の期待と不安

[https://careercarver.jp/doc/contents/pages/cs/articles/0019\\_programming.html](https://careercarver.jp/doc/contents/pages/cs/articles/0019_programming.html)

（平成 29 年 12 月 14 日閲覧）

情報教育/プログラミング教育はコンピューショナル・シンキングと

メイキングがキーワードみたいかな？

[http://beyondbb.jp/Materials/ProgEdu\\_Go0ta.pdf#search](http://beyondbb.jp/Materials/ProgEdu_Go0ta.pdf#search)

（平成29年12月14日閲覧）

総務省 『プログラミング教育に関する総務省の取組』

[http://www.soumu.go.jp/main\\_content/000477738.pdf](http://www.soumu.go.jp/main_content/000477738.pdf)

（平成 29 年 12 月 14 日閲覧）

武雄市教育委員会 HP 『ICT を活用した教育パンフレット』

<https://www.city.takeo.lg.jp/kyouiku/docs/20170324smileedu01.pdf>

（平成 29 年 12 月 14 日閲覧）

楽しく学べる 松田式プログラミング学習帳

<https://dynabook.com/solution/dynaschool/contents/matsudashiki-programming/vol1.html>

（平成 29 年 12 月 26 日閲覧）

新潟県立教育センター 『小学校段階におけるプログラミング教育の実施』

<http://www.nipec.nein.ed.jp/sc/project/h29/curri-mana/29karimane-8.pdf>

（平成 29 年 12 月 14 日閲覧）

福岡県教育センター 「『思考力・判断力・表現力』の評価と授業づくりガイドブック小学校編」

<http://www.educ.pref.fukuoka.jp/html/chosa/h22/guide-book.pdf>

（平成 29 年 12 月 14 日閲覧）

古河市立大和田小学校 『教科の中で行うプログラミング教育』

[http://owada.koga.ed.jp/ICT教育/?action=cabinet\\_action\\_main\\_download&block\\_id](http://owada.koga.ed.jp/ICT教育/?action=cabinet_action_main_download&block_id)

[=26&room\\_id=1&cabinet\\_id=2&file\\_id=29&upload\\_id=1376](http://owada.koga.ed.jp/ICT教育/?action=cabinet_action_main_download&block_id=26&room_id=1&cabinet_id=2&file_id=29&upload_id=1376)

（平成 29 年 12 月 14 日閲覧）

「プログラミング教育」小学校必修化を前に～克服すべき 3 つの難題

[http://www.huffingtonpost.jp/kei-hata/programming-education\\_b\\_11345214.html](http://www.huffingtonpost.jp/kei-hata/programming-education_b_11345214.html)

（平成 29 年 12 月 14 日閲覧）

未来の学びコンソーシアム

<http://miraino-manabi.jp/>

（平成 29 年 12 月 26 日閲覧）

文部科学省 『プログラミング教育実践ガイド』

[http://jouhouka.mext.go.jp/school/pdf/programing\\_guide.pdf](http://jouhouka.mext.go.jp/school/pdf/programing_guide.pdf)

（平成 29 年 12 月 14 日閲覧）

文部科学省 『情報ワーキンググループにおける審議の取りまとめ』

[http://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo3/059/sonota/\\_\\_icsFiles/afieldfile/2016/09/12/1377017\\_1.pdf](http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/059/sonota/__icsFiles/afieldfile/2016/09/12/1377017_1.pdf) (平成29年12月14日閲覧)

文部科学省『算数・数学ワーキンググループにおける審議の取りまとめ』

[http://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo3/073/sonota/\\_\\_icsFiles/afieldfile/2016/09/12/1376993.pdf](http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/073/sonota/__icsFiles/afieldfile/2016/09/12/1376993.pdf) (平成29年12月14日閲覧)

文部科学省『理科ワーキンググループにおける審議の取りまとめ』

[http://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo3/060/sonota/\\_\\_icsFiles/afieldfile/2016/09/12/1376994.pdf](http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/060/sonota/__icsFiles/afieldfile/2016/09/12/1376994.pdf) (平成29年12月14日閲覧)

文部科学省『幼稚園，小学校，中学校，高等学校及び特別支援学校の  
学習指導要領の改善及び必要な方策等について』

[http://www.mext.go.jp/b\\_menu/shingi/chukyo/chukyo0/toushin/\\_\\_icsFiles/afieldfile/2017/01/10/1380902\\_0.pdf](http://www.mext.go.jp/b_menu/shingi/chukyo/chukyo0/toushin/__icsFiles/afieldfile/2017/01/10/1380902_0.pdf) (平成29年12月14日閲覧)

文科省の言うプログラミング的思考

<http://www.anlyznews.com/2016/07/blog-post.html> (平成29年12月14日閲覧)