

小学校プログラミング教育の鍵概念の検討

—実践のための提言—

藤川 大祐

千葉大学教育学部

2020年度から全面実施される小学校学習指導要領では、初めてプログラミング教育が導入された。小学校プログラミング教育の鍵概念となっている「論理的思考力」や「プログラミング的思考」といった概念には、多義的であったり概念相互の関係が不明であったり等の問題が見られることが確認できる。こうした問題を反映して、小学校プログラミング教育に関する言説には混乱が見られ、今後、実践を進めるにあたって対応策が検討される必要がある。本稿では対応策として、「論理的思考力」を「論理的推論を行う力」と読み替えること、プログラミング教育が育成する「論理的思考力」は教科等においても育成される「論理的思考力」全般でなく「論理的思考力」の一部であると考え、こと、「プログラミング的思考」は「論理的思考」と「創造的思考」の両方を含むものと考え、こと、「プログラミング的思考」は時代を越えて普遍的に求められる力であると同時に社会の変化とともに意味合いが変化しているものと捉えることを提言した。

キーワード：プログラミング教育、論理的思考力、プログラミング的思考、Computing Thinking、Logical Reasoning

1. 鍵概念をめぐる混乱

1.1. 「論理的思考力」

2020年度から全面実施される小学校学習指導要領（文部科学省 2017a）では、初めてプログラミング教育が導入された。このたびの学習指導要領改訂では「人工知能（AI）の飛躍的な進化」（文部科学省 2017b）等の社会の変化に対応すべきことがふまえられており、直接的な記載はないもののプログラミング教育の導入もAIの進化等の変化への対応という面があることがうかがわれる。

学習指導要領は、プログラミング教育について、各教科等の特質に応じて、「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」を行うとしている。

だが、学習指導要領のこの説明は多義的であり、混乱を生じさせるものである。すなわち、プログラミング教育で身に付けさせるとされる「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」という表現は、次の二つの異なる意味に解される。

(A) 論理的思考力一般であり、この論理的思考力一般は、

コンピュータに意図した処理を行わせるために必要なものである。

(B) ある種の限定された論理的思考力であり、どのような種類かという、コンピュータに意図した処理を行わせるために必要な種類の論理的思考力である。

英語で書けば、それぞれ下記ようになる。

- (A) the logical thinking, which is required to make the computer perform the intended processing
 (B) the logical thinking which is required to make the computer perform the intended processing

両者の違いは、“the logical thinking”の直後のカンマの有無のみである。英文法用語で言えば、(A)では「関係代名詞の非制限用法」が使われており、関係代名詞 **which** 以降は **the logical thinking** の補足説明となっている。(B)では「関係代名詞の制限用法」が使われており、関係代名詞 **which** 以降は **the logical thinking** の意味を限定している。

この「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」をめぐる解釈の違いは、プログラミング教育の目的についての理解の違いに直接結びついてしまう。すなわち、(A)の解釈をとればプログラミング教育の目的は論理的思考力一般を育てることとなり、(B)の解釈をとればプログラミング教育の目的は

ある種の限定された論理的思考力を育てることだとなる。学習指導要領におけるプログラミング教育の目的に関わる鍵概念であるはずの「論理的思考力」の意味が、このように一意的に決まらない状況がある。

1.2. 「プログラミング的思考」

では、小学校プログラミング教育をめぐる言説では、プログラミング教育の目的はどのように解されているのだろうか。

論文検索サイト CiNii で「小学校」かつ「プログラミング教育」で検索を行うと、最近1年間（2019年1月～12月）発行の文献が103件ヒットする。これらのうち、学習指導要領が掲げる「論理的思考力」でもヒットする文献はわずか3件であり、すべてが文部科学省発行の『初等教育資料』の特集名（「論理的思考力等を育むためのプログラミング教育の在り方」）がヒットしたものである。「論理的思考力」を「論理的思考」としても、ヒットする文献は2件増えるだけだ。学習指導要領が掲げる「論理的思考力」という語は、小学校プログラミング教育に関する文献ではあまり取り上げられていない。

他方、学習指導要領解説（文部科学省 2017b）で強調されている語である「プログラミング的思考」でもヒットする文献は20件である。「プログラミング的思考」という語は学習指導要領自体には見られないが、小学校プログラミング教育が論じられる際には「論理的思考力」より「プログラミング的思考」のほうが顕著に多く取り上げられている。

学習指導要領解説は、プログラミング教育について以下のように述べている。

子供たちが将来どのような職業に就くとしても時代を越えて普遍的に求められる「プログラミング的思考」（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）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計画的に実施することとしている。

非常に長い一文であるが、要は、「プログラミング的思考」を育むため、「論理的思考力」を身に付けるための学習活動を実施するということが述べられている。ここでは「ため」という語が二重に使われており、素直に

読めば、「プログラミング的思考」と「論理的思考力」とは別のものであり、前者がより高次の目的であって、これを育むための手段として後者を育むものと解される。だが、「プログラミング的思考」やこれに類する表現は学習指導要領では使われておらず、学習指導要領解説が学習指導要領にない高次の目的を記すことは「解説」に許されない越権行為であるようにも思われる。また、「プログラミング的思考」の定義の中に「論理的に考えていく力」とあることから、「プログラミング的思考」と「論理的思考力」とが同一の概念を指しているという解釈もなされうる。「プログラミング的思考」と「論理的思考力」との関係についても、これを一意的に解釈することは困難である。

また、学習指導要領解説が「プログラミング的思考」について「時代を越えて普遍的に求められる」としていることについても、整合的な解釈が難しい。そもそもプログラミング教育はコンピュータの発展を背景に導入されるはずのものであり、前述のように学習指導要領解説でも「人工知能（AI）の飛躍的な進化」等を背景とした学習指導要領改訂であったことが示唆されている。たしかに複雑な物事を実現するために手順を整理して実行することは普遍的に求められると言えるだろうが、普遍的に求められるのであれば今回こうした力がことさらに強調されるのはなぜかを理解することが困難である。

以上のように、学習指導要領解説におけるプログラミング教育の説明は一意的、整合的に解釈することが難しい。なぜこのような説明がなされるようになったのだろうか。

学習指導要領解説における「プログラミング的思考」の説明は、文部科学省が設置した有識者会議¹の「議論の取りまとめ」（文部科学省 2016）が元になっている。この取りまとめ中の「学校教育として実施するプログラミング教育は何を目指すのか」という項目で、育成すべき資質・能力のうちの「思考力・判断力・表現力等」として、「プログラミング的思考」が挙げられており、プログラミング的思考の「定義」として、「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」と記されている。そして、この「定義」について、「いわゆる『コンピュテーショナル・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」という注が付されている。

こうした記述より、「プログラミング的思考」という概念が提案されるにあたっては、プログラミングと論理

的思考との関係が整理される必要があったと読み取ることができる。すなわち、プログラミングと論理的思考との関係は、整理が必要とされるほど複雑だとされていることになる。ただし、プログラミングと論理的思考との関係がどこでどのように整理されたのかは、有識者会議の議事録等から確認できない。

1.3. Computational Thinking との関係

では、この有識者会議では、プログラミングと論理的思考との関係をどのように整理したのだろうか。有識者会議の議事録をたどってみよう。

「プログラミング的思考」という表現が最初に出されたのは、2016年5月19日の第2回会議であった。この会議で、礮津政明委員(株式会社ソニー・グローバルエデュケーション代表取締役社長²⁾) が、次のように発言している³⁾。やや長いが、まとめて引用する。

(略)プログラミング教育の本質が何であるかという、御存じのとおり、コーディングというよりは、プログラミング的思考、先ほどもありましたが、コンピューターショナル・シンキングと言われているところがプログラミング教育の本質ではないかと考えておまして、この部分を強めるのが、プログラミング教育のあるべき姿だと感じているところです。

そのコンピューターショナル・シンキングなんですけれども、CTと海外では略されていますけれども、なかなか的確な日本語訳がないのですが、今日のところはプログラミング的思考ということでお話しさせていただきたいんですが、コンピューターショナル・シンキング自体は、コンピューターを活用して課題を解決する方法論ですね。決してコンピューターのように考えることではありません。海外ではかなり厳密に分類されていたりもするんですけれども、いろいろな分類がある中で、ここに挙げている色が付いている四つぐらいに大体収束するかなと思っていて、人によっては五つだったり六つだったりするんですけれども、大体四つ五つに分類されるようなものです。

例えば物事を分解するとか、問題の法則を発見するとか、問題の余計な部分を切り落として抽象化するとか、あるいは最後にアルゴリズムをデザインするとかといった、こういった考え方の方法論、そういったものをできるだけコンピューターを使えるような形で論理的に考えていこうというのが、このコンピューターショナル・シンキングで、これ、四つのコンセプト、五つのコンセプトみたいな形で呼ばれているものです。

この礮津委員の発言では、海外で Computational Thinking (以下、CT とする) とされる思考が「プログラミング的思考」とされている。礮津委員が「四つのコンセプト」あるいは「五つのコンセプト」と呼んでいるものが何か、議事録からは読み取れないが、CT のコンセプトとしていくつかの項目を挙げている文献は見られる。たとえば、CAS (2015) は、①アルゴリズム的に (algorithmically) 考える能力、②分解 (decomposition) の観点で考える能力、③一般化 (generalizations) して、すなわち同一性を見いだし (identifying) パターンを使って考える能力、④抽象化 (abstractions) して、すなわちよい表象 (representations) を使って考える能力、⑤評価 (evaluation) の観点で考える能力を含んでいるとしている。礮津委員の発言における「四つのコンセプト」「五つのコンセプト」とは、この種のことを指しているものと考えられる。

この直後の2016年5月27日、プログラミング教育に関する議論が、中央教育審議会初等中等教育分科会教育課程部会小学校部会(第6回)でなされた⁴⁾。この会議で、大杉教育課程企画室長より有識者会議での議論の状況について報告がなされている。大杉室長は、小学校におけるプログラミング教育について、「プログラミング的思考を育んでいく」、「物事を分解して考える、法則を発見する、抽象化する、解法をデザインする、評価するということをしっかりと行えるようにしていく」ことが求められているという議論を紹介している。大杉室長のこうした説明では、「プログラミング的思考」は CT と同義として扱われていると解される。

この後、部会の委員らからは、小学校教員の負担等、小学校にプログラミング教育を導入することへの懸念が示される。そして、そうした懸念を受けて、有識者会議の委員でもある無藤隆教育課程部会長(白梅学園大学子ども学部教授兼子ども学研究科長)が次のように発言している。

私個人の意見としては、特定の言語というのは中学校以降に任せて、小学校では、プログラミング的思考というのは、英語ではコンピューターショナルシンキングというそうですけれども、それは要するに、情報の流れで明示化し明確化しながら考える論理的思考といったものだと思いますけれども、それが明確になるようなものを各教科ごとにやっていくことなんだと思います。そのポイントは、幾つかの操作すべき記号、言語と、操作されるべき限定された現実というものの対応関係の中で考えるということだと思うんですね。

(略)

そういう意味で私は、現段階では、各教科ごとの教材あるいは単元開発の中でコンピューテーショナルな考え方というものを取り入れるという試みが、ふさわしいのではないかと。もちろん 2 回の発表の中で、いろいろな NPO その他の方々の努力の中で、小学生といえども、かなり才能を発揮している子供たちがいることは確かですが、それはむしろ課外活動その他で任せても十分できることではないかと、そんな印象を持ちました。

この無藤部会長の発言においても、「プログラミング的思考」は CT と同義として扱われている。なお、CT について「論理的思考」という言葉が使われ、各教科の中でこうした思考が扱われるべきことが示唆されていることについては、それぞれ別途検討したい。

この部会の議論を経て、最終回となる有識者会議第 3 回会議（2016 年 6 月 3 日開催）において、議論の取りまとめ案が審議された⁵。この案において、「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力としての『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」「『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」といったことが盛り込まれ、学習指導要領解説でも使われることとなる「プログラミング的思考」の「定義」も記された。

有識者会議第 3 回会議の議事録によれば、この会議で上記の取りまとめ案の方向性は各委員から支持されていることが読み取れる。ただし、兼宗進委員（大阪電気通信大学工学部教授）からは「プログラミング的思考」や CT について意味合いを確認する質問が出されており、これに対し、大杉室長からは「1 回目、2 回目の議論の中で、複数の先生方から、Computational Thinking の重要性を御発言いただいたことを踏まえて、それをプログラミング的思考という形で、しかも、小学校教育ということ踏まえて、少し表現を工夫しながら、御相談しながら置かせていただいているものでございます」という説明がなされ、堀田龍也主査（東北大学大学院情報科学研究科教授）からは「プログラミング的思考というものをどう定義するかというのは、非常に難しいところですよ」という発言もなされている。この段階においても、「プログラミング的思考」は CT の言い換えとして扱われていた一方で、定義が難しいという認識が示されていたことが確認できる。

こうした議論を経て、前出のように、2016 年 6 月 16 日に公表された有識者会議議論の取りまとめにおいては、「プログラミング的思考」に「いわゆる『コンピュ

テーショナル・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」という注が付されるに至る。有識者会議における議論では、「プログラミング的思考」は基本的に CT の言い換えとして扱われていたにもかかわらず、取りまとめ案において初めて、明示的に「プログラミング的思考」は CT そのものではなく新たに提言された概念として位置づけられたということになる。そして、阪東他（2017）や林（2018）が指摘するように、CT についての研究動向をどのように踏まえ、CT をどのように踏襲しようとしたのかについては言及されていないままである。

1.4. 「論理的」の使われ方

以上のような経緯で、小学校プログラミング教育をめぐる言説において重要な役割を担っている「プログラミング的思考」は、CT を踏まえつつも CT とは区別される概念として提言されるに至った。「プログラミング的思考」の「定義」は「プログラミングと論理的思考との関係を整理しながら」提言されたものとされているものの、誰がどこでどのように「整理」を行ったのかを、公表された資料から読み取ることはできない。

では、「プログラミング的思考」と CT とはどのように異なるものなのか。明確な説明を見つけることはできないが、「プログラミング的思考」の「定義」には「論理的に考えていく力」とあり、注には「論理的思考」とあることから、「論理的」であることを積極的に表現しているところに CT とは異なる「プログラミング的思考」という概念の特徴があると見ることができる。

では、この文脈において「論理的」とはどのようなことを意味しているのだろうか。

取りまとめを見ても学習指導要領解説を見ても、「論理的」とはどのようなことなのかについての明示的な説明を見いだすことはできない。そこで、有識者会議や教育課程部会小学校部会におけるプログラミング教育をめぐる議論の中で、「論理的」という語がどのように使われているかを見ていこう。

有識者会議第 1 回会議で大杉室長は、「言語の力で言えば、言葉の働きなどという知識・技能、また、そういったことを使いながら創造的思考、論理的思考、感性・情緒の側面」と述べている⁶。この段階では、汎用的な能力の一つとして「論理的思考」という語が使われていると言える。

有識者会議第 2 回を見ると、序盤で新井紀子委員（国立情報学研究所教授）が「論理的な説明文の読解力」と言語のあり方として「論理的」という語を用いており、大杉室長の発言に近い形で「論理的」という語が使われていると言える。

同じ第2回で、前出のように磯津委員がCTについて言及し、その中で「コンピューターを使えるような形で論理的に考えていこうというのが、このコンピューショナル・シンキングで」と、「論理的」という語を用いている。ここで初めて、コンピューターを使う文脈において「論理的」という語が使われたことになる。

また同じ第2回で、奈須正裕委員（上智大学総合人間科学部教授）が次のように、従来の教科等で扱われてきた「論理的思考力」と、プログラミング教育に関わる「論理的思考力」との関係を次のように述べている。

論理的思考力や創造力、問題を解決する能力というのは、在来の教科等でもやってきたことなわけですが、第4次産業革命が起こってくるとか、プログラミング教育ということが入ってくることによって、従来僕らがイメージしてきた論理的思考力や創造性、問題解決力ということのニュアンスとか強調点とか重要度が変わってきたり、その構造化の筋道が変わってきたりするということが、多分一番大事なのかなと思っています。

その後の教育課程部会小学校部会（第6回）では、大杉室長が「読解力でありますとか、数学的な思考力、創造性、問題解決能力、論理的思考力などなど、時代を超えて常に重要な力ということがあるわけでございます」と、「論理的思考力」をやはり汎用的な能力の一つとして述べている。こうした説明に対して、松川禮子委員（岐阜県教育委員会教育長）からは「社会の中でのプログラミングの役割だとか、それから論理的思考だとかという話になってくると、既存の教科の中、例えば小学校で言えば、算数だとかそういう中での論理的思考を教えるということと、どう関わるのか」という疑問が出されている。そして、前出のように無藤部会長は、「プログラミング的思考」とCAとを同等とした上で、「プログラミング的思考」について「情報の流れで明示化し明確化していながら考える論理的思考といったものだと思います」と、「論理的思考」という語を使って説明している。

有識者会議第3回会議では、兼宗委員が「コンピューターというのは、人間でもない、他人ですから、そこにやってほしいことを伝えるためには、やはり論理的に伝えなければいけない」とし、プログラミングという営み自体がコンピューターに「論理的に」伝える営みだとしている。

以上のように、「論理的」という語はさまざまな形で用いられている上に、「論理的」という語の意味があらためて確認されることはなかった。以下の点で、「論理的」という語の用いられ方に問題があると言える。

第一に、「論理的思考力」が、情報化を経ても変わらないものなのか否かについて、奈須委員からの指摘があったにもかかわらず検討がなされず、取りまとめにおいては変わらないものとして扱われているということがある。

第二に、「論理的」ということが、プログラミングの外の文脈とプログラミングあるいはプログラミング教育の中の文脈との両方で用いられているため、プログラミング教育を進めることが「論理的」であることを必然的に扱えるということなのか、プログラミング教育と「論理的」であることを組み合わせる必要があるのかがはっきりしなくなっている。

第三に、仮にプログラミングという営みの中に「論理的」ということが含まれるとしても、それが手順等を明示的に並べるといっているのか、コンピューターに明確に伝えるといっているのか、それとも他の意味合いがあるのかといったことがはっきりしない。

1.5. 「論理的思考」と logical reasoning

日本語における「論理的思考」や「論理的思考力」といった語は、素朴には英語における logical thinking と対応しているように思われる。では、CTに関わる議論で、logical thinking はどのように扱われているのであろうか。

CTに関わる議論において、logical thinking が扱われることはある。たとえば、応用数学の一領域である離散数学の教育におけるCTについて論じた Liu and Wang (2010) は、CTの内容として抽象的思考 (abstract thinking)、モデル思考 (modeling thinking)、構成的思考 (constructive thinking) とともに logical thinking を挙げている。Liu and Wang は logical thinking を「推論の一貫性 (reasoning consistency) を使って結論へと至る過程」と定義している。そして、logical thinking の中核あるいは基礎をなすのは、要素を鎖状につなげていく順序的な思考 (sequential thought) だと述べている。数学における定理の証明で、こうした段階を追った logical thinking が使われていると指摘されている。

他方、CTにおける logical thinking という概念の扱いは広すぎ、定義がよくなされていないという指摘もある。CTの定義について検討した Selby and Woollard (2014) は、CTの定義に含めるべき概念について検討した結果、①抽象的に考える能力、②分解という観点で考える能力、③アルゴリズム的に考える能力、④評価の観点で考える能力、⑤一般化して考える能力の5点を定義に含めるべきであり、logical thinking、問題解決、システム・デザイン等についてはCTの定義に含めるべ

きでないと結論づけている。

たしかに、日本語における「論理的思考」あるいは「論理的思考力」といった概念と同様に、英語における **logical thinking** についても安定した定義がなされておらず、それゆえ CT の議論について扱いが難しいと言えらるだろう。しかし、それでも、CT に関する議論において、**logical** という点を軽々に外してよいとは考えにくい。

前出の CAS (2015) では、CT の定義に **logical** という要素を含めてはいないものの、CT の定義とは別のレベルで、「CT は、**logical reasoning** を含む認知あるいは思考の過程であり、これによって問題が解決されたり人工物や手続きやシステムがよりよく理解されたりする」と述べ、**logical reasoning** という概念について論じている。**reasoning** とは日本語では「推論」にあたるもので、一定の事実や前提をもとに結論を導く営みだと解される。CAS では、**logical reasoning** を、明晰かつ厳密な思考を通して事実を分析しチェックすることによって、学習者が物事について理解をなすことを可能にするものだとし、予測を立てて検証したり結論を出したりするのに使われると説明している。すなわち、ここで言う **logical reasoning** とは、あるプログラムが意図した動きをするであろうということ、あるプログラムを動かした結果が予測と違ったのはある箇所にバグがあったからだということなどを、一定の事実や前提をもとに結論付ける推論の営みを指すものと解される。Liu and Wang (2010) においても **logical thinking** の説明に **reasoning** という語が用いられており、**logical thinking** を **logical reasoning** に置き換えて論ずることが可能であると思われる。

すでに見たように、日本におけるプログラミング教育をめぐる議論では、「論理的思考」あるいは「論理的思考力」をめぐる議論に混乱が見られた。CA に関する議論でも、**logical thinking** については広く使われていてよく定義されていないと指摘されている。たしかに、「思考」あるいは **thinking** の概念が意味することは広く、こうした語を用いてしまうと議論が混乱することは避けたいと考えられる。これに代わって、**logical reasoning** という概念を用いることにすれば、**reasoning** は「思考」あるいは **thinking** より狭く限定された意味をもつことから、混乱を避けることができるかもしれない。

そもそも「論理」(**logic**) という語は古代より、数学の公理に基づく論証やアリストテレスの三段論法など、一定の前提から一定の規則に基づいて結論を導く営みとして扱われてきた。このことを考えれば、論理的 (**logical**) であるとは、**logical reasoning** ができるとたとえ解することに無理はないはずである。「論理的思

考力」や **logical thinking** という意味の広すぎる表現でなく、**logical reasoning** やその訳語としての「論理的推論」という表現を用いることで、混乱が回避できる可能性がある。

2. 実践的な対応策の検討

2.1. プログラミング教育をめぐる言説の状況

プログラミング教育の鍵概念に混乱が生じていることを見てきたが、こうした混乱はその後のプログラミング教育をめぐる言説にどのように反映されており、今後のプログラミング教育の実践のためにどのような対応策が必要か、検討していこう。

新学習指導要領が公示されて以降のプログラミング教育をめぐる言説において、学習指導要領が掲げる「論理的思考力」や学習指導要領解説が掲げる「論理的に考えていく力」について、どのように論じられているだろうか。

赤堀 (2018) は、まさに「プログラミング教育における論理的な思考とは何か」という問いを掲げ、調査をもとに検討を行っている。赤堀の問題意識は、プログラミング教育のねらいとしての論理的思考と算数・数学や理科、国語などの教科で目指すものとの異同を明らかにすることにある。そして、大学生を対象に、2種類のプログラミングの問題と教科の問題を解かせ、両者の関係を見る調査を行っている。この結果、プログラミングの問題はその種類によって数学や理科と相関が見られるものと国語や社会との相関が見られるものに分かれ、かつ、プログラミングの問題 2 問の間には強い相関が見られた。このことから赤堀は、「プログラミング的思考は、他教科・領域の論理的思考と関連するような総合的な論理的思考ではないかと思われる」と結論づけている。赤堀は、プログラミング教育における論理的思考とは何かを直接問うのではなく、論理的思考を問題を解く際に発揮されるものとし、教科の問題を解く際に発揮される能力との異同を問うたことになる。言わば、論理的思考について操作的に定義して扱っていることとなる。

藤原他 (2019) は、論理的思考としてのプログラミング的思考について考察した上で、人はプログラミングをする際に、論理的思考だけを働かせているのではなく、「創造的思考」をも働かせていると指摘する。そして、試行錯誤をしながら新たな価値を生み出すことを「創造性の具体的な姿」として育成すべきだと言う。藤原他はこうした試行錯誤を重ねる営みを、「現象、道具、素材をいろいろと直接いじくりまわして遊ぶこと」としての「ティンカリング」(Wilkinson and Petrich 2015) と捉えることを提案する。藤原他は、プログラミング教育において論理的思考力のみをねらいとするのではなく、

「創造的思考」あるいは「ティンカリング」をもねらいとすべきだと主張していることとなる⁷。

民部田他 (2019) は、『『プログラミング的思考』とは論理的思考の一部でしかない』という説明をしつつ、小学校理科において「論理的思考としてのプログラミングの思考法を身に付ける学習活動」を行うとし、授業実践について報告している。この授業では、複数の水溶液の同定を行う課題が扱われており、フローチャートを作成する作業において「児童が論理的に順序立てて解決の方法を組み立てることができた」かどうかが問われている。ここでは、順序立てて問題解決を行うことが、「論理的思考」として扱われていると読み取ることができる。

長山 (2019) は、小学校音楽科の表現領域においてプログラミング活動を行わせた実践を報告している。この中で、「自らの思いや意図を実現させるための方法(プログラム)と、表現結果の因果関係を論理的に考えることには、音楽科としての学びと『プログラミング的思考』を育むことの両立を実現することが期待できよう」と述べられており、プログラムと表現結果の因果関係を考えることについて「論理的」という語が用いられている。このような長山の立場は、「論理的」ということを *logical reasoning* として扱うものと解すことができる。

以上のように、プログラミング教育をめぐる言説において、学習指導要領が掲げる「論理的思考力」について、操作的定義において扱ったり、このこととは別に「創造的思考力」の重要性が主張されたりする言説が見られるとともに、実践においては何をもち「論理的」とするかについては順序立てて問題解決を行うことや因果関係を考えることについて「論理的」という表現が用いられている等の状況が見られた。これらは、「論理的思考力」自体の定義には踏み込んではいない。

以上検討してきたように、学習指導要領や学習指導要領解説が定める小学校プログラミング教育に関しては、以下のような問題が解決せずに残されているということになる。

- a. プログラミング教育がねらいとする「論理的思考力」とは何か。
- b. プログラミング教育がねらいとする「論理的思考力」は、「論理的思考力」一般を意味するのか、ある種の限定された「論理的思考力」を意味するのか。
- c. プログラミング教育がねらいとする「論理的思考力」は、従前の教科で育てられるとされてきた能力と同じものか、異なるものか。
- d. プログラミング教育がねらいとする「論理的思考力」と、学習指導要領解説で言う「プログラミング的思考」とは、どのような関係にあるのか。
- e. 学習指導要領解説で言う「プログラミング的思考」

は、時代を越えて普遍的に求められるとされているが、時代の変化とともにその意味合いは全く変わらないものなのか。だとしたら、「人工知能 (AI) の飛躍的な進化」と小学校へのプログラミング教育の導入は無関係ということなのか。

2.2. 実践的な対応策の提案

上の a~e の問題は、小学校においてプログラミング教育を実践する際に、教員らが直面する問題だと考えられる。「論理的思考力」や「プログラミング的思考」は、プログラミング教育を進める上で鍵となる概念である。教員らが実践を進める際に、こうした鍵となる概念の意味を問いたくなることは当然である。しかし、現状では、学習指導要領にも学習指導要領解説にもこうした問いへの答えは見出せない。

こうした状況をふまえれば、各教員が上記の問題について、一定の答えをもつことが必要だと言える。その答えは、暫定的なものでもかまわないであろう。実践を重ねる中で、答えがより洗練されたものとなっていくことも考えられる。

では、暫定的な答えをどのようにして導いたらよieldろうか。本稿で見えてきたような、プログラミング教育に関する検討の過程やプログラミング教育に関する諸元説を手掛かりにして、それぞれの問題への答えが互いに整合的であるように答えを構築するしかない。以下、そうした方向で検討を行おう。

まず、a についてだが、「論理的思考力」という語の意味は広すぎ、この語を厳密に定義することは難しいことが確認されている。そこで、CAS (2015) にならい、*logical reasoning* の概念を活用し、「論理的思考力」を「論理的推論」を行う力と読み替えて活用することが考えられる。「論理的推論」は、一定の事実や前提をもとに緻密に結論を導くこととしておけば、実践に適用するのに十分であろう。自分たちの知識をもとにあるプログラムが意図した動きをするはずだという結論を導いたり、実験の結果をもとにあるプログラムのどこが意図と違う動きをもたらしたのかという結論を導いたりすることが、論理的推論である。

次に、b 及び c についてであるが、プログラミング教育が扱うのは限定された「論理的思考力」であるという立場をとることが実践を進める上では妥当であろう。プログラミング教育においては、あくまでもプログラムの構築に関わる論理的推論を扱うのであり、従前の教科で扱われるさまざまな「論理的思考」をすべて扱えると言え根拠はない。赤堀 (2018) によって、プログラミング教育が扱う「論理的思考」は教科で扱われるものとの間に共通性があることは示されているが、それでも、プログラミング教育が教科で扱う「論理的思考」のすべ

てを扱えるかについては結論が出ていない。無用な混乱を避けるために、プログラミング教育が扱う「論理的思考」は限定されたものであり、教科で育てられるものとは重なりはあっても区別されるものだという立場をとるのが妥当だ。

d については、藤原他（2019）が指摘するように、プログラミング教育において「創造的思考」も扱うという立場をとり、「プログラミング的思考」には「論理的思考」だけでなく「創造的思考」も含まれるという立場をとるのが適切であろう。実践を行うと、「論理的思考」を相対的に重視した実践も、「創造的思考」を相対的に重視した実践もどちらもなされうると考えられる。学習指導要領のみを見れば前者の実践が求められていると解されるが、今後のプログラミング教育実践の研究の発展を考えると、少なくとも現段階で「創造的思考」を相対的に重視した実践が否定されるべきではない。「プログラミング的思考」には「論理的思考」だけでなく「創造的思考」も含まれるという立場をとり、「論理的思考」を扱いつつも「創造的思考」を重視した実践が研究される道を残しておくべきであろう。

e については、有識者会議における奈須委員の発言をふまえ、「プログラミング的思考」は時代を越えて普遍的に求められるものであるが、コンピュータの発達等の社会の変化とともにその意味合いも変化しているものとして捉えるという立場がとられるべきであろう。コンピュータの発達以前にも、人間や機械を段取りよく動かすよう思考することが求められてきたことは間違いなく、こうした思考を「プログラミング的思考」とみなしてもよいだろう。だが、コンピュータはデータの処理が人間や従来の機械と比べものにならないほど早く、大量のデータを効率よく処理したり、同様の作業を高速に膨大な回数繰り返したり、複雑な作業を自動で処理したり、離れた場所からデータを集めたりといったことができる。人間や従前の機会を動かす思考とコンピュータを動かす思考との間には異同があるということを前提にし、コンピュータの特性を理解してコンピュータを動かす活動をさせるという視点が必要となるであろう。

以上をまとめると、次のことを提言することとなる。

- ・ プログラミング教育のねらいにある「論理的思考力」は、「論理的推論を行う力」と読み替える。「論理的推論」とは、一定の事実や前提をもとに緻密に結論を導くことである。
- ・ プログラミング教育が育成する「論理的思考力」は、教科等においても育成される「論理的思考力」全般でなく、「論理的思考力」の一部であると考えられる。
- ・ 「プログラミング的思考」は、「論理的思考」と「創

造的思考」の両方を含むものとし、実践における両者のバランスは多様にありうるものとする。

- ・ 「プログラミング的思考」は時代を越えて普遍的に求められる力であると同時に、コンピュータの発達等、社会の変化とともに意味合いが変化しているものと捉える。

3. 終わりに

プログラミング教育は、AI の進化等の社会の変化に対応した教育として重要であると考えられる。また、意図する活動について一点の曇りもなくプログラムを記述する活動は、送り手の立場でコミュニケーション能力を高める営みとして、大いなる可能性をもつものと言える。

しかしながら、このたびの小学校へのプログラミング教育においては、鍵となる「論理的思考力」や「プログラミング的思考」の規定が曖昧であり、このまま広くプログラミング教育が実践されれば混乱が生ずることは必至である。

本稿では、暫定的なものであるかもしれないが、これまでの言説を展望して問題を整理し、実践を進めるために必要な対応策を提案した。本稿での提案が活かされることを願っている。

¹ 正式名称は、小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議。

² 会議での発言に関わる個人の肩書きは、当時のもの。他も同様。

³ 小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議（第2回）議事録

https://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/gijiroku/1382200.htm（2019年12月22日最終確認）

⁴ 教育課程部会 小学校部会（第6回）議事録

https://www.mext.go.jp/b_menu/shingi/chukyo/chukyo3/074/siryu/1382091.htm（2019年12月22日最終確認）

⁵ 小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議（第3回）議事録

https://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/gijiroku/1382219.htm（2019年12月22日最終確認）

⁶ 小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議（第1回）議事録

https://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/gijiroku/1382201.htm（2019年12月22日最終確認）

⁷ 「ティンカリング」（tinkering）という語には、上手でない修理職人がいじり回すという意味合いがあり、コンピュータでプログラミングするより実際の物をいじることが想像されるように思われる。このため、プログラミングの創造的な面を強調する際にこの「ティンカリング」という表現が適切かどうかについては検討が必要と考えられる。

引用文献

- 赤堀侃司 (2018) 「プログラミング教育における論理的な思考とは何か」、学習情報研究、第 261 巻第 4 号、pp.56-61
- 阪東徹也・黒田昌克・福井昌則・森山潤 (2017) 「我が国の初等中等教育におけるプログラミング教育の制度化に関する批判的検討」、兵庫教育大学学校教育学研究、第 30 巻、pp.173-184
- Computing At School (CAS) (2015) , Computational Thinking A guide for teachers.
<https://community.computingatschool.org.uk/resources/2324/single> (2019 年 12 月 22 日最終確認)
- 藤原伸彦・阪東徹也・曾根直人・長野仁志・山田哲也・伊藤陽介「ティンカリングとしてのプログラミング」、鳴門教育大学情報教育ジャーナル、No.16、pp.21-26
- Liu, J. and Wang, L. (2010) “Computational Thinking in Discrete Mathematics”, 2010 Second International Workshop on Education Technology and Computer Science, pp.413-416
- 民部田悟・植草創太・福田陽・沼倉徹 (2019) 「プログラミング的思考を活用した水溶液の同定：小学校 6 学年理科の授業実践を通して」、千葉敬愛短期大学紀要、第 41 巻、pp.47-62
- 文部科学省 (初等中等教育局教育課程課教育課程企画室) (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)」
https://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/houkoku/1372522.htm (2019 年 12 月 22 日最終確認)
- 文部科学省 (2017a) 『小学校学習指導要領』
- 文部科学省 (2017b) 『小学校学習指導要領解説 総則編』
- 長山弘 (2019) 「小学校音楽科におけるプログラミング教育のあり方の検討：授業実践事例を手がかりに」、初等教育カリキュラム研究、第 7 巻、pp.55-67
- 林向達「Computational Thinking に関する言説の動向」、日本教育工学会研究報告集、第 18 巻第 2 号、pp.165-172
- Selby, C. C. and Woollard, J. (2014) “Computational Thinking: The Developing Definition”, In Proceedings of the 45th ACM Technical Symposium on Computer Science Education, SIGCSE 2014. ACM
- Wilkinson, K. and Petrich, M. (2014) “The Art of Tinkering”, Weldon Owen. (金井哲夫 (訳) 『ティンカリングをはじめようーアート, サイエンス, テクノロジーの交差点で作って遊ぶ (Make:Japan Books)』オライリー・ジャパン、2015)