

プログラミング的思考を育成する授業の在り方（第二年次）

－言語化による再現性を重視したプログラミング体験を通して－

長期研究員 加藤 政 記

《研究の要旨》

「プログラミング的思考」の育成は、小学校段階におけるプログラミング教育の中核である。第二年次となる本研究では、各教科等の指導を通じた「プログラミング的思考」の育成をねらいとした。そこで、各教科等において、問題を解決するまでの思考の筋道を言語化させ、言語化した解決方法に再現性があるかを検討する場を設定し、各教科等の目標と「プログラミング的思考」の育成の二つを達成する授業実践を行った。

I 研究の趣旨

「小学校段階におけるプログラミング教育の在り方について（議論の取りまとめ）」（平成28年6月）の中で、「プログラミング的思考」は、コンピューショナル・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された。コンピューショナル・シンキングは、Wing（2006）の「Computational Thinking」（以下、CT）で定義されたものである。CTを教科に取り入れているのがイギリスである。イギリスの教科コンピューティングでは、CTの概念※1を〈抽象化〉〈分解〉〈アルゴリズム的思考〉〈評価〉〈一般化〉としている。本研究では、CTの概念をプログラミング的思考に求められる力とし、各教科等の指導を通して育成しようと考えた。

第一年次は、「プログラミング的思考」が育成された児童の姿を「問題を解決し、解決方法を言語化できる」姿ととらえ、実践した。算数科において、プログラミング体験が、「プログラミング的思考」の育成と、数学的な資質・能力の育成の両方に有効であることが確認できた。課題は、基礎的読解力※2が不足している児童が、コンピュータを使ったプログラミング体験に取り組んだ際、ビジュアルコンピュータ言語の意味を理解できず、当てずっぽうな試行錯誤に陥ってしまったことである。

第二年次は、算数科以外の教科の指導を通して、根拠に基づいた試行錯誤を重ねることで五つの力の育成を図り、教科等の目標と「プログラミング的思考」の育成の両方を達成する授業を目指すこととした。なお、本研究において、プログラミング体験とは、コンピュータを使わないアンプラグド体験も含むこととする。

II 研究の概要

1 研究仮説

プログラミング体験を取り入れた各教科等の指導において、以下の手だてを講じれば、各教科等で身に付ける

知識及び技能等の定着が図られ、「プログラミング的思考」が育成されるであろう。

【手だて1】思考の筋道を言語化する活動

【手だて2】再現性を検討する対話的な学び

※1 英国の教科コンピューティングでのコンピューショナル・シンキングの概念（太田剛，他 2016）

概念	概要
抽象化	問題を単純化するため、重要な部分は残し、不要な詳細は削除する。
デコンポジション（分解）	問題や事象をいくつかの部分に、理解や解決できるように分解する。
アルゴリズム的思考	問題を解決するための明確な手順で、同様の問題に共通して利用できるものである。
評価	アルゴリズム、システムや手順などの解決方法が正しいか、確認する過程である。
一般化	類似性からパターンを見つけて、それを予測、規則の作成、問題解決に使用する。

※2 基礎的読解力の調査には、教育のための科学研究所で開発したリーディングスキルテストを活用した。

2 研究の内容と実際

(1) 授業実践における手だてについて

基礎的読解力調査の結果を踏まえ、児童が、根拠に基づいた試行錯誤により、自分の意図した一連の活動を実現することができるようにするために、以下の手だてを講じた。

① 【手だて1】思考の筋道を言語化する活動

「プログラミング的思考」の育成に必要な五つの力のうち、〈抽象化〉〈分解〉〈アルゴリズム的思考〉の能力を高めるために、問題解決の手順を言語化させる。教科の特質によって言語化する方法は異なるが、問題解決の過程では、仕様※3を読み解き、解決すべき問題を〈抽象化〉してとらえ、問題解決の手順を細かく〈分解〉し、〈アルゴリズム的思考〉によって順序立てて整理することが求められる。そこで、授業の中で、問題を解決するまでの思考の筋道を言語化する場面を設定することとした。

※3 本研究では、各教科等の学習で、問題として提示される場面でのルールや定義など、問題解決までに満たさなければならぬ条件ととらえる。

② 【手だて2】再現性を検討する対話的な学び

プログラミングでは、仕様に基づき「誰がやっても同じ結果になる」という再現性が求められる。「プログラミング的思考」の育成に必要な五つの力のうち、〈評価〉（一般化）の能力を高めるために、問題解決までの思考の筋道を言語化したものに、仕様に基づいた再現性があるかを検討させることとした。児童は、再現性を検討する過程で、思考の筋道を言語化したものを、自分や他人との対話で〈評価〉し修正を加えていく。再現性の有無を視点に修正させることで、根拠に基づいた試行錯誤が繰り返され、〈一般化〉の力を高めることができると考えた。

(2) 授業の実際と考察

対象児童	第4学年（21名）、第5学年（34名） 第6学年（41名）
授業実践	第4学年 社会科「地図記号クイズ」 国語科「文と文をつなぐ言葉」 第5学年 家庭科「整理・整とんの工夫」 総合的な学習の時間 「情報化の進展と社会の変化」 第6学年 家庭科「いためて作る朝食」

実践は上記のとおりであるが、この中から第5学年家庭科の実践を紹介する。

授業実践 第5学年「整理・整とんの工夫」（4時間）

① 単元について

本単元は、気持ちよく生活するために、住まいの整理・整頓や清掃が必要であることが分かり、身の回りの整理・整頓や清掃の仕方を理解し、適切にできるようにすることが目的である。児童の身の回りの題材として、学習用具を入れる道具箱を取り上げ、整理・整頓する過程での試行錯誤を通して、「プログラミング的思考」の育成を図った。道具箱を整理・整頓する際には、学習用具の使用目的や頻度、大きさや形等によって工夫することが求められる。何をどこに置くと一目瞭然か、必要な物がすぐに取り出せるか、空間を有効に使えるかといった視点から考える。使いやすい道具箱にするために、学習用具を使用頻度に応じて、取り出しやすいように道具箱に配置する。児童は、使用する学習用具の数や大きさ、道具箱の大きさなどの制約の中で、試行錯誤を繰り返して整理・整頓に取り組む。整理・整頓するプログラムを考え、それに基づいて取り組み、生活習慣に応じて修正していく。生活の改善を目指した、アンプラグドプログラミング教材ととらえた。

② 授業の実際と考察

第1時では、「なぜ整理・整頓が必要なのか」を考えさせた。児童からは、「気持ちよく生活するため」「使いたいときにすぐに使えるようにするため」「無駄な買い物を

しないようにするため」といった意見が出された。整理・整頓の目的を共通理解し、自分の道具箱が使いやすいか考えているか考えた。すると、「使いづらい」「何が入っているか分からない」「不必要なものがある」と意見が出た（図1）。そこで、自分の考えた方法で、道具箱を整理・

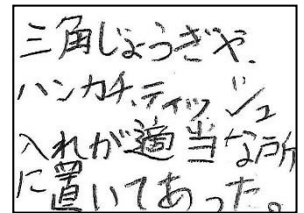


図1 整理・整頓する前

整頓させた。児童は、試行錯誤して整理・整頓に取り組んだ。児童を見ると、整理・整頓の仕方を筋道立てて考えながら試行錯誤する児童と、学習用具の大きさだけに着目し、とりあえず道具箱に収納することだけ考えて試行錯誤する児童に分かれた。授業のめあて（仕様）を踏まえた根拠に基づく試行錯誤と、授業のめあて（仕様）とは関係ない当てずっぽうな試行錯誤との違いと考えられる。「プログラミング的思考」の育成には、根拠に基づく試行錯誤が求められる。根拠に基づいた試行錯誤を全員に確保するために、あえて、第1時では、自分の考えた方法で整理・整頓に取り組ませた。

第2時は、整理・整頓の仕方を言語化した【手だて1】。児童は、1週間後に第2時を迎えた。1週間を経て、児童の整理・整頓の結果に差が現れた。全員の道具箱を見ると、整理・整頓された状態が継続していた児童と、もとの使いづらい状態に戻っている児童に分かれていた。つまり、整理・整頓の方法に再現性がある児童と、そうでない児童とに差が出ていた。それぞれの道具箱を見た後で、本時のめあてを「整理・整頓された状態を続けるためには、どうすればよいか」と設定した。児童は、1週間後の道具箱の状態になぜ差が出たのかを話し合った〈評価〉。すると、整理・整頓された状態が継続している児童に、共通点があることに気付いた〈抽象化〉。整理・整頓された状態が続いている児童に、どのようにして整理・整頓したのかを説明させ、全体で共有した。児童は、「必要な物と、不必要なものに分け、必要な物の中でも、よく使う物は、取り出しやすい所に整頓した」と述べた。児童の話を受け、学級全体で、整理・整頓の手順を小さなステップに分け〈分解〉、順序立てて並べ〈アルゴリズム的思考〉、図に整理していった（図2）【手だて1】。これは、整理・整頓するためのプログラムといえる。「よく使う物は道具箱の手前に置き、あまり使わない物は奥に置くように整頓する」という工夫も考えた。さらに、整頓するために、空き箱を

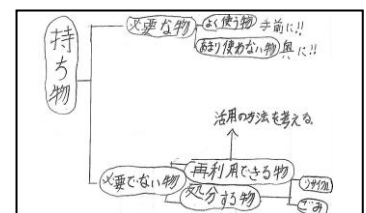


図2 整理・整頓の手順

活用しようという意見が出された。話し合いの結果、本時のまとめを、「よく使う物とあまり使わない物を、手前と奥に分けて整頓し、片付ける場所を決めておく」とした。これが新たな仕様となる。児童には、家庭にある空き箱を、次の授業までに学校へ持ってくるように伝えた。

第3時・第4時は、図2のプログラムに従って、道具箱を整理・整頓させた。1時目とは違い、全員が根拠に基づいて試行錯誤し、整理・整頓に取り組んだ。まず、児童は、道具箱の中身を必要な物と不必要な物に分けた。その後、必要な物をよく使う物とあまり使わない物に分類し〈抽象化〉、空き箱を活用して整頓した。次に、学習用具の形や大きさに合わせて、空き箱を選んでいった。

児童は、道具箱の大きさにちょうどよく収まるように、試行錯誤〈評価〉して選んでいた。自分が思い描いた道具箱の状態になるように、空き箱を選ぶ姿が見られた(図3)。よく使う物は、道具箱の手前に、あまり使わない物は、道具箱の奥に整頓した(図4)。試行錯誤を通し、「プログラミング的思考」を働かせて問題を解決していく姿が見られた。

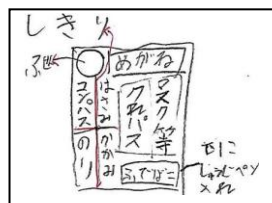


図3 道具箱の整頓計画



図4 整頓後の道具箱

道具箱を整理・整頓した後、一人一人に再現性があるかどうかを検討していった【手だて2】。その際に、仕様に合っているかをチェックさせた。チェックすべき仕様は、「よく使う物とあまり使わない物を分けて整頓し、片付ける場所を決めておく」ことである。児童Aは、「よく使う物を上に、あまり使わない物を下に重ねて置く」という工夫をしていた(図5)。

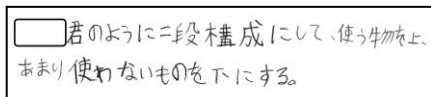


図5 再現性の検討

児童が、整頓する物に応じて、プログラムを修正して取り組んだ結果である〈評価〉。また、片付ける場所を決めておくために、「整頓する場所に何が入っているかを書く」という工夫をする児童もいた。これらの工夫は、根拠に基づいた試行錯誤の結果である。再現性を検討した結果を集約し、単元のまとめを「整理・整頓された状態を続けるためには、よく使う物とあまり使わない物を取り出しやすいように分けて整頓し、片付ける場所を決めて表記しておく」とした〈一般化〉。単元を通して、【手だて1】と【手だて2】により、全員が根拠に基づいた試行錯誤

に取り組み、五つの力を使って問題を解決することができた。

III 研究のまとめ

1 研究の成果

(1) 把持テストの結果

実践後に、教科書会社による指導書の添付テストを事後テストとして行った。「知識・理解」に関する問題は、正答率が92%だった。また、把持テストを実施し、定着率は90%だった。根拠に基づいた試行錯誤によって、獲得した知識の定着が見られた。教師が、「プログラミング的思考」に求められる五つの力を意識した授業を行うことで、教科の目標が達成できると確認できた。

(2) 「プログラミング的思考」について

「プログラミング的思考」(図6)が育成されたかを把握するために、第4時に、道具箱の整理・整頓の仕方を記述させた。

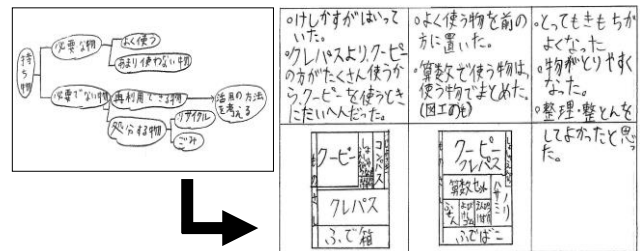


図6 プログラミング的思考の現れ

問題を〈抽象化〉して解決すべき課題を見だし、解決過程を〈分解〉してとらえ、順序立てて整理し〈アルゴリズム的思考〉、〈評価〉した結果を基に試行錯誤して解決する。さらに、再現性を確保〈一般化〉して言語化する。これらの過程が記述できているかを検証した。「問題を解決し、解決方法を言語化できた」児童は、70%に達した。

2 今後の課題

本研究では、アンプラグドプログラミング体験によって育成された「プログラミング的思考」が、コンピュータを活用したプログラミング体験において、どのように関連するのかまでは明らかにすることができなかった。また、地域や民間等と連携し、学校外の人的・物的資源等の教育資源を活用した授業についても課題が残った。しかし、カリキュラム・マネジメントを通じてプログラミング教育を実施するに当たっては、アンプラグドプログラミング体験と、コンピュータを使ったプログラミング体験の、それぞれのよさを理解することが必要である。今後は、各教科の指導を通して育成された「プログラミング的思考」を生かしたプログラミング体験に取り組ませることができる実践を目指していきたい。