

[教育方法一般]

# 各教科の学習の助けとなるアンプラグドプログラミング学習の試み - 複雑な手順を論理的に操作する活動を通して -

武川 亮平\*

## 1 主題設定の理由

小学校におけるプログラミング教育が2020年度から必修化された。文部科学省『小学校プログラミング教育の手引き第三版』によると、小学校におけるプログラミング教育のねらいは、次の通りである（表1）。

表1 小学校におけるプログラミング教育のねらい

- |   |
|---|
| <ul style="list-style-type: none"><li>①「プログラミング的思考を」を育むこと。</li><li>②プログラムの働きや良さ、情報社会がコンピュータ等の情報技術によって支えられていることなどに気付くことができるようにするとともに、コンピュータ等を上手に活用して身近な問題を解決したり、よりよい社会を築いたりしようとする態度を育むこと。</li><li>③各教科等の内容を指導する中で実施する場合には、各教科等での学びをより確実なものとする。</li></ul> |
|---|

つまり、プログラミングの技能を習得すること自体がねらいなのではない。コンピュータ等でプログラミングを行う場合、どうしてもある程度のプログラミングの技能が児童には求められる。例えばICTを活用した教育の例として、正多角形の作図（小5）の学習でスクラッチを活用することが小学校プログラミング教育の手引きには挙げられている。これは図形の作図方法を順序立てて理解することが期待できるが、少なからずスクラッチを扱うプログラミングの技能が求められることも事実である。そのため、一番のねらいである図形の作図方法の習得達成のために、プログラミングの技能を習得しなければならない。本来教科の学びをより確実にするためのプログラミング教育のはずが、プログラミングの技能の習得自体に重点を置かざるを得ないことになる。

また、プログラミング教育の課題の一つとして、金子は、「大部分の教員がプログラミングの授業を受けたことがなく、初めて聞く用語も多いため、指導者の立場としてそれを習得することは非常に困難である」（金子. 2019）と述べている。ICTを活用したプログラミング学習を指導できる教員自体が不足しているのである。

しかし、プログラミング教育は、必ずしもICTを活用しなければならないということはない。田中らは、「児童のプログラミング的思考を育成するには、日常的に授業でプログラミング的思考に触れることが必要であり、アンプラグドな授業も適切に織り交ぜて実施しなければならない」（田中. 中田. 2020.3）と述べている。アンプラグドプログラミングであれば、児童がコンピュータ等を操作するプログラミングの技能は必要ない。また、教員のICTに関する指導力や各学校のICT環境による壁が生じることもないのである。

さらに、アンプラグドであることにより、学習形態の工夫も期待できる。紙ベースをはじめとするアンプラグド形式のプログラミングツールでは、自由に移動しながら学ぶことができたり、複数の児童で同時に操作することができたりする利点が挙げられる。

これらの課題や背景から、本研究ではアンプラグドのプログラミング教育に焦点を当て、アンプラグド形式のプログラミング教育を教科に取り入れて指導することの有効性を検証する。

## 2 研究の目的

本研究では、アンプラグド形式のプログラミング教育を教科に取り入れて指導することの有効性について検証することを目的とする。

\*見附市立葛巻小学校

### 3 プログラミング的思考を促す単元と視覚化ツール

#### (1) プログラミングを用いることが学習の助けとなる単元の選択

プログラミングのメリットは、複雑な手順を視覚的に整理できることにある。本研究では、手順が複雑で児童にとって頭の中で整理することが難しいと考えられる学習場面で、アンブラグドのプログラミング教育を取り入れる。実際には、5年生算数「分数のたし算とひき算（学校図書）」及び4年生外国語「This is my favorite place.（啓林館）」の2つの単元を取り扱う。それぞれ、複雑な手順を視覚化して整理することで、児童の理解を深めることをねらいとする。

#### (2) 手順を視覚化するツールの活用

複雑な手順を視覚化する方法として、5年生算数「分数のたし算とひき算」の計算手順はフローチャート、4年生外国語「This is my favorite place.」の道案内の手順はナビゲーションプログラムをそれぞれ用いる。どちらも児童の思考を可視化することができる。

### 4 指導の実際と考察

#### (1) 5年生算数「分数のたし算とひき算」の実践

以下は、5年生算数「分数のたし算とひき算」の6時間目及び7時間目で、異分母分数のたし算の手順をフローチャートに表して考える授業である。授業のねらいは、「異分母分数のたし算の計算手順を、自分に合ったフローチャートで表す」及び「作成したフローチャートを用い、あらゆる難易度の分数のたし算に対応することができる」である。

##### ① 単元について

本単元では、分数の加法・減法の計算の仕方を習得する。4年次には同分母分数の加法・減法を学習したのに対し、今回は異分母分数の加法・減法を学習する。そのため通分や約分といった計算の処理が求められ、計算のプロセスも複雑化する。そこで物事を論理的に整理して考えることのできるフローチャートを用い、計算手順をプログラミング教育として学習することを設定する。今回のプログラミングは順次処理と分岐処理から成り立っている。分数の加法・減法も「まず2つの分母が同じであれば分子を加減し、異なる場合は通分をする」といった順次処理と分岐処理の連続で答えに辿り着く。通分や約分、帯分数や仮分数の処理を整理し、フローチャートに視覚的に表すことで、これまで学習してきた様々なパターンの計算に対して論理的に考えて対応する力を高める。

教科書では、約分や帯分数、繰り上がりの有無等によって計算の問題が難易度ごとに分かれている。しかし実際には、様々な難易度の問題に応じて、必要な計算処理を自ら判断しなければならない。児童がフローチャートの仕組みを理解し、あらゆる問題が一つのフローチャートで処理できるという利便性に気付けるようにする。

また、児童は一人一人の思考回路や処理能力が異なるため、フローチャートの正解が一つとは限らない。例えば「最小公倍数を求める」という処理を瞬時にできる児童は、その一つをブロックとするが、瞬時にできない児童は「片方の倍数を書き出して考える」というもう一つステップが必要になってくる。細かすぎてもかえって混乱してしまうことになりかねないので、どこまで細かくフローチャートに表すのか、児童一人一人に合ったものが作れるよう考慮する。フローチャートの分岐を用い、児童が自分に合った手順で計算できるという点もプログラミングの良さだと考える。

##### ② 児童の実態（男子17名 女子12名 計29名）

本学級の児童は、算数に得意意識をもち意欲的に取り組む児童、算数に苦手意識をもち消極的になってしまう児童の二極化が見られる。計算の処理速度も大きく差があり、異分母分数の計算に欠かせない「分母同士の最小公倍数を求める」・「分母と分子の最大公約数を求める」という処理が瞬時にできる児童とそうでない児童とでは、効果的なフローチャートは異なる。個のレベルに応じて自分なりの分かりやすいフローチャートを作成し、計算の手順を明確にしておく。

学級全体として、算数が得意な児童は苦手な児童に積極的に教える雰囲気がある。今回は小グループで意見交換しながらの活動が主である。仲間同士で教え合うと共に、仲間の違った表現に気づき、考えを広げていくことができる。

##### ③ 目指す児童の姿と手立て

###### ア プログラミング的思考の育成

児童はこれまでフローチャートを扱った経験がないため、前時でフローチャートの例を示して仕組みを理解させる。

また、異分母分数のたし算の手順をいきなり考えるのではなく、まずは既習の同分母分数のたし算の計算手順をフローチャートに表す学習を取り入れる。普段は頭の中で計算していることが、フローチャートに表し視覚化することで、一つ一つのステップを踏んで最終的な答えまで行き着くことに気付かせ、論理的思考を育む。

イ 児童の正確な計算力の育成

本単元では、通分と約分の計算処理が求められるため、4年次に学習した同分母のたし算ひき算に比べ、難易度は格段に上がる。そのため途中で計算をあきらめてしまう児童や答えまで導き出したとしても正確性に欠ける児童が出てくるのが予想される。「この次はどうするのだろうか？」と戸惑う児童に対しては、作成したフローチャートに当てはめ、今どこまでできているのか、この次にどうすればよいのかを逐一確認させながら計算させる。時間は要するかもしれないが、自分の力で解けたという達成感を味わわせる。また、答えまでたどり着いたとき、「約分はこれ以上できないか？」といった答えの確かめとしてもフローチャートを活用させる。

④ 授業の考察

ア 単元6時間目（プログラミング学習1時間目）

異分母分数同士のたし算の前段階として、既習の同分母分数同士のたし算の手順をフローチャートで考えた。その中でも難易度によって、「 $\frac{2}{7} + \frac{3}{7} = \frac{5}{7}$ 」の繰り上がりのない真分数同士のたし算（レベル1）と「 $1\frac{3}{7} + 2\frac{5}{7} = 4\frac{1}{7}$ 」の繰り上がりのある帯分数同士のたし算（レベル2）に分けて取り組んだ（図1）。



図1 フローチャートを作る様子

レベル1では、ほとんどの児童が迷うことなく必要なブロックを選択し、フローチャートを完成させていった。

レベル2では、レベル1同様にフローチャートを作っていく中で、ブロックの順序が児童によって異なるという場面があった。

C1：分母がそろっているか確認した後は、整数同士を足すんだよね。  
 C2：いや、分子同士を先に足すんじゃないの。  
 C3：どっちを先にしてもいいんじゃないかな。  
 C1：確かに答えは同じになる。

フローチャートで計算の手順を考えることで、児童は必ずしも計算の手順が皆同じではないということに気付いた。それぞれのやり方で答えを確かめ合うことにより、学びを深めることができた。また、異分母分数同士のたし算になると、フローチャートはかなり複雑化するのではないかと見通しをもつ児童もいた。

イ 単元7時間目（プログラミング学習2時間目）

まず、「 $4\frac{5}{6} + 3\frac{8}{21} = 8\frac{3}{14}$ 」の異分母分数のたし算（レベル3）の計算をした。通分や約分の複雑な処理に苦戦する児童が多く、自力で答えまで辿り着いたのは29人中7人であった。その後、その計算のフローチャートをグループで話し合いながら一人一人が作成した。教師が、児童が必要だと思われるブロックを予想して用意したほか、児童が自分の言葉で書き表せる空白のブロックも用意した。通分や約分といった計算処理をどのように細かくブロックで手順を表すか、児童は実際に計算しながら考えていった。

通分の手順では、作成するフローチャートに違いが見られた（図2）。児童Aでは、

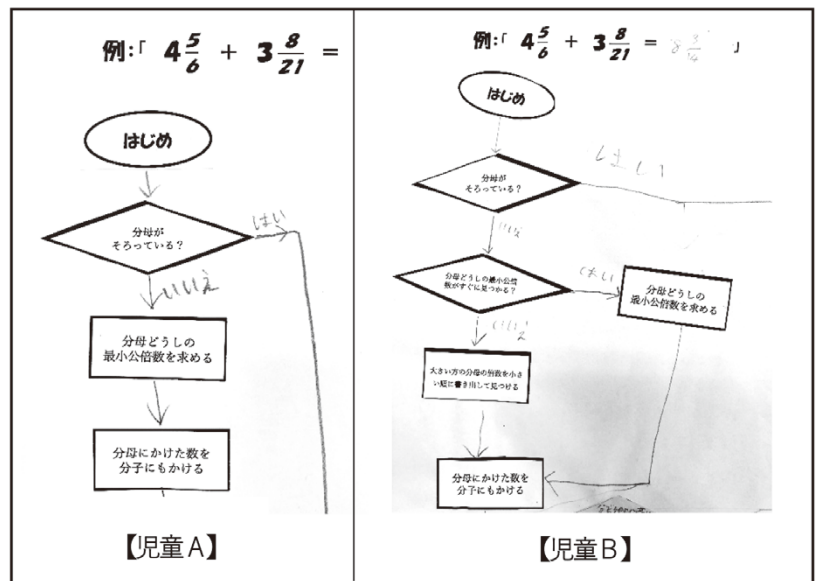


図2 児童が作成したフローチャートの一部

通分の手順を「分母同士の最小公倍数を求める」という一つのブロックで表しているのに対し、児童Bでは、「分母同士の最小公倍数がすぐに見つからない」場合を分岐のブロックでさらに細かく分けている。さらに、ある児童からは「通分の手順を細かく表さなくても理解できる」という発言もあった。約分の手順でも同様なことが見られ、児童の習熟度やあらゆる問題を想定するかどうかで、フローチャートはオリジナルなものとなった。

フローチャートを完成させた後、フローチャートを見ながら難易度が同程度のたし算の問題を解いていった。指で矢印を辿りながら一つ一つの計算手順を確認して解く様子が見られ、約分まで忘れずに正確に和を求めていた。自力で答えに辿り着けた児童は20人であった。振り返りには、「フローチャートを使えば、次にやるのがわかって迷わないで解ける」と記述した児童もいた。フローチャートを活用することが計算の助けとなった。一方、児童によってはフローチャートを活用せずに問題を解く姿も見られた。通分や約分といった計算処理が速く、計算が定着している児童にとっては、フローチャートを見る必要性がなかったと考えられる。以上から、計算手順のフローチャートは、まだ計算がそれほど定着していない児童の方が有効であると言える。

#### ウ その後の授業

単元8時間目以降の授業では、児童が作成したフローチャートをもとに、教師が分数のたし算だけでなく、ひき算でも使えるフローチャートを教室に掲示した(図3)。分岐のブロックで手順を分けることができるので、計算をフローチャートに当てはめればどのような難易度でも答えにたどり着けるということを理解し、特に複雑な計算の際にはそれを見ながら計算をする児童も見られた。異分母分数同士のたし算ひき算の定着が完全ではない児童にとっては、フローチャートをもとにして考えることが有効だった。

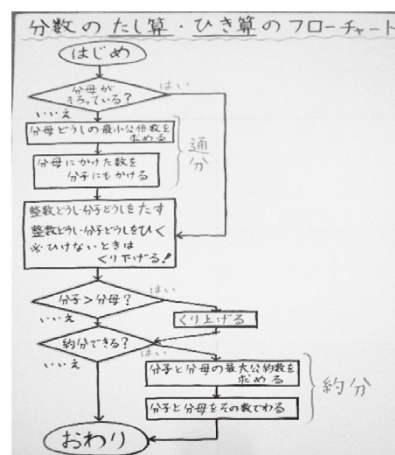


図3 教室に掲示したフローチャート

#### (2) 4年生外国語「This is my favorite place.」の実践

以下は、4年生外国語「Unit 8: This is my favorite place.」の3時間目及び4時間目で、校内のお気に入りの場所を英語で案内するという授業である。授業のねらいは、「ナビゲーションプログラムを用いて校内のお気に入りの場所を案内し合う。」である。

##### ① 単元について

本単元では、校内の教室名に慣れ親しみ、自分のお気に入りの場所へ案内するという活動を設定する。案内する側は、相手に正確に伝えることを意識し相手の理解を確認しながら、また案内される側は、自身の理解を確かめたり、自身が理解していることを相手に示したりしながらコミュニケーションを図る。

教科書によれば、案内する際に学習する表現としては、「Go straight.」や「Turn right/left.」の2つが主であり、同じ階だけでの案内活動が想定される。しかし今回は「Down the stairs.」や「Repeat 4 times.」といった発展的な内容も学習し、違う階へも案内できるようにする。このような難易度の高い学習であっても、児童が主体的に案内できるよう、案内の手順をナビゲーションプログラムに示し、それを見ながら児童同士で案内し合う活動を設定する。場所によってはスタート地点から遠く離れた場所がゴールとなるが、ゴールまでの道のりを可視化することで、自信をもって伝え合うことができる。

##### ② 児童の実態(男子20名 女子14名 計34名)

本学級の児童は、自分の考えていることを素直に表現できる児童と、自分に自信がもてず相手に思いを伝えることを苦手とする児童の二極化が見られる。英語となると、より消極的になってしまう児童も少なくない。このことから、ナビゲーションプログラムであらかじめ案内の手順を可視化しておくことは、指示を相手に伝えるというハードルが低くなるという利点につながる。ペアを組んで道案内のナビゲーションプログラムを確認しながら作成し、実際にほかのペアを案内することで、考えを共有しながら主体的に学習に取り組みせる。

##### ③ 目指す児童の姿と手立て

###### ア プログラミング的思考の育成

児童は、本学校の校舎図を見ながら、スタートからゴールまでの道筋をプログラムしていく。「今どこにいて、どちらを向いているから、次の指示は〇〇になる。」というように、ペアで対話をしながら状況に応じた指示を選択できる

ようにする。道案内をするということが、一つ一つのコマンドの指示の積み重ねによって成り立っているということに気付かせる。

#### イ 適切な英語表現の習得

児童は当然ながら本学校の教室の場所を熟知している。そのため、案内する側が正確な英語表現を使わなかったとしても、案内される側はある程度予想しながら動くことができてしまう。それを防ぐためにも、案内する側はナビプログラムを見ながら正確な英語で相手に伝えることを意識させる。そのことで自然と必要な英語表現が身につくと考える。

#### ④ 授業の考察

##### ア 単元3時間目（プログラミング学習1時間目）

実際にナビゲーションプログラムで道案内をする前段階として、まずはナビゲーションプログラムの使い方を学習した。これまで習得した「Go straight.」や「Turn right/left.」といった一つ一つの指示（コマンド）を組み合わせることによって、目的の教室まで案内することができることを理解し、実際に同じ階の近くにある教室までのナビゲーションプログラムをペアで作成した。完成したプログラムはほかのペアと交換し、お互いに校舎図を用いて英語表現を声に出しながら確かめ合い、間違えていたところは修正を加えていった。

##### イ 単元4時間目（プログラミング学習2時間目）

前時に扱った英語表現「Go straight.」「Turn right/left.」に加え、「Down the stairs.」や「Repeat ○times.」といった発展的な表現を学習した。その後前時同様にペアで、お気に入りの場所へのナビゲーションプログラムを作成した（図4）。目的地が違う階にあるということで、プログラムは複雑化するということに気付き、「このときどちらを向いているか」など、ペアで何度も作成したプログラムを確認する姿が見られた。ナビゲーションプログラムの完成後、ペアが自分たちで作成したプログラムを用い、ほかのペアを目的地まで案内する活動を行った（図5）。案内する側は、プログラムを見ながら、一つ一つの英語の指示を的確に伝えることができた。慣れてきた児童は、プログラムを見ず、相手とのアイコンタクトを意識して話すこともできていた。実際の「人」を相手に案内するという活動により、児童には当事者意識が生まれ、主体的な学習につながったと考える。

一方で、次のような問題も生じた。本来案内される側は、案内する側の指示だけを聞いてその通りに動かなければならないが、ある程度次の指示を予想して動いてしまったり、一つ一つのマスで確実に止まることができなかつたりするという様子も一部見られた。スクラッチなどのパソコン上であれば、必ず指示通りにしか動かせないため、これはアンプラグド形式のプログラミング学習ならではの課題と言える。

## 5 研究のまとめ

本研究の成果として、アンプラグド形式のプログラミング学習における児童の思考や活動の様子を示した。その中で、その有効性を2つ確認できた。

1点目は、アンプラグド形式のプログラミング学習は、複雑な手順を要する学習の補助として活用できるということである。児童にとって学習する内容が複雑だと感じる場合、フローチャート等の手順を視覚化するツールを活用することによって児童の考えが整理されるため、学習の難易度が高い場合や、発展的な内容に取り組む場合に有効である。教科や単元を精選すれば、論理的思考の育成と各教科の学びを深めるということが同時にできるという可能性をもつ。

2点目は、アンプラグド形式のプログラミング学習は、コンピュータ等のICTを活用するプログラミング学習と比べて、あらゆる学習形態に対応できるということである。本研究の算数科の実践では、グループの児童同士で意見を交わしながらフローチャートを作成することができた。また、外国語での授業実践では、画面上ではなく、児童が実際に案内するという活動を取り入れることができた。アンプラグド形式のナビゲーションプログラムやフローチャートは、各教科の単元で行いたい活動に合わせて活用できると考える。

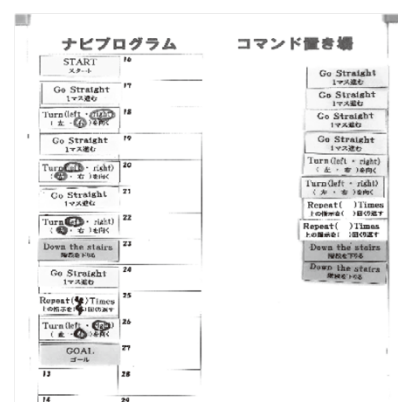


図4 児童が作成したナビゲーションプログラム



図5 道案内をする様子

一方で、2つの課題も明らかとなった。

1点目は、プログラミング学習とその教科の学びの関連性は、個人差によるところが大きいということである。本研究の算数科の実践では、分数の計算が完全に定着しているため、フローチャートを作成することに意味を見出せなかった児童もいた。児童がプログラミング学習の必要性と活用の見通しをもてるよう、教科の単元や内容の精選は必要不可欠であると言える。

2点目は、アンプラグド形式のプログラミング学習としたことで、「指示通りに動く」というプログラミング学習の醍醐味が薄れてしまったということである。本研究の外国語の実践では、指示を聞いて動く役割を児童が行ったため、完全なロボットのようにはいかなかった。今後は、学習内容やねらいとする児童の姿に応じて、アンプラグド形式とコンピュータ等を使う形式とを両立させながらプログラミング学習を行っていきたい。

## 引用・参考文献

- ・金子大輔『小学校におけるプログラミング教育の必修化とその実施上の課題』北星論集(経)第59巻, 第1号, 2019年
- ・田中良研・中田 充『小学校におけるアンプラグド形式のプログラミング教育実践』山口大学教育学部附属教育実践総合センター研究紀要第49号, 2020年3月
- ・早川裕貴『既存の授業でプログラミング的思考力を高める指導方法の工夫－4年「垂直・平行と四角形」の授業を通して－』教育実践研究第31集, 上越教育大学学校教育実践研究センター, 43-48, 2021年
- ・文部科学省『小学校プログラミング教育の手引き(第三版)』, 令和2年2月