

小学校第1学年におけるプログラミング教育導入時の カリキュラム開発に関する一考察

A consideration of programming education in elementary school 1st grade A Study on Curriculum Development

中村 めぐみ*・中川 一史**
つくば市教育委員会*・放送大学**

文部科学省は、新学習指導要領公示の中で、2020年度からの小学校におけるプログラミング教育必修化を示している。しかし、その導入の仕方については、具体的に示されておらず自治体や学校の実態にゆだねられている。教科としての位置付けではないことから、現在の教育課程の中にどのように導入していったら良いのかを考え工夫しなければならない。そこで、本研究では、義務教育最初の小学校第1学年を対象として、プログラミング教育導入のためのカリキュラム開発案件を検討した。そして、プログラミング的思考を育成していくには、段階があることが明らかになった。

キーワード：小学校 プログラミング教育 プログラミング的思考 育成 カリキュラム

1. はじめに

文部科学省は、新学習指導要領公示の中で、2020年度からの小学校におけるプログラミング教育必修化を示している。背景の1つとして挙げられているのは、将来の予測が難しい社会の到来である。情報化の進展により社会や人々の生活が急速に変化するのと同時に、地球規模で災害その他の自然現象がこれまでの予想を遙かに超えている。このような社会を生きぬかなければならない次世代を担う子どもたちに、未知の課題を解決するすべとなる資質・能力を身に付けさせる必要がある。その1つが「プログラミング的思考」であると考え。プログラミング教育を通して身に付くこの思考の様式は、私たちの身近な製品のほとんどに組み込まれているコンピュータシステムに通じるものであり、そのシステムを作り出す際に必要な思考の仕方である。つまり、これらを身に付けることで、情報や情報技術を活用することができ、急速に進む情報化社会において生き抜く力となると考えられる。また、様々な未知の課題に対しても解決の糸口を見いだす事のできる思考の仕方だと考える。

これらの必要性から、2020年から学校現場において子どもたちにプログラミング教育をしていくことが迫られているが、その導入の仕方については、具体的に示されておらず自治体や学校の実態にゆだね

られている。しかし、教科としての位置付けではないことから、現在の教育課程の中にどのように導入していったら良いのかを各自治体、学校が考え工夫しなければならない。つまり、カリキュラムマネジメントが必要である。プログラミング教育初年度においては、どんな学校においてもスキルも、環境も整っておらず、何をどのように、どこから取り入れていくことが必要で、効率的であるのだろうかを考える必要がある。

これらの問題を解決するために、プログラミング的思考を働かせてカリキュラムマネジメントを考えることができる。まず、プログラミング教育を支える様々な要素を分解してみる。文部科学省による「小学校プログラミング学習の手引」を下に考えると、A及びBにあたる教科の中での位置付け、Cにあたる領域等での位置付けの主に2つである。これらはさらにコンピュータを使う場合と、アンプラグドと呼ばれるコンピュータを使わない場合の条件によって分けられる。これらを、1年間の教育課程に適切に組み合わせて位置付けていくことで、効果的かつ効率的にプログラミング的思考を育成できるのではないかと考える。

そこで、これまで実践してきた、4つの分類のプログラミング学習を通して、小学校第1学年の教育課程における効果的なプログラミング的思考の育成のためのモデルカリキュラムの開発案件を検討する。

第1学年においては、学習指導要領に示されるように、言語活動を通じた思考力、判断力、表現力等を育む「自覚的な学び」の始まりである。幼児期の非言語によるコミュニケーションを中心とした幼児教育から、話したり聞いたり、読んだり書いたりすることで他者と関わり合う教育へ移行する。つまり、言語活動による思考の芽生えの時期であると考え。このことから、言語活動を基盤とした、論理的思考や推論を必要とするプログラミング的思考もこの時期に段階的に育成することができることから、第1学年におけるカリキュラム開発を検討することとする。

2. 研究のねらい

小学校第1学年の教育課程において、各教科および領域等の中でコンピュータを活用したプログラミング学習とアンブラグドによるプログラミング学習を意図的に組合せることにより、教科学習でのプログラミングアプリケーションをツールとして活用するためのモデルカリキュラムの開発案件を検討する。

3. 研究の方法

小学校第1学年1組児童37名を対象に、2016年～2017年度に以下のようにプログラミング教育を行った。実践の視点は「コンピュータプログラミングとアンブラグドの組合せと順番によるプログラミング的思考の育成」である。対象である第1学年の児童は義務教育初年度であり、プログラミングにおいても既成概念がないことや、学習の積み重ねによる差が少ないことなどから対象とした。具体的な方法は以下の通りである。

- ① 第1学年 教科におけるアンブラグドプログラミングを活用した論理的思考力の変容
 - ・単元テストにおける記述式欄分析
 - ・単元終了後の感想文分析
 - ・授業時における発言分析
- ② 第1学年 つくばスタイル科（総合的な学習の時間）におけるコンピュータプログラミングを活用した学習におけるワークシート分析
- ③ 第1学年国語「スイミー」におけるコンピュータプログラミングを活用した学習における児童の教科における到達状況調査
 - ・単元テストによる得点分布調査

4. 第1学年及び第6学年におけるプログラミングを活用した授業実践

4.1. 第1学年国語・算数・生活科における論理的思考力の育成

第1学年における主な教科の学習の中で論理的思考力を意識的に高めるための手立てを考えた。国語では、自分の意見を述べる際に「なんでそう思ったの」という発問を繰り返し、根拠を明確にすることを意識的に行った。これらを続け、児童は第1学年の9月を過ぎた頃から、自然と「なぜなら」という言葉で根拠を説明することができるようになった。算数では、計算の手順を1つ1つ分解して捉えることを意識させた。例えば $9 + 5 + 1$ のような3つの数の計算をする際に、手順をフローで表しながら説明させた。このような考え方は、生活の全てにおいて取り入れた。このことにより、児童の学習の習熟度が高まっていったと考える。

4.2. 第1学年 道徳「自分のことは自分で～わすれものをしないために～」

忘れ物をしないための行動を、アンブラグドプログラミングを活用し細かく分解し、一つ一つの行動を見直す作業を行った。

本時は、ワークシートを用意せずに白紙を渡し、自分なりの忘れ物をしないための視点を記入させた。最初は、「時間割を確認する」や「連絡帳を確認する」といった内容だったが、「学校から帰ったらどうしますか。」という発問をすると、図1のように行動を順序立てて考えはじめた。行動を分解していく中で、いくつかの重要な行動を踏まえた行動パターンを繰り返すことがミスである忘れ物を防ぐことにつながることを体得することができた。

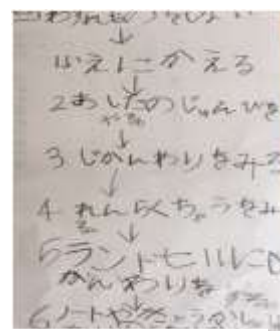


図1 課題プログラムの予想

4.3. 第1学年 つくばスタイル科におけるコンピュータプログラミングを活用した学習

つくばスタイル科（総合的な学習の時間）に課題

プログラムと同様の動きをつくるプログラムを予想させる活動を行った。図2のように、プログラミングの仕組み論理的に思考できるように手順を考えるにワークシートを活用する。課題プログラムを推測する際に、試行錯誤するのではなく、根拠に基づいて推測させるために、ワークシートで考える活動を行った。児童はワークシートに示されたフローによって、見通しをもってプログラミングをすることができ、一つの動作をプログラムするに当たっても、何通りかの方法があることに気付いた。まだ、発達段階的に確実な根拠に基づいた論理的な思考をすることは難しかったが、今までのように、根拠のない数字をあてはめるのではなく、ある程度の理由の基に数字を当てはめている様子も見られた。



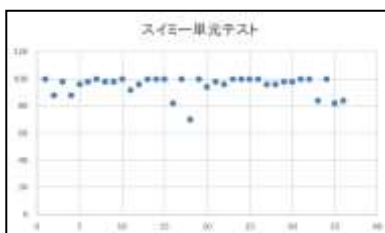
図2 課題プログラムの予想

4.4. 第1学年国語「スイミー」における コンピュータプログラミングを活用した学習

「スイミー」の物語の場面の様子をプログラミングアプリケーションでアニメーションにして音読を行うこととした。アニメーションを作るために、登場人物の大きさや、動きの速さなど細かな部分にまで目を向け、文章から読み取ろうとする姿が見られた。このことは、教科目標である「人物や場面の様子を思いうかべて読もう」に迫ることができたといえる。音読発表会の群読した様子においても、アニメーションを背景とし、これまでの音読と比べ、細かなところまで声の出し方を考えるなど、表現力が高まったと感じられた。

表1は単元テストにおける点数分布である。この単元テストの平均点は、95.3点で。最高点100点が12名（全体の32%）、最低点が70点と非常に習熟度が高い。スイミーの気持ちや行動を問う設問を記述式で答えるテストであることから、プログラミングを活用した学

表1 スイミーの単元テストにおける点数分布



習が、物語を理解することに効果的であったと考える。

5. 結果と考察

対象とした第1学年児童に37名に、プログラミング学習をした後に、児童の変容について意識調査を行った。「プログラミングをしてみたいですか」という質問に対し、学習前は27人だったが、学習後は37人全員がまたやりたいと答えた。その理由については、「自分でイラストを動かせるから」といった、プログラミングそのもの楽しさを感じている児童が9名（類似する回答を含む）や、「みんなで考えられるから」といった協働作業に楽しさを感じている児童が5名であった。これらに見られるように、プログラミング自体への興味関心の高さだけでなく、授業を進めていく上で協働的に作業を行うことに楽しさを感じていることが分かった。これは、プログラミング導入時において、導入初期は1人で学習を行うより、ペアやグループでの学習場面を設定することが重要と考える。小学校1年生の発達段階では、1人で作成したいものや自分の意思を明確に自覚することが難しく、教科学習においても友達との会話の中でヒントを得たり、模倣することで自分の作品への思いに気付いたりする傾向が見られる。また、論理的思考やプログラミング的思考が苦手な児童においては、1人で学習することで学習への不安やできない事への劣等感が生まれやすい。そこで、ペアやグループといった中で対話を繰り返すことでそれらへの気づきを促し、プログラミングへの抵抗感を軽減することができると思われる。

研究方法①の教科における論理的思考力の変容については、単元テストにおける記述式欄を見ると、理由を問う問題に対して、「なぜなら」や「だから」といった表現を多くの児童が用いながら回答し、正答率も高かった。これは、授業中の発言においても同様であり、論理的思考を意識させるための教師の「なんで」という問いが次第に少なくなり、児童自らがその説明をすることができるようになってきた。また、説明の場面においても、「まず」「次に」「なぜなら」といった接続詞を使うことが多くなり、相手にも分かりやすく説明しようとする意識するなどの変容が見られた。

算数では、フローを使ったアンプラグドプログラミングを意識することで、まず、どこを計算し、次

にどこを計算するといった思考の仕方ができるようになった。また、「計算の仕方を考えよう」という課題に対して、解を求める過程を説明する際には「最初に」「その答えと」など、手順を短い言葉で順序よく言い表すことができるようになった。計算の手順を分解することで、表現がしやすくなるため、説明の仕方が上達した。これらは、アンプラグドプログラミングのフローを意識させることで、思考が整理され、解を導き出すまでの流れを明確にすることが身に付いたと考える。

研究方法②の第1学年 つくばスタイル科におけるコンピュータプログラミングを活用した学習においては、ワークシートの分析を行った。この学習をするまでに、教科の中で、順序を意識させた活動を行ってきたことから、フローチャートに表すことは難しくなかった。また、その枠内に記入された言葉も、文章で記述するのではなく、動作に対して1つの命令が書かれていた。さらに、既存プログラムを予想する場面では、画面に現れるアニメーションのカウントダウンのプログラムについて、出る→消えるの繰り返しという分解ができていた。

研究方法③は、プログラミング教育導入において大きな目標である教科の中での効果的な活用である。国語「スイミー」におけるプログラミングアプリケーションの活用場面では、自分の好きな場面を音読する際の背景のアニメーションを作ることを通して、自分が捉えた心情や、情景を根拠を明確にしながらプログラムに表現することができていた。また、操作についても、事前までのカリキュラムマネジメントによる技能の習得により、表現活動の妨げになる様子は見られず、終始、自分の意図した動きを表現しようとすることに集中している様子が見られた。単元終了後の感想でも、プログラミングについての感想ではなく、登場人物への思いなどが書かれており、通常の国語の授業と同様の感想が得られるなど、単元目標である自分の好きな場面を選ぶことにおいて、説明する様子が見られた。これらは、プログラミングアプリケーションを音読といった言語活動の補助的な表現ツールとして活用することで、自分の意図した表現をするために文章を論理的に理解しようとする思考が働いたと考える。これまでの、紙芝居など固定された表現より、登場人物等を速さ、色の変化、様々な動きで表現できることから、よりそれらを設定するための文章の読み取りが必要にな

る。つまり、物語全体を分解し、1つ1つの文言のもつ意味などに注目して科学的な理解促したことが、スイミーの単元目標である物語の心情や情景を読み取る事へつながったと考えられる。これは、プログラミングをすることに注視することなく、アプリケーションが表現ツールとして活用できていたと考えられる。さらに、単元テストの回答の内容から、登場人物の行動の根拠が示されていることや、情景の読み取りについても正当率が高い様子から、プログラミングアプリケーションが国語の教科の見方・考え方を働かせながら、教科目標を十分に達成させるための手立ての1つになっていると言える。

これらのことから、プログラミング的思考を育成して行くには、ある程度段階があると考えられる。まずは、普段の教科及び領域、さらには行事などにおいて、意図的に論理的思考をするような手立てを講じることである。発達段階においては、条件によって解決方法がちがったりすることも体験しながら、プログラミング的思考を汎用的な問題解決のスキルとして働かせていくことが必要だと考える。次に、アンプラグドプログラミング学習を位置付け、フローチャートなどを用いて、プログラミング的思考に特化した学習を位置付ける。そして、領域等の時間を活用して、コンピュータプログラミングの操作を身に付けるための時間を設定する。この上で、教科の学習における教科目標を達成させるための手立てとして、コンピュータプログラミングを活用した学習を取り入れる必要があると考える。教科教育のツールとしての活用と言える。

このように、プログラミング的思考の育成をするための2020年プログラミング教育必修化に向け、より効果的で効率的に導入できるようなカリキュラムマネジメントをすることが、スムーズな導入への方策と考えられる。

今回の研究の課題としては、それぞれの実践に対してその達成状況を測定する方法が、事例ごとに違ってしまう、開発案件における検証が十分でなかったことがあげられる。今後、カリキュラム策定を見通した実践を行うにあたり、測定の方法を統一することで、より教育効果を根拠としたカリキュラムの策定方法を提案していきたい。

6. 参考文献

文部科学省 小学校プログラミング教育の手引第1

日本STEM教育学会 第1回年次大会 一般研究発表予稿

版（平成30年 3月）