

小学生向けプログラミング教育の実践

Programming Education for Elementary School Students

加藤 智也 KATO Tomoya

1. はじめに

Society5.0時代を見据え、日本の情報教育が大きく変わる。平成29年に告示された小学校学習指導要領では、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（プログラミング的思考力）を身に付けるための学習活動を計画的に実施することが明記され、プログラミングは児童が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の一つとしている。同じく平成29年に告示された中学校学習指導要領では、技術・家庭科（技術分野）において、「計測・制御のプログラミング」に加え、「ネットワークを利用した双方向性のあるコンテンツのプログラミング」を学ぶとある。さらに、平成30年に告示された高等学校指導要領では、情報科において共通必修科目「情報Ⅰ」を新設し、全ての生徒がプログラミングのほか、ネットワークやデータベースの基礎等について学習し、その基礎の上に選択履修の「情報Ⅱ」を開設することが示され、大学入学共通テストで入試科目「情報」の2024年度導入に向け検討が進められている。

そこで本研究では、プログラミングの必要性を明確にしたうえで、2020年度の小学校でのプログラミング教育開始に先駆けて、小学生に対するプログラミング教育を検討・実践し、その効果を検証する。

2. プログラミング教育の必要性

小学校でプログラミング教育を扱う理由としては、情報技術であふれる高度情報化社会に不可欠な教養を身に着けることと、未来を見越した職業選択ができる創造性のある人材を育成することにある。

2.1 教養としての側面

高度情報化社会においては、コンピュータやネットワークが家電や自動車をはじめ身近なあらゆるモノ、分野、環境に溶け込み、情報技術に支えられながら我々は生活している。それらはすべてプログラミングをベースに成り立っており、コンピュータが動く仕組みを知ることにより、より適切、効果的に活用できるだけでなく世の中の新たな仕組みを創造することにもつながる。あらゆる活動においてコンピュータを活用することが求められる社会では、プログラミングおよびプログラミング的思考力は文章の読み書きや計算と同様に高度情報化社会を生き抜くための誰もが身につけるべき基礎能力・教養として、変容していく社会に不可欠である。

2.2 職業選択の側面

IoT、ビッグデータ、人工知能をはじめとしたデータ利活用に関連した技術革新は、「第四次産業革命」と呼ばれ、動力の獲得、革新、自動化に次ぐ新たな産業構造の変革の契機として経済へ大きな影響を与えるものと考えられている。技術革新の急速な発展により、2045年ごろには人工知能（AI）が発達し、人間の知性を超えるシンギュラリティ（技術的特異点）を迎え人間の生活に大きな変化が起こる（レイ・カーツワイル博士、2005）、知能やロボット技術の進化により今後10～20年程度で、現在存在する仕事の半数近くが自動化される可能性が高い（オックスフォード大学准教授マイケル・オズボーン氏、2013年）、2011年度にアメリカの小学校に入学した子どもたちの65%は将来、今は存在していない職業に就く（ニューヨーク市立大学大学院センター教授キャシー・デビッドソン氏、2011年）など、さまざまな予測があるなか、「第4次産業革命」による労働市場が変化するのは確実であり、AIの得手・不得手を理解しAIができないことを創出・創造するとともにAIをどう使うか判断できる人間と、AIの監視のもと最適解に従って動く人間とに分かれる可能性がある。どちらが幸せというわけではなく、これらのことは将来の職業選択に関係するため、プログラミングを経験しAIに対する理解を深める必要がある。

2.3 小学校へのプログラミング教育導入

世界各国では、情報活用能力を備え創造性に富んだ人材の育成が急務として、社会で必要となる資質や能力を育む要素としてプログラミングが重視されている。プログラミング言語は世界共通であり、どの国も同じ土俵に立って勝負することになるため、国際競争力を高めるためにも多くの国々でプログラミング教育に力を入れた取り組みが始まっており、プログラミング教育の必修化・カリキュラム化が進んでいる。いずれの国の政府も、次世代を生きる子どもにとって、また自国の経済成長にとってプログラミングの教育が重要であるという認識は共通しており、IT人材不足の深刻化の解消としても期待されている。そこで、2020年度から実施される小学校学習指導要領の総則において、「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」と明記され、プログラミング教育が全面実施されることとなった。文部科学省の「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」の最終報告書（2016年）では、小学校段階におけるプログラミング教育について、子どもがコンピュータに意図した処理を行うよう指示することを体験することで、職業や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を育むことが目的」と定義し、時代の変化や技術革新の中で移り変わっていくことが予測される特定の技術や個別のプログラミング言語を用いた記述方法（コーディング）を覚えることではないと明示している。

3. プログラミング教育の検討

3.1 プログラミング言語の選定

プログラムとはコンピュータにさせたいことの手順や内容（命令）を記した指示の集まりのことであり、コンピュータはこのプログラムに基づき事前に決められた仕様（アルゴリズム）に沿って動く。コンピュータは自ら課題を解決することができないため、人間が課題解決手順をプログラムという形で指示を与える必要がある。プログラミングとはコンピュータを動かすためのプログラムを作る作業のことであり、プログラミング言語と呼ばれる独自のルールに基づいて書かれたテキストデータの集合である。一般的なプログラミング言語として、Java、C、Python、Ruby、JavaScriptなどさまざまなものがあるが、関数やオブジェクトなどの概念など難解なものも多く、初等教育でプログラミング言語の記述方法や命令を教育するのは難しい。

しかし近年になって、子どもでも視覚的に把握しやすいビジュアルプログラミング言語が多数開発され、プログラミング教育を行う環境が整ってきた。ビジュアルプログラミング言語は、部品として用意された命令を表すブロックを組み合わせることでプログラムを直感的に組むことが可能である。ブロックイメージの持つ機能を視覚的に認識し、組み合わせを考え、ブロックを画面上にマウスを用いてドラッグし配置するだけでよいものも多く、構文を覚えてプログラムコードを入力する必要はない。視認性、直観性、操作性が優れているため、プログラミングを容易にしている。なかでも、マサチューセッツ工科大学メディアラボが開発した「Scratch」はプログラミング教育の導入として世界的に広く使われている。ダウンロードしスタンドアロン環境でも、Webサービスとしてブラウザ上でもしかも、オブジェクト指向の概念を取り入れているため、さらに高度なプログラミングにつながやすい。また、Webサービスを利用することで自分の作品を世界中に公開でき、他人の作品も見ることができる。文部科学省が小学校教員向けに子ども向けビジュアルプログラミング言語「Scratch」を使った指導例などをまとめているなど、今後さらなる利用の拡大が見込まれる。本格的なコーディングはせずとも、命令の組み合わせや順序を論理的・創造的に思考することにより、課題を発見・解決し、新たな価値を創造する能力、すなわち「プログラミング的思考」を養うことが可能である。

そこで、今回は「Scratch (Version3.0)」を選定し、Windows10 環境でブラウザGoogle Chrome 上で実施することとした。

3.2 内容の設計

対象者は小学3～6年生で、複数の学年が混ざってプログラミングを行う。そのため知識・能力差が大きい。よって、何らかの教科等に位置付けることなく、「プログラミング的思考」を育むこと、プログラムの働きやよさに気付くことができるようにすることを目標とする。プログラミングの楽しさや面白さ、達成感などを味わえる題材などでプログラミングを体験する取組となるよう、学年によらず楽しめるゲームを考案した。具体的には、宇宙空間で画面上から下へ自動的に動く敵キャラクターおよび敵ボスキャラクターにぶつ

からないように、自分のキャラクターを左右に操作するプログラムを作成することを題材とする。ベースとなるプログラムには、敵キャラクターが自動的に画面上を動くプログラム、自分のキャラクターを自分で操作できるようにするプログラム、ぶつかったかどうかを判定しぶつかった際には動作が停止するプログラムなどとなる。これらのプログラムにより、プログラミングの基本要素である順次処理、分岐処理、繰り返し処理を理解し応用できるようにする。また、取り組む順番を綿密に計画し、どこの段階で何をどうするべきなのかを見童に考えさせる。そして、例示をもとに自分でまねをしながら進め、動作がうまくいかなければその理由を考え修正するよう導く（デバッグ）。そして、学年もさまざまということから、拡張できる要素をあらかじめ考えておき、進行に応じて取り組ませる。

4. プログラミング教育の実践

4.1 事前準備

授業内に児童が一から作り上げる時間はないため、背景、自機キャラクター、敵キャラクター、敵ボスキャラクター、音などの素材はあらかじめ準備し、小学生には難しい座標や変数がからむプログラム（敵キャラクターが時間差で上から下へ回転しながら落ちて来る）を入れた状態で「初期データ」を準備した。進行は次のとおりとした。

- ①自機キャラクターの右移動
- ②自機キャラクターの左移動
- ③スタート時の背景と音の設定
- ④敵キャラクターの動きの確認と説明（事前作成済）
- ⑤敵キャラクターの複製
- ⑥当たり判定（自機キャラクターが敵キャラに触れたら）
- ⑦ゲームオーバー時の背景と音の設定
- ⑧敵ボスキャラクターの設定（敵キャラクターのプログラムの内容を複製した後、登場を遅らせたり、移動スピードを変えたり、回転をしないようにする）

ここまでで予定のゲームが完成する。念のため、授業で困った小学生がいた場合に迫いつくために読み込ませるための「完成データ」を準備した（図1）。

さらには、もっと応用できることを示すために、そして最後にみんなで遊び楽しんで終われるように、敵が横に移動したり、自機か

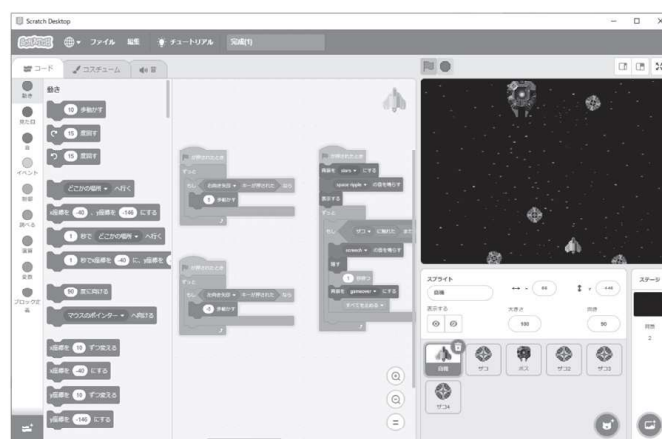


図1.ゲームの作成・完成画面

ら弾を発射したり、敵・ボスが撃たれたら画面上から消したり、スコアを表示する機能を追加し、音や画面表示の細かい点を調整した「応用データ」も準備した。

授業アシスタント 8 名については、事前に Scratch を体験させ、実施事項を把握させるとともに、授業中に戸惑わないようにプログラムを A 4 一枚にまとめた資料を配布した。

4.2 実践

小学校 3～6 年生 23 名を対象に 60 分、講師と授業アシスタント 8 名で実施した。

児童が来る前に、あらかじめパソコンを立ち上げブラウザから Scratch を開き、準備しておいた「初期データ」を読み込んだ状態で開始した。まずはじめに、プログラミング作業に入る前に 5 分程度、身近にあるコンピュータについて意識してもらい、プログラムの役割を説明した。そして、ゲームは遊ぶだけではなく、自分で作ることができると期待感を持たせた。

その後、実際にプログラミング作業に入り、準備した流れのとおり進行し、説明と授業アシスタントのフォローのもと、各々がゲームを完成させた（図 2、図 3）。ただ単に同じように操作するよう説明するだけでなく、実際に児童が自分で考えながら作業する時間を設けた。例えば、自機キャラクターの右移動だけ例示し左移動を考えさせたり、敵ボスキャラクターの動きを決める場合に敵キャラクターの動きをもとに、どこをどうどうすれば実現できるのかを考えさせたりした。

最後は「応用データ」を読む込み、もう少し工夫すればこんなに面白くできることを示し、みんなで楽しく遊んだ。



図 2. 授業の様子



図 3. プログラミングに取り組む様子

5. 評価

授業開始時から、「楽しみにしていた！」と目を輝かせる児童が多数いた。実際に始めると真剣に取り組む、授業のとおりになってくると、次第に自分なりに、「自機キャラクターを前後に動かしたい」、「敵キャラクターをもっと早く動かした方が面白い」、などさまざまな言葉が飛び交い、実際に改良する児童も複数みられ、それを児童同士で見せ合うなどの姿も見られた。「完成データ」を使うこともなく、各々順調に作業をすることができた。授

業が終わるときには、「もっとやりたい」、「めっちゃ楽しい」など肯定的な言葉を聞くことができた。

授業後の簡単なアンケートの結果は次のとおりである（図4～図6）。

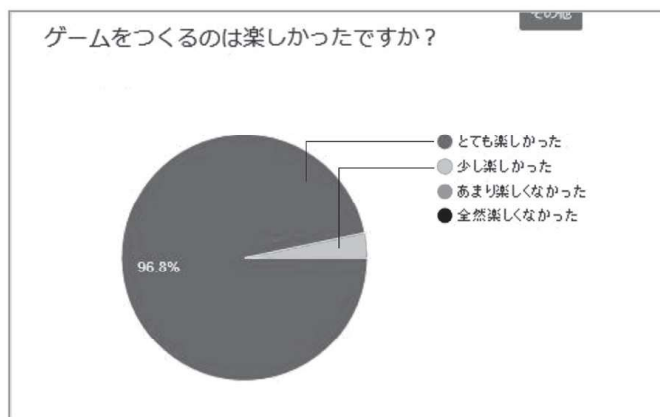


図4. 事後アンケート結果（ゲームをつくるのは楽しかったですか？）

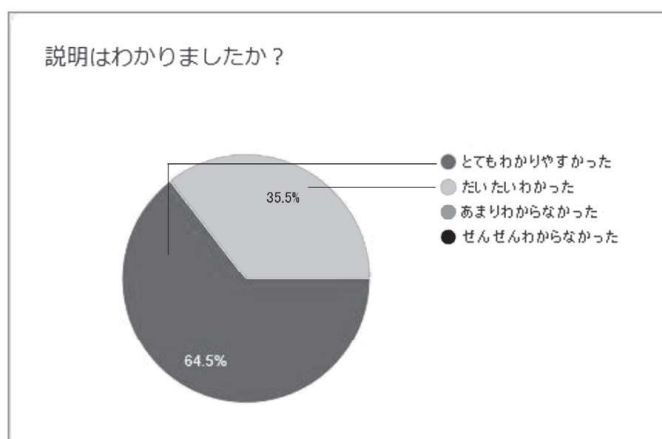


図5. 事後アンケート結果（説明はわかりましたか？）

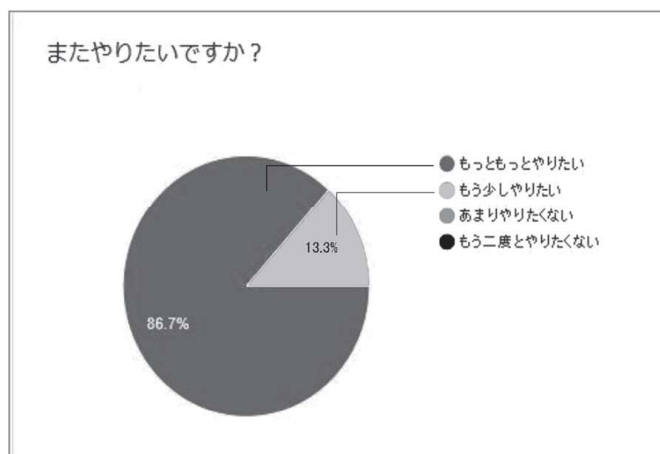


図6. 事後アンケート結果（またやりたいですか？）

「ゲームをつくるのは楽しかったですか？」「説明はわかりましたか？」「またやりたいですか？」の質問において、図で示すとおり否定的な回答は一切なく、きわめて高い評価を

得ることができた。

6. 考察

今回授業で実践した内容、評価から、①論理的思考能力、②問題解決能力、③創造・発想力④コミュニケーション・コラボレーション能力⑤学習意欲について考察する。

①論理的思考能力

ゲームの制作において、プログラミングの基本要素である順次処理、分岐処理、繰り返し処理を組み合わせることで可能となる機能を、簡単な例としてたくさん盛り込んだ。コンピュータはプログラムに書かれたとおりに忠実に動き、意図したとおりにコンピュータが動くようにプログラミングするには、最初にコンピュータに何をさせたいかを明確にしたうえで、どういった動作がどのような順番で必要かを考え、それをコンピュータが理解できるようにプログラミングの基本要素の組み合わせにより指示を与えるプログラムを作り、それを実行してコンピュータの動きを確認する。このように、ゲームの機能の実現することを通じて、物事を整理し順序立てて考える論理的思考能力を養うことができたといえる。

②問題解決能力

意図するようにキャラクターを動かしたり、キャラクターがぶつかったらどうなるべきなのか、など児童が考えながらプログラミングをすることで、トライアンドエラーを重ねながら目的とする機能を実装していくことを学んだ。プログラミングは、少しでも曖昧な記述や漏れが少しでもあれば何らかの障害が発生する。動かなかったり自分の意図と違う動きをしたときは、自分が記述したプログラムの何が違うのかを考える必要があり、正しいと考えていた前提を疑い、一つひとつを分析し、問題を見つけ、解決策を考えて、プログラムを修正する必要がある。今回、このような作業の繰り返しにより自分が思い描いた動きをするプログラミングをすることで、問題解決能力が養われたといえる。

③創造・発想力

今回はただ敵キャラクターをよける程度の仕組みを作っただけであったが、もっとこうしたいという気持ちが芽生え、実装しようとしたり、実現するにはどうすればよいか質問する姿がみられた。もちろん敢えて児童がやりたいことを自分で考える余地を残して今回のプランを練ったわけだが、自機や敵のキャラクターの動きを変えたい、敵の種類を増やしたい、弾を出して攻撃したい、スコアを表示したい、などさまざまなアイデアが生まれたようである。プログラミングで何ができるかを考え発想するのは、プログラミングをする本人次第であり、自分のアイデアをプログラミングで表現・具現化することが可能である。よって、柔軟な発想を持って物事を多角的にとらえ、創造力を高めるのに有効であるといえる。

④コミュニケーション・コラボレーション能力

プログラミングで困っている児童がいると隣の友達に質問したり、積極的に教える児童

がいたり、自分のアイデアを伝えたり、改良したものを見せ合ったりする様子から、プログラミングは「コミュニケーション・コラボレーション能力」の向上に役立ったといえる。

⑤学習意欲

今回普段楽しんでいるゲームを自分で作るということから、授業開始時から興味・関心を示す児童が多く、実際にプログラミングをさせると意欲的に取り組み時間を忘れ熱中している様子がみられた。また、他の児童がプログラミングしたものに刺激を受け、自分も同じものを作りたいと挑戦する意欲をもつようになる。よって、試行錯誤しながら自分のアイデアを実現することは、学習意欲を向上させるといえる。

7. おわりに

プログラミングを通じて高度情報化社会で不可欠な機器やサービスの仕組みを知っておくことは、世の中の仕組みを知るうえで不可欠であり、社会のありとあらゆる場面における対処能力が大きく向上し、将来どんな職種についたとしても役立つとともに、自ら新たな仕組みを創造することにつながると考えられる。

本研究では、2020年度から小学校ではじまるプログラミング教育に先駆け、プログラミング教育の必要性を明確にしたうえで、プログラミング教育用ツールとしてビジュアルプログラミング言語「Scratch」を用いて、授業を実践しその効果を検証した。その結果、論理的思考能力や問題解決能力、創造・発想力、コミュニケーション・コラボレーション能力を高め、さらには学習意欲を向上させるなど、さまざまな効果を実証できた。ただ、今回は情報を専門とする教員が授業を行い、授業アシスタントの手厚いフォローもあったため実現できたとも考えられるため、実際の授業では、各教科を担当する教員のプログラミング指導力を高めるとともに、ICT支援員の活用など検討していく必要がある。

【参考文献】

- [1] 加藤智也:「小学校におけるプログラミング教育に関する考察」『名古屋芸術大学研究紀要』第38巻、p.p.77-84 (2017)
- [2] 小学校プログラミング教育の手引(第三版)(2020)
https://www.mext.go.jp/content/20200218-mxt_jogai02-100003171_002.pdf
- [3] 世界最先端IT 国家創造宣言(改訂版)(2016)
<https://www.kantei.go.jp/jp/singi/it2/kettei/pdf/20130614/siryou1.pdf>
- [4] Scratch
<https://scratch.mit.edu/>
- [5] Scratchではじめよう! プログラミング入門(2019)
<https://scratch.mit.edu/studios/1168062/>