

教育用プログラミング環境 Scratch の算数・数学教育への応用

中里彩乃*1・市村将人*2・田川貴章*3・高数学*4

Email: a110142n@st.u-gakugei.ac.jp

*1: 東京学芸大学教育学部初等教員養成課程

*2: 東京学芸大学教育学部初等教員養成課程

*3: 横浜市立谷本中学校

*4: 東京学芸大学

◎Key Words Scratch, プログラミング, 数学教育

1. はじめに

本研究は教育用プログラミング環境 Scratch の算数・数学教育への応用に関するものである。

社会の情報化に伴い、子どもたちのコンピュータ・リテラシーを向上するために、学校現場へのコンピュータの導入が進められてきた。以来、学校現場では、調べ学習におけるホームページの閲覧や、表計算ソフトの使用、絵やグラフ等の描画、プレゼンテーションを中心にコンピュータが用いられてきた。しかし、これらはコンピュータ利用のごく一部にすぎず、コンピュータにしかできないことの代表的なものは、プログラミングやそれに伴うシミュレーションである。プログラミングやシミュレーションには、抽象的概念を具体化する、複雑な現象を複雑なまま表現するといった特長を持つ。学校現場では、この種のコンピュータ利用はほとんどなされておらず、コンピュータというリソースの機能を十分に活かしているとはいえない。コンピュータの特性を活かすことでしか実現できない授業があるのではないかと考える。

そこで本研究では、中学生を対象に、教育用プログラミング環境 Scratch を用いた数学科における、コンピュータの特性を活かした授業実践を試み、その効果を検討する。本稿では、まず、本研究におけるプログラミング教育の意義を述べ、次に本研究で用いた教育用プログラミング環境 Scratch の概要と導入意義を述べる。最後に実施した授業の概要と分析結果を述べる。

2. 本研究におけるプログラミング教育の意義

2.1 なぜ子どもにプログラミングさせるのか

プログラミングとは、プログラミング言語を使用して、人間が意図する動作や処理をコンピュータに実行させるプログラムを書くことである。プログラムの作成には、コンピュータへの正確な動作命令が必要なため、どういう手順を考えたら、コンピュータが意図した通りに動いてくれるのか、注意深く思考する必要がある。従って、ものごとの手順の考え方、つまりアルゴリズムの考え方を身につけることができると考える。子どもたちにアルゴリズムを体得させ、ものごとを論理的に見る目を養うことは重要であり、子どもたちにプログラミングさせる意義であると考えられる。また、同じ動作や処理を意図した場合でも、プログラムの中身

は子どもによって全くバラバラで違ったものになる。プログラムに子どもたちの個性が表れ、学び合うことで学習の幅が広がると考える。

さらに、プログラミングは、理論や数式をシミュレーションして、抽象的な概念や複雑な現象をコンピュータ上に実現する特長をもつ。子どもたちにとってプログラミングは、学校で学習した理論を実証できる道具であるといえる。特に、複雑で一見よくわからない現象は、現実では実現させることができない。しかし、コンピュータ上なら実現でき、子どもたちに複雑な現象を体感させることが可能となる。今まで、頭の中だけで理解していた理論や概念を実際に体感することで、子どもの世界が広がるのではないかと考える。

以上より、子どもたちがプログラミングを通して、アルゴリズムの考え方や、理論の実証による、理論や概念の本質をつかむことをプログラミング教育の意義とする。プログラミングの知識や高度な技術を教えることを意義とするものではない。

次の項目では学校現場で実際に行われている情報教育と教科教育の領域での意義を述べる。

2.2 情報教育における意義

情報教育の目指すところは、子どもたちの「情報活用能力」の育成を図ることである。学習指導要領では「情報活用能力」は「情報活用の実践力」、「情報の科学的理解」、「情報社会に参画しようとする態度」の3要素で構成されると明記されている。この3要素の1つである「情報の科学的な理解」には、情報活用の基礎となる情報手段の特性の理解が含まれる。中学校段階では、具体的に、コンピュータの構成と基本的な情報処理の仕組み、情報通信ネットワークにおける基本的な情報利用の仕組みについて理解することが目指される。この目標に対応する教育方法にプログラミング学習が挙げられる。なぜなら、プログラミング学習はコンピュータに情報処理の命令を出すことであり、コンピュータの本質そのものを理解することにつながるからである。現在、プログラミング学習は、中学校の技術・家庭科分野での「デジタル作品の設計・制作」「コンピュータを利用した計測・制御の基本的な仕組み」において必修化されている。また、「情報活用能力」は各教科を通じて育成されることが望まれるため、プロ

プログラミング学習を、技術・家庭科以外の教科教育にも取り入れ、情報教育の中でのプログラミング学習の比重を増やすことには意義があると考えます。

2.3 教科教育における意義

プログラミングやそれに伴うシミュレーションには、抽象的概念を具体化する、複雑な現象を複雑なまま表現するといった利点をもつ。特に理数教科においては、先に挙げた抽象的概念や複雑な現象を取り扱う場面が多く、コンピュータを利用する意義があると考えます。

当方研究室では、過去数回にわたって Scratch と Squeak を用いた授業実践を行っている。Squeak とは、Scratch の環境を実装しているプログラミング環境であり、Scratch は Squeak の流れを汲んでいる。Scratch を使った、中学校数学科における「 π 」を取り扱った授業実践¹では、無限多角形を描画することによって、円周率が 3.14 に収束するという学習を行い、「無限」という抽象的概念には「収束」という側面があることを生徒たちが体験的に理解することができた。Squeak を使った、自動販売機をつくる授業実践²では、自動販売機をつくることを通して、売買行為を学習させることができた。

さらに、プログラミングを通じた教科教育の効果については、教育用プログラミング環境 LOGO の開発者である S.Papert による検討が知られている。S.Papert は、ある事象をコンピュータ上に自分でプログラミングしシミュレーションすることによって、その事象についてのより深い理解が得られると述べている。

本研究ではコンピュータの、複雑な現象を複雑なまま表現できる特性に着目し、授業案を考案する。授業案の概要については実践の内容で詳しく記載する。

3. なぜ Scratch なのか

3.1 Scratch の概要

Scratch とは、コンピュータを「使って消費する人」から「使って作品を作る人」に変わろうという理念のもと、マサチューセッツ工科大学メディアラボで開発された、子ども向けのプログラミング環境である。Scratch は 1 つのオブジェクトに対してプログラムを書く、オブジェクト志向型のプログラミング環境となっており、複数のオブジェクトをまとめることで複雑なものをつくりだすことが可能である。

下の図 1 が Scratch のプログラミング・シミュレーション画面である。画面左側のブロックパレットエリアには、命令が書かれたブロックが並んでいる。この中からブロックを選び、画面中央のスクリプティングエリアでブロックをつなぎ合わせることでプログラムを書く。作成されたプログラムは画面右上のアクションエリアで実行され、スプライトと呼ばれるオブジェクトが、プログラムの命令通りに動くことになる。

プログラムを構築するブロックは多数搭載されている。プログラミングの重要な構成要素である、「条件分岐」「繰り返し」は「制御」のブロックに搭載されてい

る。演算に関するブロックも搭載されており、乱数や変数を作成することも可能である。また、動物や人の声や楽器など「音」に関するブロックもあり、子どもの興味を引くような工夫が施されている。スプライトも、起動時はネコに統一されているが、他にも様々なキャラクターが複数搭載されている。

Scratch には、コンピュータにインストールして使用する Scratch1.4 バージョンと、web ブラウザー上で使用する Scratch2.0 バージョンが現在までに発表されている。また、Scratch はフリーソフトウェアであり、Windows、Mac 等の主要なプラットフォームで動作可能である。



図 1. プログラミングの一例 (ランダム・ウォーク)

3.2 Scratch を用いる意義

「R」「Basic」等のプログラミング環境では、多くの命令を覚えて、コードを打ち込み、プログラムを書く。一方、Scratch ではブロックをつなぎ合わせることでプログラムを書く。コードを打ち込む必要がないため、文法の誤りを気にせずにプログラミングできる。また操作方法も基本的にマウスによるドラッグ&ドロップのみであるため、比較的簡単にプログラミングできる仕様となっている。従って Scratch はプログラミング初心者やコンピュータに不慣れな者にも、簡単にプログラミングに入門できる環境であると考えます。

Scratch の概要で述べたように、Scratch はオブジェクト志向型のプログラミング環境であり、1 つのオブジェクトに対して、1 つのプログラムを書くことになるので、アルゴリズムの学習に適しているといえる。また、ブロックも多数搭載されているので、子どもが自由に発想したことをコンピュータに命令し、目に見える形で表現される過程を楽しみながら、アルゴリズムの学習が可能になると考える。

以上より本実践では Scratch を用いることとした。バージョンは 1.4 バージョンを用いる。1.4 バージョンでは Scratch を実装する Smalltalk 環境まで降りることができ、Scratch の仕様を変更することができる。今回の実践では Scratch の仕様の変更はないが、授業内容によっては、仕様を変更する必要もあるため、学校現場では 1.4 バージョンを使用することが望ましいと考えた。

4. 数学教育における授業実践報告

4.1 本実践の概要

本実践は、Scratch によるプログラミング・シミュレーションを数学に応用した授業の効果を分析するため

¹ 富野, 高籾, 田川(2009)

² 手塚, 高籾, 小森, 鈴木, 早川(2008)

に行うものである。本実践は、横浜市立谷本中学校の協力を得て行った。作成した授業案は全6時間であり、1回の実践毎に2時間ずつ実施する。第1回授業実践での2時間では、プログラミングに入門する授業を行った。第2回と第3回の授業実践では、中学校数学科の関数分野と確率分野における授業案を作成した。大まかな授業内容は以下の通りである。

第1回授業実践…簡単なピンボールゲームを作成して、プログラミングに入門する。

第2回授業実践…ランダム・ウォークや正規分布をシミュレーションし、確率と関数の本質を考える。

第3回授業実践…カオス理論に従う複雑な現象をシミュレーションし、「複雑さ」を理解する。

第1回授業実践は、生徒たちはプログラミング初心者であるため、Scratchの基本的な仕組みを押さえ、簡単なゲームを作成するなどして、Scratchを楽しむことをねらいとして設定した。また、プログラミングの重要な構成要素である「条件分岐」や「繰り返し」の学習も第1回授業実践で行う。第2回目の授業実践は、中学校では、数学を苦手としている生徒たちが多く、その中でも関数や確率を苦手とする生徒たちが多くと考え、プログラミング・シミュレーションを取り入れ、関数と確率の本質を掴ませることを目標とした授業を設定した。第3回授業実践は、第2回授業実践で学習したことの応用として、関数と確率の中間に位置するカオス理論に従う現象を取り扱い、関数という決定論的なものから、予測不可能で複雑なものがつくりだされることを、プログラミング・シミュレーションにより実現し、複雑さについて学習することを目標として設定した。

4.2 実践の内容

4.2.1 第1回授業実践の内容

まず、第1回授業実践について報告する。本実践は中学校1,2,3年生の10名を対象とし、Scratchを使ったプログラミングに入門する授業を行った。授業は50分×2コマ行った。本実践における生徒たちの達成目標は、プログラミングを通して簡単なゲームを作成し、プログラミングの構成要素について学び、コンピュータの特性について理解することである。前半50分は図形の描画等を通してScratchの基本操作の練習、後半50分は簡単なピンボールゲームの作成を主に行った。授業の流れは以下の通りである。

- (1)授業前アンケート
- (2)Scratchの紹介
- (3)Scratchで正方形、正三角形、正五角形の描画。
- (4)「繰り返し」を使った正多角形の描画。
- (5)ピンボールゲームのフローチャートの作成
- (6)フローチャートに沿ってプログラムを組む
- (7)完成したプログラムにアレンジを加える
- (8)授業後アンケート

4.2.2 第1回授業実践の分析結果

(1)のアンケート調査では、①学校の授業や家ではどんなことにパソコンを利用しているか、②パソコンしかできないことはどんなことだと思うか、の2点を質

問した。

①の主な回答は以下の通りである。

- ・調べもの ・動画再生 ・表計算
- ・表、グラフづくり ・カレンダー作り
- ・プログラム作り ・オンラインゲーム

②の主な回答は以下の通りである。

- ・インターネット ・動画再生 ・タイピング
- ・膨大なデータの集約 ・大きな画面の利用
- ・プログラム ・画像に動作をつける
- ・スマートフォンにはないアプリケーション
- ・必要な情報を瞬時に得ることができる

アンケートの結果から、情報収集や表計算等にコンピュータが多く活用されていることがわかった。プログラムについては、技術でプログラミング学習を既に行っている生徒から回答が得られた。

次に授業計画の項目ごとに考察を行う。(3)の正方形の描画では、正方形をスプライトに描かせるためには、スプライトを直進させ、90度回す作業を4回繰り返すと実行できることを確認し、生徒たちにプログラムを組ませた。続く正三角形では、スプライトを回す角度を60度にする生徒が多くみられ、どこを直したら正三角形を描くことができるか試行錯誤している様子がみられた。ここでは、もう一度、正三角形を描くにはスプライトを何度回したらよいか注意深く考えることで、自ら回す角度を120度に訂正する生徒が見受けられた。デバックを通して正確な手続きを試行錯誤している様子が見受けられた。(4)ではプログラミングの重要な要素である「繰り返し」を使用し、角度や繰り返す回数などの数値を変える、新しいブロックを組み込むことで様々な図形を生徒たちが描くことができた。(6)では「条件分岐」という、プログラミングにおいて重要な概念を理解できることがわかり、ヒントを与えながら「条件分岐」を使ってプログラムを組むことができたことが確認できた。(7)ではスプライトの数を増やすなどして「繰り返し」と「条件分岐」を使いこなす生徒も見られた。(8)の授業後アンケートでは、①Scratchの楽しかったところはどんなところか、②Scratchの難しかった所はどんなところか、③パソコンにしかできないことはどんなところだと思うか、の3点質問した。

①の主な回答は以下の通りである。

- ・自分の好きな命令を出せたこと
- ・ゲームを作れるところ ・図形を作れるところ
- ・いつも遊んでいるゲームの仕組みが少しわかった
- ・少しおかしくなったところが難しかった
- ・ネコの動きを考えること ・効果音の作成
- ・指示通りにネコが動くところ

②の主な回答は以下の通りである。

- ・「～したいときは」と考えたとき、どれを使えば良いかすぐにわからなかったところ
- ・知識がないとうまくいかない・自分で考えること
- ・図形の角度を考えるところ
- ・説明を聞けば難しくない ・たくさんあった
- ・どのようにすればその動きになるのか考えること
- ・「動き」や「制御」のピースを組み立てるところ

③の主な回答は以下の通りである。

- ・自分でゲームを作ることができる
- ・プログラム ・複雑なものが作れる
- ・調べることだけであった ・いろんなこと
- ・Google ・グラフ作り
- ・指示でキャラクターが動くこと

授業後アンケートから、多くの生徒が実現したい動きのプログラムを考えることや、プログラムが正常に作動しなかった際に、プログラムを修正することに苦戦していることがわかった。

第1回目の実践から、生徒たちはコンピュータの「繰り返し」「条件分岐」を扱い、手作業では行うことのできない膨大な回数の作業や、動作を正確に判断することができるという、コンピュータの役割の一端に触れることができたと考える。また、本実践では生徒たちが物事の手順を正確に考え、プログラムの改善に試行錯誤する様子が見られたため、長期的にプログラミング学習を行うことが論理的思考力や課題解決能力の育成にもつながることが期待できると考える。Scratch を用いたことで、楽しみながらプログラミングに入門し体験できたと考える。

4.2.3 第2回授業実践の内容

第2回授業実践では、中学校3年生を対象として、関数分野と確率分野にあたる授業を計画した。この授業における、生徒たちの達成目標は、①関数法則に従う事象は将来のことを予測できるものであり、一様分布という確率法則に従う事象は将来のことを全く予測することができないこと、②正規分布という確率法則に従う事象は、期待値と分散という概念を考えることで、ある程度予測することが可能である、という2点の知識を獲得することである。従来の関数や確率の学習では、問題を解き、正確な答えを導くことに重点が置かれており、本質についての学習の比重が少ないのではないかと考え、上記のような達成目標を設定した。授業の大まかな流れと内容については以下の通りである。

- (1) 関数法則に従う事象と確率法則に従う事象を比べる。関数法則に従うものは、ある値を一つ入力すると対応する唯一の値が出力されるため、1対1対応の決定論的なものになっており、将来の動きを100%予測することができること、確率法則に従うものは、将来の動きがランダムで予測することができない非決定論的で不確かなものであることを理解する。
- (2) コンピュータ上にランダムなものを発生させ、ホワイト・ノイズをシミュレーションする。簡単な法則から複雑な現象が作りだされることを体感する。
- (3) ホワイト・ノイズ、サイコロ、コイン等はそれぞれの動きや値の確率が全て等しく一様分布という確率法則に従っていることを理解する。
- (4) コインの表が出たら10、裏がでたら-10と設定して、コインを繰り返し、値を足し合わせていく。
- (5) コイントスの回数を無限回に近い回数まで増やし、コンピュータ上で試行ランダム・ウォークをシミュレーションする。
- (6) 試行回数を無限に増やしていくと、足し合わせた値が正規分布に従うことを理解する。
- (7) 正規分布という確率法則に従う事象はそれぞれの

値や動きの確率の濃淡があり、出やすさや出にくさというものがあることを理解する。

(8) 正規分布の分散という概念を考えることで、確率法則に従う事象がどれだけ不確かなのか予想することができることを理解する。

5. おわりに

第1回の授業実践により、子どもたちは簡単なゲームをプログラミングで作成できること、また効果音をつける、オブジェクトを増やすなどオリジナルのアレンジを加えプログラミングできることがわかった。また、インターネットや表計算ソフトの利用がコンピュータ利用のほとんどである子どもたちにとって、「繰り返し」や「条件分岐」といったコンピュータ本来の特性や役割を学習することができたと考える。

今後の課題としては、①Scratch とは、そもそも子ども自身で時間をかけてプログラミング学習を進めていく環境になっているが、学校現場では時間の制約が厳しいため、どのように教師が介入し運用していくか検討する必要がある、②数学などの実際の教科で運用する際も、プログラミング・シミュレーションには利点もあるが、1コマの短い時間の中で運用することは難しく、子どもたちに何をプログラミングさせるのか適切に決定する必要がある、の2点が挙げられる。

報告当日では、上に挙げた課題に対する報告、第2回と第3回の授業実践の詳細についての報告、授業で行ったプログラムとシミュレーションの実行結果について報告する。

6. 参考文献

- (1) Scratch : <http://scratch.mit.edu>
- (2) 阿部和弘：『小学生から始めるわくわくプログラミング』、日経BP社(2013)
- (3) S.Papert, 奥村貴世子訳：『マインドストーム 子ども、コンピュータ、そして強力なアイデア』、未来社(1982)
- (4) 富野友里恵, 高数学, 田川貴章：「Scratch によるπと無限の学習」(2009)
- (5) 手塚奈緒, 高数学, 小森隆正, 鈴木翔太, 早川裕貴：「Squeak によるプログラミング学習を通じた課題解決能力の育成」(2009)
- (6) 松原望：『入門確率過程』、東京図書(2009)