

小学校教員養成課程における プログラミング教育の指導力の育成

—— Viscuit と LEGO Mindstorms を用いた実践 ——

山 下 祐 一 郎

要旨：2020 年度から全面実施が予定されている小学校学習指導要領（平成 29 年 3 月公示）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を身に付けるための学習活動」が盛り込まれている。これに伴い、小学校教員養成課程では、プログラミング教育の実施に必要な知識・技能を修得するためのカリキュラムが求められている。そこで、本研究では、プログラミング教育に必要な知識・技能を大学生が修得するための授業計画を開発し、授業実践を行った。この授業実践では、プログラミング言語として Viscuit とレゴマインドストームを利用する。そして、大学 2 年生 2 名と大学 3 年生 2 名の計 4 名に対して全 15 回の授業を行った。授業を受講した学生らの自己評価の結果、4 名中 3 名がプログラミング技能の向上を自覚し、4 名中 3 名の学生が小学校におけるプログラミング教育に関する理解が深まったと回答した。また、4 名中 3 名が指導案略案にプログラミングを盛り込むことができた。

キーワード：プログラミング、小学校、教員養成、授業実践

1. 背 景

1.1. は じ め に

2020 年度から全面実施が予定されている小学校学習指導要領（平成 29 年 3 月公示）（文部科学省 2017a）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を身に付けるための学習活動」が盛り込まれている。そのため、小学校教員養成課程の大学生は、プログラミング教育を実施する能力を在学中に修得する必要がある。

小学校学習指導要領（平成 29 年 3 月公示）と併せて小学校学習指導要領解説総則編（文部科学省 2017b）を概観する。まず、小学校学習指導要領解説総則編によると、「小学校においては特に、情報手段の基本的な操作の習得に関する学習活動及びプログラミングの体験を通して論理的思考力を身に付けるための学習活動を、カリキュラム・マネジメントにより各教科等の特質に応じて計画的に実施すること」とある。ここで記述されている「プログラミングの体験を通して身につける論理的思考」はプログラミング的思考と呼ばれており、小学校学習指導要領解説総則編においては、「自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えて

いく力」と定義されている。次に、小学校におけるプログラミング教育の注意点として「小学校段階において学習活動としてプログラミングに取り組むねらいは、プログラミング言語を覚えたり、プログラミングの技能を習得したりといったことではなく」とある。そのため、日本の多くの高等教育機関で行われているプログラミング教育とは異なる。日本の高等教育におけるプログラミング教育では、プログラミング言語やフローチャート、UML (Unified Modeling Language) などの修得が目的とされることが多い。さらに、小学校学習指導要領解説総則編では、プログラミング的思考力を育むとともに「プログラムの働きやよさ、情報社会がコンピュータをはじめとする情報技術によって支えられていることなどに気付き、身近な問題の解決に主体的に取り組む態度やコンピュータ等情報社会の基盤となっている情報技術に対する関心・意欲・態度を育むこと」とあり、情報技術に対する関心・意欲・態度についても言及されている。また、小学校学習指導要領解説総則編では、「教科等で学ぶ知識及び技能等をより確実に身に付けさせること」という文言も加えられている。プログラミング教育は各教科の中で実施することとされているため、プログラミングを取り入れた教科が、プログラミングによってわかりやすくなることが望まれている。なお、各小学校のプログラミング教育の実施方法については、「教科等における学習上の必要性や学習内容と関連付けながら計画的かつ無理なく確実に実施されるものであることに留意する必要があることを踏まえ、小学校においては、教育課程全体を見渡し、プログラミングを実施する単元を位置付けていく学年や教科等を決定する必要がある」とされている。つまり、カリキュラム・マネジメントを活かした計画的な教育課程の実現が望まれている。

1.2. プログラミング教育の実施例

小学校のプログラミング教育に関する実践例が、文部科学省より例示されている。小学校学習指導要領解説総則編によると「算数科、理科、総合的な学習の時間において、児童がプログラミングを体験しながら、論理的思考力を身に付けるための学習活動を取り上げる内容やその取扱いについて例示しているが（第2章第3節算数第3の2(2)及び同第4節理科第3の2(2)、第5章総合的な学習の時間第3の2(2)）」とある。

まず、算数の例である。小学校学習指導要領の第2章第3節算数第3の2(2)によると、「例えば第2の各学年の内容の〔第5学年〕の「B 図形」の(1)における正多角形の作図を行う学習に関連して、正確な繰り返し作業を行う必要があり、更に一部を変えることでいろいろな正多角形を同様に考えることができる場面などで取り扱うこと」とある。つまり、プログラミングによって正多角形を作図することができる。さらに、プログラムで作図した正多角形であれば、角数の変更が容易に行うことができる。この機能を利用することで、プログラミングの体験と同時に、正多角形の理解を深めている。

次に、理科の例である。小学校学習指導要領の第2章第4節理科第3の2(2)によると、「例えば第2の各学年の内容の〔第6学年〕の「A 物質・エネルギー」の(4)における電気の性質

や働きを利用した道具があることを捉える学習など、与えた条件に応じて動作していることを考察し、更に条件を変えることにより、動作が変化することについて考える場面で取り扱うものとする」とある。例示されているような電気の性質に関するシミュレーションは、電気の理解を深めるためによく利用されている教材のひとつである。児童が主体となってこのようなシミュレーションプログラムを作成することで、電気の理解を深めるとともにプログラミングの体験を行う。

最後に、小学校学習指導要領の第5章総合的な学習の時間第3の2(2)によると、「情報に関する学習を行う際には、探究的な学習に取り組むことを通して、情報を収集・整理・発信したり、情報が日常生活や社会に与える影響を考えたりするなどの学習活動が行われるようにすること」とある。例えば、ブロックなどでロボットを組み立て、そのロボットをプログラムで意図した動作をさせるなどの活動が挙げられる。

ところで、小学校学習指導要領解説総則編によると「例示以外の内容や教科等においても、プログラミングを学習活動として実施することが可能であり、プログラミングに取り組むねらいを踏まえつつ、学校の教育目標や児童の実情等に応じて工夫して取り入れていくことが求められる」とある。つまり、解説での例示は、算数及び理科、総合的な学修のみであるが、それら以外の教科でも実施が可能であることを示している。そこで、プログラミング教育実践ガイド（文部科学省 2015）においても小学校における実践事例4点を紹介しているため、これらの事例を概観する。

1点目は、小学校1年生を対象とした生活科、特別活動の授業である。プログラミング言語である Viscuit（開発元 合同会社デジタルポケット）を使用し、しゃくとり虫の絵をプログラミングで動かすという実践である。

2点目は、小学校4年生を対象とした図画工作の授業である。プロロボ USB プラス（開発元 山崎教育システム株式会社）という車型の教材を使用した実践である。紙粘土による寿司ネタの作成を行い、プロロボを使って目的の場所にお寿司を運ぶ。このため、プロロボが動作をするためのフローチャートを考え、実際にセンサカー（車のこと）を動かすことを通して、プログラミングの仕組みを体験的に理解する。

3点目は、小学校4年生及び5年生、6年生を対象とした総合的な学習の時間での実践である。レゴマインドストーム（開発元 LEGO）を用い、センサやアームを使って災害現場から救助して病院に運ぶという課題に対して、課題解決のためのプログラムを考えるという実践である。

4点目は、小学校6年生を対象とした総合的な学習の時間である。Scratch（開発元 MIT メディアラボ）で、自分の調べた人物について表現し、プログラミングのしくみを体験的に理解するという実践である。

プログラミング教育実践ガイドが示す通り、小学校でのプログラミング教育において使用されるプログラミング言語は多様である。これは、プログラミング言語の指定が無いためである。そのため、プログラミング教育が始まった場合、使用するプログラミング言語に関しては、各小学校または各教員などの任意となり、地域差ができる可能性が高い。

1.3. プログラミング言語

プログラミングとは、プログラムを作成することである。そして、プログラムとは、人間がコンピュータへ与える指示のことである。つまり、プログラミングとは、人間の意図した処理を行うようにコンピュータへ指示を与える行為である。そして、プログラミングを行うためのツールがプログラミング言語である。プログラミング言語は、コンピュータに対して動作などを適切に指示するために用いられる言語である。主にコンピュータ上で使用することができる。プログラミング言語が必要な理由は、コンピュータへの指示を人間が理解することにある。コンピュータに解釈実行が可能な命令は、0 と 1 から構成される機械語である。しかし、人間が 0 と 1 だけの機械語で指示をすることは難しい。そこで、人間が解釈しやすい単語や文法などで書かれた指示を機械語に翻訳するという手法が取られている。この人間が解釈しやすい指示を作成するための道具がプログラミング言語である。

次に、プログラミング言語で書かれた指示を例示する。例えば、最も普及しているプログラミング言語のひとつである C 言語を用いて、コンピュータのディスプレイに「Hello Word」と表示するプログラムは以下になる。

```
#include<stdio.h>
int main()
{
    printf("Hello World \n");
    return 0;
}
```

「#include<stdio.h>」や「int main()」などは、C 言語のプログラムに必ず必要な記述であり、構文のルールとなっている。そして、「printf」などは、コンピュータに指示をするための命令文である。C 言語を利用するためには、これらのルールと命令文をある程度覚えなくてはならない。

ところで、プログラミング言語でソフトウェアを開発するためには、IDE (Integrated Development Environment) または統合開発環境と呼ばれるシステムのサポートを受ける場合が多い。IDE のサポートを用いることで、効率的にプログラミングを実施できるようになる。しかし、IDE の利用にも専門的な知識が不可欠であり、小学生が利用することは難しい。そこで、教育に利用されるプログラミング言語は、プログラミング言語自体をより平易かつ直感的にすることが考えられている。例えば、Seymour Papert and Cynthia Solomon (1971) は、それまでよりも平易な学習用プログラミング言語を開発し、プログラミング言語を教えることと学ぶことに利用している。このプログラミング言語は LOGO と呼称されている。

近年、さらに平易かつ直感的に使用可能なプログラミング言語として、ビジュアルプログラミ

ング言語が提案されている。ビジュアルプログラミング言語は、C言語のように文字による命令文で構成するプログラムと異なり、ブロックや絵などのパーツを組み合わせてプログラミングを行う。そのため、ビジュアルプログラミング言語の多くは、printfなどの命令文を覚えなくとも直感的にプログラミングを行うことができる。さらに、C言語の「#include<stdio.h>」や「int main()」などのような、プログラミングのためのルールが少ない。また、正しいプログラムを記述していなくとも、何らかのアクションが起きて利用者の注意を惹くための工夫がされている。

以上を踏まえて、小学校で利用するプログラミング言語としては、ビジュアルプログラミング言語が望ましいと判断される。そこで、ビジュアルプログラミング言語について概観する。

1) Scratch (スクラッチ)

Scratch は、特に8歳から16歳向けにデザインされているが、年代を問わず多くの人々に利用されている。利用場所に関しては、自宅や学校、博物館、美術館、図書館、公民館、コミュニティセンターなどで利用されている。さらに、150以上の国で使用されており、40カ国語以上に翻訳されている(MIT メディアラボ 2017)。

Scratch の画面サンプルを図1に示す。図1のプログラムの箇所でするように、Scratch のプログラムは様々なブロック型のパーツの組み合わせである。パーツは、図1の「パーツ一覧」に用意されている。一覧に用意されているパーツをドラッグして移動し、プログラムに使用する。こうして作成したプログラムによって絵が動く。図1の絵は著者が書いたりんごであるが、スクラッチのライブラリにデフォルトで用意されている絵を表示させることもできる。

2) Viscuit (ビスケット)

Viscuit は就学前児童に対しても教育実績があるプログラミング言語であり、機能が少ないため比較容易に修得が可能である。Viscuit の画面サンプルを図2に示す。図2の「描画機能」と示された部分を押下すると、絵を描画することができる。図2では例として三角形を描画している。描画した三角形を、図2中の「絵(この絵が動く)」という箇所配置している。ここに配置された絵が、作成したプログラムに応じた動きをする。プログラミングの方法は、図2の「プログラム」という箇所示している。プログラムの箇所にメガネのような絵があり、メガネのレンズに当たる部分に三角形が配置されている。メガネの左右の三角形の位置は少し異なっており、



図1 Scratch の画面サンプル

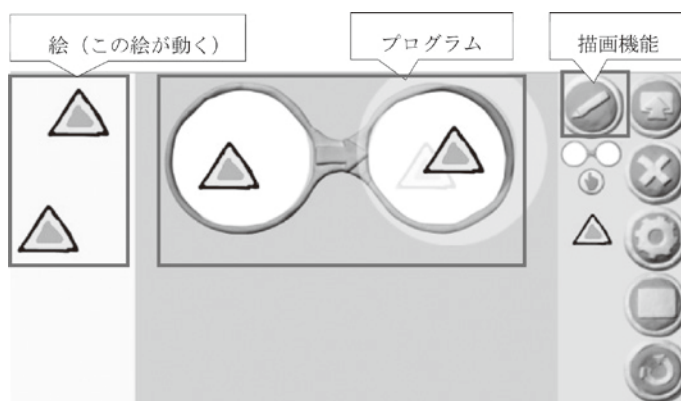


図2 Viscuit の画面サンプル

この差だけ位置が変化することを示している。すなわち、絵 (この絵が動く) の三角形の位置が変わるということは、三角形が動くということである。

Viscuit のプログラムは、基本的にメガネという仕組み一つだけである。このメガネを利用することで、単純なプログラムからとても複雑なプログラムまで作ることができる。仕組みは単純だが、組み合わせ方が様々なので複雑なことができる (デジタルポケット 2017)。

3) LEGO Mindstorms (レゴ マインドストーム)

レゴマインドストームのサンプル画面を図3に示す。レゴマインドストームは、ブロック、センサ及びモーターなどで本体を作り、そして、モーターとセンサを制御動作するためのプログラムを作成し本体へ組み込む。図3の左側にブロックで作成した車の写真を載せている。なお、ブロックで作成できる形状は様々であり、例では車型を示しているが、2足歩行のロボット型やヘビ型なども組み立てられる。プログラムの例は、図3の右図である。Scratch と同様に命令が記述されたパーツが用意されており、それを組み合わせることでプログラムを作成することができる。なお、プログラミングはパソコンやタブレットなどのデバイスを用いて行い、そのプログラ

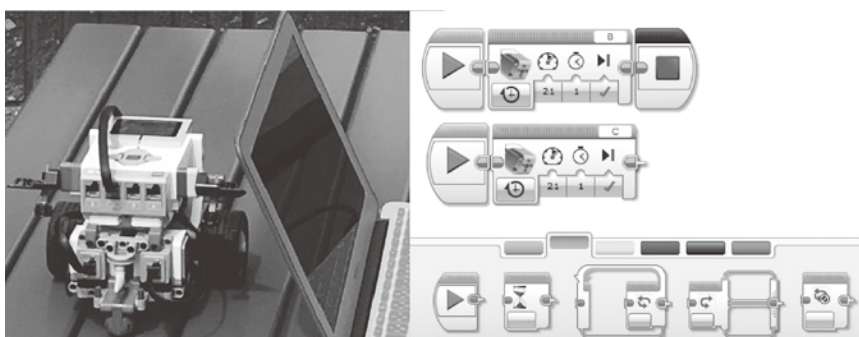


図3 レゴマインドストームの画面サンプル

ムを本体にインストールする。図3の左側の写真では、ブロックで作成した車とパソコンをケーブルで接続してプログラムをインストールしようとしている。レゴマインドストームは、ブロックで組み立てた車を動かすため、直進走行するようなプログラムを組んだつもりでも、路面の状態や本体の構造によっては直進しないという難しさがある。

2. 目 的

本研究は、小学校教員養成課程の大学生がプログラミング教育を実施できるようになることを目的としている。

山下（2017）は、小学校教員養成課程の学生に対して、Scratchを用いたプログラミング教育の実践研究を行った。この実践研究では、プログラミング教育を実施するために必要な知識・技能として以下の3点を挙げている。

- ・授業力（指導力）
- ・小学校におけるプログラミング教育の知識
- ・プログラミングの知識・技能

まず、1点目の授業力に関しては、プログラミング教育に限らず授業を実施するために必要な力である。小学校教員養成課程において授業を実施する力の育成は、各教科の概論や演習、実習などを通じて行われる。そのため、単一の授業のみでは育成が難しいと判断される。次に2点目の小学校におけるプログラミング教育の知識については、小学校学習指導要領及び小学校学習指導要領解説に関する知識となる。最後に3点目は、プログラミングの知識・技能である。小学校教員養成課程の学生は、プログラマほどのプログラミングに関する知識は必要ない。しかし、教員としてプログラミング教育を実現するためには、ある程度のプログラミングに関する知識・技能が必要である。Scratchによる授業実践では、学生がScratchの技能をある程度修得ができたものの、小学校におけるプログラミング教育の特徴を理解することについては難しく、プログラミング教育を計画・実施するまでには至らなかった。

この理由として、小学校教員養成課程の学生が修得するには、Scratchが高難易度であることが挙げられる。先に示したようにScratchによるプログラミング技能の修得は可能であったものの、修得に時間がかかってしまい、指導要領などの理解までは及ばなかったと判断される。一般的に小学校教員を目指す学生の多くは、教科化されている科目の学修や実習などを重視する傾向がある。つまり、学生たちにとってプログラミング教育に関する学修の優先順位は高くない。そのため、学生らがより簡単に修得することが可能なプログラミング言語が必要である。より簡単にプログラミングを修得できれば、学生らは指導要領などを理解する時間の確保ができる。そして、指導要領をはじめとする小学校におけるプログラミング教育の知識を得られれば、プログラミング授業の計画・実施が可能となると予想している。

本研究では、Scratch よりは簡単に修得可能なプログラミング言語として、Viscuit に着目した。前章で記述した通り、Viscuit は就学前児童に対しても利用実績があり、Scratch よりも前提となる知識が少なく、より直感的に使用することができる。そのため、学生が短い時間で修熟できる可能性がある。

ところで、Scratch と Viscuit はディスプレイに表示した絵を動かす機能が最も基本的である。一方で、プログラミング教育の事例を概観すると、プログラムを用いてロボットを動かすということを取り入れた授業も多く存在する。実体のあるロボットには、周辺環境を考慮した指示を与える必要があるため、ディスプレイの絵を動かすこととは異なる試行錯誤が必要となる。例えば、車型のロボットを動かす場合、デコボコの土面と舗装されたアスファルトの路面では、同じプログラムを与えても同じようには動かないことがある。そのため、本研究では、小学校教員養成課程の大学生に対して、実体のある機械をプログラムで動かす体験が必要であると判断した。そこで、Viscuit の次にレゴマインドストームの体験を取り入れる。レゴマインドストームを選択した理由は、世界的な規模でシェアが拡大しており、2016 年上半期の売上は前年比で 10% 増加している (LEGO.com 2016)。そのため、小学校教員養成課程の学生が将来利用する可能性が高い。そして、利用率に比例してノウハウが蓄積されているからである。

以上をまとめると、小学校教員養成課程の学生に対して、小学校におけるプログラミング教育に関する知識、及び、プログラミングの知識・技能を修得させることが必要である。そのことに加え、プログラミング教育の模擬授業の体験が必要であると考えた。このことを実現するため、本研究では新しい授業計画を開発して対応する。本研究における授業計画の特徴は 3 点ある。1 点目は、より簡単に修得が可能なビジュアルプログラミング言語である Viscuit を利用する。2 点目は、より深い学びを得るために、レゴマインドストームを用いることにある。3 点目は、学生らが模擬授業を設計して実施する。

3. 授業計画とルーブリック

3.1. 授業計画

本研究は、小学校教員養成課程において、プログラミング教育に関する知識、及び、プログラミングの知識・技能を修得するための授業計画を開発することにある。そのため、プログラミング言語として、ビジュアルプログラミング言語の Viscuit とレゴマインドストームを用いる。さらに、学生らに模擬授業を実施させる。これらを踏まえて本研究で開発した授業計画を以下に示す。

- ・第 1 回：オリエンテーション
- ・第 2 回：Viscuit の概要・練習
- ・第 3 回：Viscuit の練習・作成

- ・第4回：Viscuitの作成・発表
- ・第5回：プログラミング教育の意義
- ・第6回：レゴマインドストームの概要
- ・第7回：レゴマインドストームの組み立て（1）
- ・第8回：レゴマインドストームの組み立て（2）
- ・第9回：レゴマインドストームのプログラミング（1）
- ・第10回：レゴマインドストームのプログラミング（2）
- ・第11回：レゴマインドストームの発表
- ・第12回：模擬授業指導案の立案（1）
- ・第13回：模擬授業指導案の立案（2）
- ・第14回：模擬授業の実施
- ・第15回：授業のまとめ

この授業計画は、1回90分の授業で15回分を想定している。これは、多くの大学で半期の授業に相当している。

本研究で開発した授業計画では、第2回から第4回までの全3回がViscuitに関する授業である。この段階ではプログラミングの初学者を想定して、プログラミングの体験を主な目的としている。そこで、第2回（Viscuit第1回）では、Viscuitの使い方の説明を実施する。第2回から第3回（Viscuit第2回）にかけてはViscuitの操作方法の練習であり、教員が示した通りにプログラミングを行う。次に、テーマを与え、そのテーマに関するプログラミングを学生が独力で行う。本実践ではテーマとして「たまご割りゲームの作成」とした。たまご割りゲームの条件は、たまごの絵を描き、たまごを割るという操作を含むことである。その他の仕様については指定を行わなかった。たまご割りゲームの完成後、学生が作成したプログラムを操作して発表を行う。さらに、第3回後半から第4回（Viscuit第3回）にかけては、文章を読み、その文章の情景を絵とプログラミングで表わす。そして、プログラムの作成後に、発表を行う。

第5回は、小学校学習指導要領（平成29年3月公示）のプログラミング教育に関する説明を行う。学生はViscuitを通じてプログラミングを体験しており、プログラミングのイメージを持っている。そのため、プログラミングのイメージと関連付けて、プログラミング教育を理解することが可能となる。

第6回から第11回は、レゴマインドストームを用いた授業である。本研究における授業計画では、受講生として小学校教員養成課程の大学生を想定している。そこで、学生らが主体的にプログラミングに関する知識を自ら調べ、それを利用する態度を養う必要があると判断した。そのため、レゴマインドストームのプログラム言語の詳細な使い方の説明はせず、代わりにレゴマインドストームのプログラミングに関する情報検索の方法を示した。

レゴマインドストームで学生らに示した課題は、「ブロックで任意の形状の本体を組み立て、

校庭で8字に動かすこと」である。動かす距離は50m程度である。

第6回（レゴマインドストーム 第1回）で、プログラミングのための準備を行う。例えば、プログラミングツールのインストールを行ったり、本体にプログラムを組み込む方法などを説明する。第7回（レゴマインドストーム 第2回）と第8回（レゴマインドストーム 第3回）を本体の組み立てとし、第9回（レゴマインドストーム 第4回）と第10回（レゴマインドストーム 第5回）をプログラミングとしている。ただし、最終的に本体とプログラムが完成すれば良いものとし、厳密に区分したわけではない。

第12回から第14回は、模擬授業である。第12回（模擬授業 第1回）と第13回（模擬授業 第2回）でプログラミング教育に関する模擬授業の計画を立てる。模擬授業では、使用するプログラミング言語は任意とし、実施する教科も任意とした。なお、模擬授業にあたっては、指導案（略案で可とした）を作成する。そして、第14回（模擬授業 第3回）において、授業の導入部分の10分間を実施するものとした。

3.2. ルーブリック

本研究では、プログラミング教育のルーブリックを作成する。ルーブリックを学生らに示すことで、学修目標を明確に示す。さらに、学生らはルーブリックを用いて自己評価を行うことで、授業目標に対する自分の修熟度を客観視することが可能となる。本研究におけるルーブリックを以下に示す。このルーブリックは、5段階の形式であり、5に近いほど授業目標が達成されていることになる。

① 小学校におけるプログラミング教育の意義

5	4	3	2	1
4に加えて、初等教育においてプログラミング教育を実施できる	3に加えて、コンピュータに意図した処理を行わせることについて説明できる	2に加えて、指導要領のプログラミングについて説明できる	小学校学習指導要領（平成29年3月公示）を読んだことがある	プログラミング教育について説明できない

② プログラミング技能に関する内容（プログラム言語の種類は問わない）

5	4	3	2	1
4に加えて、プログラムの作り方に関して指導ができる	3に加えて、少し複雑なプログラミングができる	2に加えて、簡単なプログラミングができる	プログラミングの概要について説明できる	プログラミングの概要について説明できない

③ 小学校におけるプログラミング教育の指導力

5	4	3	2	1
4に加えて、設計したプログラミングを取り入れた授業を実施できる	3に加え、自分の好きな教科で、プログラミングを取り入れた授業を設計できる	2に加え、自分の好きな教科であれば、児童に実習・演習を行わせることができる	自分の好きな教科であれば、授業を行うことができる	自分の好きな教科であっても、授業を行うことができない

学生らはこれらのルーブリックを用いて自己評価を行う。自己評価を行うタイミングは、授業の始まりである第1回、Viscuitの終了後である第5回、レゴマインドストームの終了後である第11回、授業の最後である第15回の計4回である。

4. 授業実践

本研究では、小学校教員養成課程に在籍し、小学校教員の免許取得を目指す学生を対象に授業実践を行った。受講した学生は、大学2年生2名、大学3年生が2名の計4名である。ただし、2年生と3年生は別々に授業を行った。いずれの学生も教育実習に行っていない。

本研究における授業実践では、前章で示した授業計画とルーブリックを用いている。この授業実践の様子を図4に示す。図4の左側の写真は、Viscuitで開発したプログラムを電子黒板に表示して発表している様子である。図4の右側の写真は、レゴマインドストームの組み立てを行っている様子である。

本研究では、学生らが模擬授業を行った。各模擬授業の概要を示す。

(1) 学生A

- ・プログラミング言語：Scratch
- ・教科：算数
- ・概要：四角形と三角形の面積を求める。面積を計算するプログラムを作成し、長さなどの数

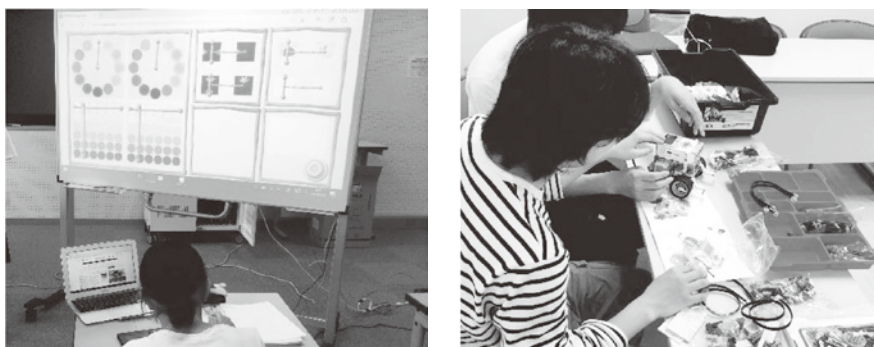


図4 授業実践の様子（左 Viscuit、右 レゴマインドストーム）

値を変えながら面積の変化を確認する。

(2) 学生 B

- ・プログラミング言語：Scratch
- ・教科：社会
- ・概要：戦国時代から安土桃山時代、江戸時代と3つの時代の中で活躍した偉人について調べ、偉人の業績を紹介するプログラムを作成し、理解を深める。

(3) 学生 C

- ・プログラミング言語：Viscuit
- ・教科：国語
- ・概要：漢字の書順を表示するプログラムを作成し、書順を確認する。

(4) 学生 D

- ・プログラミング言語：Viscuit
- ・教科：国語
- ・概要：教科書の文章を読み、その情景を絵と動きで表現するためのプログラムを作成する。

5. 結 果

まず、学生によるループリックの自己評価の推移を表1に示す。表1の学生Aから学生Dは授業実践に参加した各学生の識別を表している。そして、学生Aから学生Dの横の数値は、各学生が回答したループリックの番号である。ループリックの内容は前述した通りであり、数値が大きいほど知識・技能の修得が進んでいることを示す。また、表1中の①から③は、ループリックの種類を示している。すなわち、① 小学校におけるプログラミング教育の意義、② プログラミング技能に関する内容（プログラム言語の種類は問わない）、③ 小学校におけるプログラミング教育の指導力を示している。授業回の1, 5, 11, 15は、ループリックの自己評価を行った授業回を示している。第1回は初回、第5回はViscuitの終了時、第11回はレゴマインドストームの終了時、第15回は模擬授業の終了時であるとともに授業全体の終了時である。

表1 ループリックの自己評価結果

	①				②				③			
授業回	1	5	11	15	1	5	11	15	1	5	11	15
学生 A	2	3	3	3	1	3	3	4	2	2	4	4
学生 B	2	2	2	2	1	3	1	3	1	1	1	2
学生 C	1	1	3	3	1	3	3	3	1	1	1	2
学生 D	1	3	3	4	1	3	3	3	1	1	1	1

次に、本研究では、第 14 回の模擬授業終了後に、授業の自己評価と他者評価をアンケート形式で実施した。このアンケートの結果を表 2 に示す。表 2 中の ① から ⑥ はアンケートの評価項目であり、内容を以下に示す。なお、各項目の回答は 4 件法であり、4 に近いほど評価が高く、1 に近いほど評価低い。

- ①（自己評価）受講生はあなたの授業を楽しむことができましたか。
- ②（自己評価）小学生がこの授業を楽しむことができますか。
- ③（自己評価）小学生がこの授業で、プログラムの作成を通じてコンピュータに意図した処理をさせることが身につくと思いますか。
- ④（他者評価）授業を楽しむことができましたか。
- ⑤（他者評価）小学生がこの授業を楽しむことができますか。
- ⑥（他者評価）小学生がこの授業で、プログラムの作成を通じてコンピュータに意図した処理をさせることが身につくと思いますか。

表 2 の学生 A から学生 D は各学生の識別を表している。そして、学生 A から学生 D の横に記されている数値のうち、「自己による評価」は各学生が自己評価で回答した番号を示している。「他者による評価」は、他者からの模擬授業の評価である。学生 A と学生 B が、学生 C と学生 D がペアとなっているため、他者による評価の欄にはそれぞれの相手からの評価が記入されている。

6. 考 察

本研究で開発した授業計画について考察する。本研究で開発した授業計画は、プログラミング教育を実施するために、小学校におけるプログラミング教育の知識、プログラミングの知識・技能の 2 点を修得することを目標としている。

まず、小学校におけるプログラミング教育の知識については、表 1 の ① に関する自己評価を第 1 回と第 15 回で比較をすると概ね向上している。つまり、小学校におけるプログラミング教育の概要に関する知識を、ある程度は修得できたものと判断される。これは、授業計画の中で小学校学習指導要領（平成 29 年 3 月公示）の説明を行った効果である。

表 2 模擬授業の評価結果

項目	自己による評価			他者からの評価		
	①	②	③	④	⑤	⑥
学生 A	3	4	3	3	3	4
学生 B	2	2	2	4	4	3
学生 C	3	3	3	4	4	4
学生 D	2	3	4	4	4	3

次に、プログラミングの技能に関しては、表1の②の数値を第1回と最終回を比較すると、全員の自己評価が向上している。特に、表1②の第1回からViscuit終了後の第5回にかけて自己評価が2段階向上している。この理由として、Viscuitが修得しやすかったと判断される。実際、「授業の中で使えたから（簡単なプログラミングができると思う）」という趣旨のコメントが得られている。

また、プログラミング教育の指導力については、受講生の多くが表1の③の第15回で、2の「自分の好きな教科であれば、授業を行うことができる」をクリアしている。この理由は、学生らが、他の授業や演習において模擬授業の実施を経験したことが挙げられる。また、本授業計画に基づいて、学生らが模擬授業の指導案作成及び実施を行ったために自信がついたものと判断される。実際、「授業自体は他の講義で模擬授業を行って、プログラミングについては簡単なものなら取り入れることができそうだった」と「授業の中で、10分間だが、導入の部分の授業をすることができたから」など、実際にやってみたためにできると感じたという趣旨のコメントが多かった。さらに、自己評価の向上がみられたのは第11回から第15回（模擬授業の終了後）であり、模擬授業の効果が現われている。しかし、学生らは10分間の模擬授業しか行っておらず、実際に児童に対して実習を行っていないため3の「2に加え、自分の好きな教科であれば、児童に実習・演習を行わせることができる」までは到達することが難しかったようである。また、「10分程度の授業なら行えるが、1時間分の授業を行うことがまだできない」というコメントもあり、模擬授業の限界を示すコメントもあった。加えて、表2を概観すると、他者からの評価は全員が3以上を獲得しており、小学生が楽しんでプログラミングの実習を行えると予想される授業内容であったと判断される。本研究では、小学校におけるプログラミング教育の知識とプログラミングの知識・技能の修得が進めば模擬授業が可能になると推測していた。この推測が概ね正しかったと判断される。

ところで、表1の①の第5回から第11回（レゴマインドストームの終了時）にかけて、1名の自己評価が上昇しているものの、他3名の自己評価は変わっていない。本研究における授業計画では、レゴマインドストームの体験を通じてプログラミング教育をより深く理解することを狙ったが、改善が必要であると判断される。この理由として、学生らにとってレゴマインドストームの課題が難しく、課題の取組に多くの時間が割かれた。そのため、レゴマインドストームの実習とプログラミング教育を関係させて考える時間が無かった。実際、レゴマインドストームの課題を最終的にクリアできた学生はいなかった。

今後は、レゴマインドストームを使った課題について、プログラミングの方法をどこまで学生らに提示するのかを検討していきたい。提示しすぎると試行錯誤をする機会を奪ってしまうが、提示が少なすぎると課題をクリアできなくなる。この点に関する配慮が必要である。

参 考 文 献

- 1) デジタルポケット 2017, ビスケット viscuit コンピュータは粘土だ!!, <http://www.viscuit.com/> (参照日 2017 年 08 月 31 日)
- 2) LEGO.com 2016, レゴ・グループ, 2016 年上半期の世界売上高が 10% 増加, <https://www.lego.com/ja-jp/aboutus/news-room/2016/september/interim-result> (参照日 2017 年 08 月 24 日)
- 3) MIT メディアラボ 2017, Scratch について, <https://scratch.mit.edu/about> (参照日 2017 年 08 月 25 日)
- 4) 文部科学省 2015, プログラミング教育実践ガイド, http://jouhouka.mext.go.jp/school/programming_zirei/ (参照日 2017 年 08 月 14 日)
- 5) 文部科学省 2017a, 学習指導要領「生きる力」小学校学習指導要領, http://www.mext.go.jp/a_menu/shotou/new-cs/1384661.htm (参照日 2017 年 08 月 14 日)
- 6) 文部科学省 2017b, 小学校学習指導要領解説 総則編, http://www.mext.go.jp/a_menu/shotou/new-cs/1387014.htm (参照日 2017 年 08 月 14 日)
- 7) Seymour Papert and Cynthia Solomon 1971, Twenty Things To Do with a Computer, AI Laboratory MIT, Artificial Intelligence Memo No 248, pp. 1-40
- 8) 山下祐一郎 2017, Scratch を利用した教員養成課程におけるプログラミング教育指導力の育成, 東北福祉大学教職・教育研究 2016, pp. 189-196