

小中学校で活用可能な複数言語に対応した プログラミング教材の開発

西原 貴史*・中原 久志**・塚本 光夫***

Development of multi-language programming software for elementary or junior high school students

Takashi Nishihara*, Hisashi Nakahara** and Mitsuo Tsukamoto***

(Received September 29, 2017)

A multi-language software application of visual type to learn programming education for beginners has been developed. Block-type visual programming language is suited to the teaching material of programming education for elementary or junior high school students as in many cases. However, even if it is suitable for beginners, it is necessary to develop to the learning of descriptive language to obtain high interest. Therefore, the feature of this software application has tools that can learn both block-type graphic and text based programming language. Other features of this software application are as follows: simple input by selection, simultaneous display of text based language, learning by preset task, function to notify that debugging is necessary.

Key words : multi-language programming, block-type programming, text based programming language

1. はじめに

1.1 研究の目的

本研究の目的は、プログラミング初学者向けの複数言語に対応したビジュアルプログラミングツールの開発を行うことである。加えて、その活用可能性を示す。具体的には、タブレット端末やPC上で動作可能な形態であり、視覚的な操作でプログラミングが可能であると同時に、言語の記述についても学習可能な複数タスクのツールを開発する。

1.2 研究の背景

高度情報通信ネットワーク社会推進戦略本部において、第8回新戦略推進専門調査会人材育成分科会（平成27年4月）では、プログラミング教育の推進が提言されている¹⁾。その中では、プログラミング教育を基礎学力として、また初等中等教育における授業科目として設置することについても言及されており、プログラミング教育の学校教育における意義はより高まっ

ていると言える。

そのような中、文部科学省では、「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」において、小学校段階で育成すべき資質・能力と効果的なプログラミング教育の在り方や、効果的なプログラミング教育を実現するために必要な条件整備等について検討を行っている。その中で、プログラミング教育を「子供たちに、コンピュータに意図した処理を行うよう指示することができるということを体験させながら、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としての「プログラミング的思考」などを育むこと」と位置付けている²⁾。また、プログラミング的思考を、「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良いのか、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」と定義している²⁾。このプログラミング的思考は、「コンピューショナル・

* 熊本大学教育学研究科

** 大分大学教育学部

*** 熊本大学教育学部

シンキング」の考え方を踏まえつつ、プログラミングと論理的思考との関係を整理しながら定義されている。

このようなプログラミング教育の中で育成を目指す資質・能力として、①知識・技能、②思考力・判断力・表現力、③学びに向かう力・人間性等が挙げられている²⁾。①の知識・技能は、小学校においては、「身近な生活でコンピュータが活用されていることや、問題の解決には必要な手順があることに気付くこと」、中学校においては、「社会におけるコンピュータの役割や影響を理解するとともに、簡単なプログラムを作成できるようにすること」、高校においては、「コンピュータの働きを科学的に理解するとともに、実際の問題解決にコンピュータを活用できるようにすること」が目標として位置づけられている。②の思考力・判断力・表現力に関しては、「発達の段階に即して、「プログラミング的思考」を育成すること」が示されている。③の学びに向かう力・人間性では、「発達の段階に即して、コンピュータの働きを、よりよい人生や社会づくりに生かそうとする態度を涵養すること」が掲げられている。

1.3 先行研究の整理

1.3.1 初学者へのプログラミング教育の困難さ

西田らによると、プログラミング初学者にとっては、「タイプミスに基づく文法エラーの多さとそれらに起因する意味の理解できないコンパイルエラーメッセージ」や「論理エラーの修正の難しさ」などが、学習の継続を困難とさせていることが指摘されている³⁾。このような学習の継続の困難さを解消するものとして、ブロック型などのビジュアルなプログラミング環境の導入⁴⁾や、江木・竹内が開発したトレースを指導するデバッグ支援システムなどが挙げられる⁵⁾。しかし、何かしらのプログラミング言語の学習を通して行われるアルゴリズムに関する理解が得られないままになることが危惧される。

小学生に対するプログラミング教育を図るという視点からは、ブロック型のプログラミングソフトやツールが、児童にとっても指導者である教員にとっても取り組みやすいと考えられる。例えば、Scratchでは、ブロックに基づく直感的なプログラミングができるとともに、高度な作品作りも可能である。また、児童の興味・関心に応じた多様な作品作りや、インターネット上での共有に基づく、作品を介したコミュニケーションが可能など、様々なメリットがある。

しかし、このような初学者が活用可能なプログラミングツールの学習は、子どもたちにとって興味や生活、社会の課題解決と関連を持たせることが困難である。また、経験や事前の学習との関連を持たせづらく、「は

いまわり」をひきおこす可能性がある。うまくいかなかったときのガイドや、うまくいったときの探求の促進を行うことができないことも危惧される。

ブロック型のプログラミングは、あくまでプログラミング言語を用いた記述型のプログラミングの導入として、興味・関心を高めるものとして位置づけられなければ、中学校技術科「D 情報に関する技術」や高等学校情報やコーディングの学習に結びつきにくい。

1.3.2 児童・生徒を対象としたプログラミング教育

児童・生徒を対象としたプログラミング教育は数多くその実践事例が報告されている。例えば、森らは、Scratchを用いて、小学校4年生向けにプログラミングの授業をデザインし、実践を行っている⁶⁾。その結果、26時間の授業を通じて、繰り返し命令を含めた作品をつくることができ、条件分岐やキー人力の判別処理にも8割を超える児童が取り組むことができていたことを確認している。また、深谷・宮地は、小学生向けにプログラミング授業を行うために短時間でプログラミングを教えることを検討している⁷⁾。Webブラウザ上で動作するプログラミング環境プログラミンを用いて、小学生6年生を対象としたプログラミング授業のための教材を作成して実践し、その有用性を明らかにしている。若菜は、小学生を対象としたプログラミング教育の実践を学校外のスクールに関連した実践例として報告している⁸⁾。西はArduBlockプログラミング環境とマイコンカーを用いたプログラミングの授業をデザインし、走行シミュレータを導入した状態での実践を中学校と学校外のスクールの小学生に対して行っている⁹⁾。

また、文部科学省は、初等中等教育段階におけるプログラミング教育を推進するため、児童生徒の発達段階に応じたプログラミングに関する学習の事例を収集し、教員向けの指導に役立つ参考資料を作成し、プログラミング教育実践ガイドをWEB上に公開している¹⁰⁾。そこでは、小学校1年生から高校3年生まで、各学年を対象としたプログラミング教育について、12例が紹介されている。

1.3.3 中学校技術科におけるプログラミング学習の実践事例

中学校技術科では、現行の学習指導要領下で、内容「D 情報に関する技術」の「プログラムによる計測・制御」において、プログラミングに関する学習が行われている。それらは大きく2つに分けられ、一つは、実際にハードウェアを制御するためにプログラミングを用いるものである。もう一つはPC等の画面上でキャラクター等を動作させるプログラミングに分けら

れる。

ロボットなどのハードウェアを動作させる先行研究として、例えば、室伏・高木は、タブレット端末上でプログラムを作成し、ロボットに送信して制御することができる教材開発を行っている¹¹⁾。教材はタブレット端末が搭載するセンサの計測データを利用し、ロボットのプログラムに反映させるものとしている。また、樋口は、汎用計測・制御基盤を用いた教材を開発し、プログラミングソフトであるドリトルを用いて日本語にてプログラミングを行い、LEDを用いたオリジナルイルミネーションを製作する実践を紹介している¹²⁾。

PC等の画面上でプログラミングを行うものとしては、例えば、村田・加藤は、プログラミングでパズルを操作し、アルゴリズムの理解を目的とした実践を行っている¹³⁾。また、尾崎は、アニメーション制作を通してプログラミングの基礎学習を取り扱っている。題材として文部科学省が提供しているビジュアルプログラミングツールであるプログラミンを用いて、動物や乗り物などのイラストを動かすプログラミング実践を行っている¹⁰⁾。

1.4 問題の所在

先行研究を整理し、プログラミング教育についてまとめたところ、現在、学校現場において、実質的にプログラミングに関する学習内容を展開しているのは、中学校技術・家庭科の技術分野や高等学校情報科であるが、授業時数や学習内容等の問題から、プログラミングに特化した授業展開を効率的に行うには様々な問題がある。また、小学校においてプログラミング教育が導入される予定であるものの、指導者、教材、機器などの様々な障壁があり、難航することが予想される。

しかし、今後の情報化社会を生きる子どもたちは、コンピュータやネットワークを利用した学習活動において効率的かつ効果的に問題を発見・解決・評価・フィードバックすることは、論理的思考力や問題解決能力などを育成するために重要であり、学習の成功・失敗経験を積み重ねることによって、自分自身で工夫・創造する力が身に付くと考えられる。しかし、プログラミングのスキル教育に傾倒すると、変化の著しい情報機器やソフトウェアに対応することはできない。5大機能やプログラミング言語による制御という普遍的な性質を理解するプログラミング教育の展開こそが重要であると考えられる。

既存のビジュアルプログラミングツールにおいても、小学生に対して有効なものいくつか存在する。例えば、Scratchでは視覚的なプログラミングを作成することを目的とし、ブロック化された命令をつなげて、

キャラクターを動作させることができる。それぞれのブロックが色や形で分けられており、直感的な操作が可能である。また、文部科学省が開発したプログラミンでは、あらかじめ操作できるキャラクターを複数設定してあり、子どもの興味・関心を引きやすいものとなっている。Scratchやプログラミンなどの子どもが操作可能なビジュアルプログラミングツールは、各命令構文を意味するブロックにパズルのような凹凸をつけることで、動作の保証が可能となっているところに有意性がある。

しかし、プログラムをつくるという作業において、記述した命令の中からバグを見つけ、プログラムを改善することは、プログラムの構想・設計と同様に重要な学習機会であると考えられる。動作を保障し、成功経験を積み重ねることは、初学者である子どもの興味・関心を持たせやすいという側面もあるが、「はいまわり」を起こす危険性もある。すなわち、自分の作ったプログラムでキャラクターが動くということにのみ面白さを見出し、与えられた環境下での課題達成以上の活動が期待できないことがある。プログラミング教育のように、ただのスキル教育ではなく、コンピュータの仕組みやその制御という普遍的な性質を理解する教育を実践するためには、ビジュアルプログラミングツールを用いて子どもたちに興味・関心を持たせると同時に、その先にある問題解決的、論理的な思考の育成には課題設定が重要であり、そのためにはデバッグを楽しむことができ、試行錯誤する「あそび」の部分を併せ持ったプログラミングツールの開発と実践が必要である。このように、既存のビジュアルプログラミングツールの枠組みを超え、社会で使われているプログラミング言語への円滑な学習移行が行えるツールの開発を行う必要があると考えられる。

子どもたちが自由な発想で、創造的にプログラムを構想・設計した場合に、プログラミング言語を使えばそれを形にすることは可能であるが、子どもたちにコーディングするスキルはない。フローチャートの考え方を使ってビジュアルプログラミングツールを用いれば、容易に作成可能だが必ずしも自由な発想につながるとは限らない。別の手順を試したりコードを見なおしたりする活動や応用性を持った次のプログラムへの発想の妨げになる可能性がある。それらの問題を解決すべく、子どもたちの考えを図形的表現として記述し、それらをつなげて制御するためには、ブロックと並行で言語の記述が可能なビジュアルプログラミングが適していると考えた。具体的には、それぞれのブロックの横にプログラミングできる要件を備える対応を行う。そして、形になったブロックをテキストとして翻訳した後改善する学習活動を通して、仮説を立てなが

らエラーやバグを見つけ、構想・設計したプログラミングに近づけていくことができる。このような学習活動が可能になれば、プログラミング学習は深化し、ビジュアルプログラミングにおける「はいまわり」を防止できることが期待できる。

本研究では、これらの知見を踏まえ、プログラミング初学者向けの複数言語に対応したビジュアルプログラミングツールの開発を行いその活用可能性について検討することとする。

2. 仕様の策定

2.1 ソフトウェアの開発方針

本研究で開発するソフトウェア（以下、「本ソフトウェア」という。）は、学校現場で多く使われている WindowsOS が入った PC 上で起動することを想定し、Visual Studio で開発を行う。画面上に表示するプログラミング言語は、プログラミングの教育用に開発され、児童生徒が触れたことのある英単語が多く用いられた制御文を使っている Visual Basic が望ましいと考えた。

2.2 ソフトウェアの機能

2.2.1 ブロック型プログラミングツール機能

フローチャートに手順を書き込んで、情報処理の手順を考える学習ができる機能である。具体的には、図 1 に示すように用意しておいたフローチャートの枠に手順を入力することでプログラミングできる機能である。

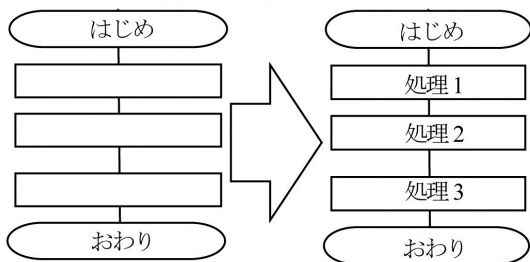


図 1 ブロック型プログラミングツール機能の構想
左の枠組みの中に処理を入力する

2.2.2 テキスト型プログラミングツール機能

簡素化・簡易化した Visual Basic でプログラミングできる機能である。具体的には、図 2 に示すようにコードを穴埋め形式にし、正しくコードされたらプログラムができるようになる機能である。コードが正しくなければ、デバックの必要があることを知らせる機能も搭載している。

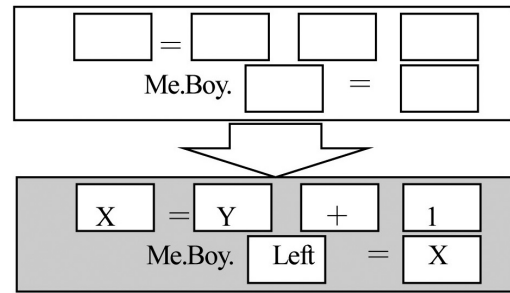


図 2 テキスト型プログラミングツール機能の構想

2.2.3 同時入力機能

図 3 に、いずれかのプログラミングツール機能を使ってプログラミングした場合、もう一方のプログラミングツール機能にも同じ意味をもつプログラムが同時に入力される機能を示す。

動きを表示するブロック型プログラミングツール機能とコードを表示するテキスト型プログラミングツール機能を両立し同時に対応させることで、ソフトウェアの内部の処理が見え、フローチャートによる情報処理の手順の考察ができる。また、プログラミング言語のつながりを体感し、テキスト型によるプログラミングの視覚的なデバックをすることができる。

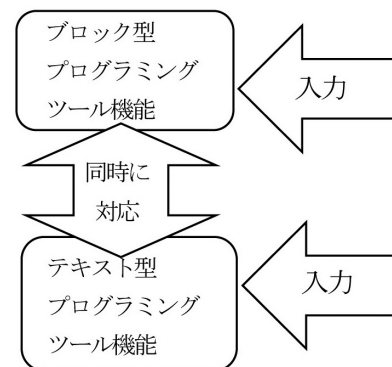
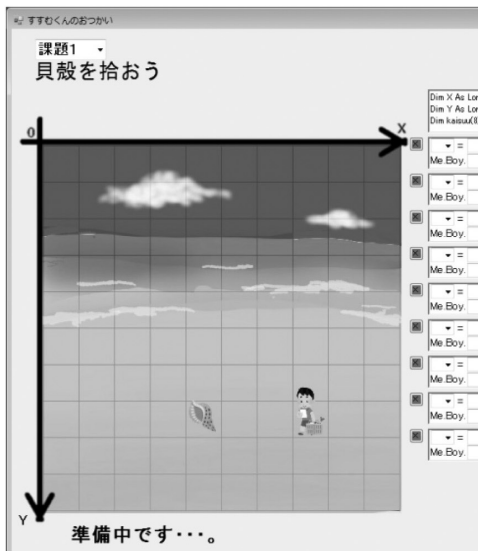


図 3 同時入力機能の構想

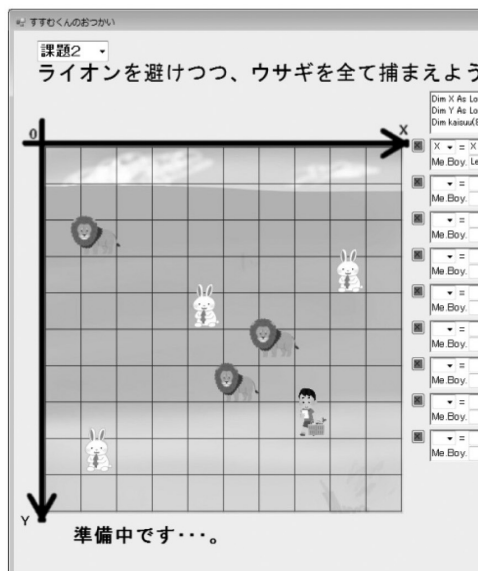
2.2.4 課題機能

図 4 に本ソフトウェアを使った学習をより行いやすく、授業を構成しやすくするために「課題 1」、「課題 2」、「課題 3」の 3 種類の課題を提示する機能を示す。「課題 1」は主に本ソフトウェアの使い方を理解するために端的に終わるものを設定した。「課題 2」と「課題 3」は手順を組み合わせることで達成できるものを設定した。「課題 2」では、障害物を設けたものを設定し、「課題 3」は必ず戻ってくる場所を設けたものを設定し、作成するプログラムに一定の制約を設けている。

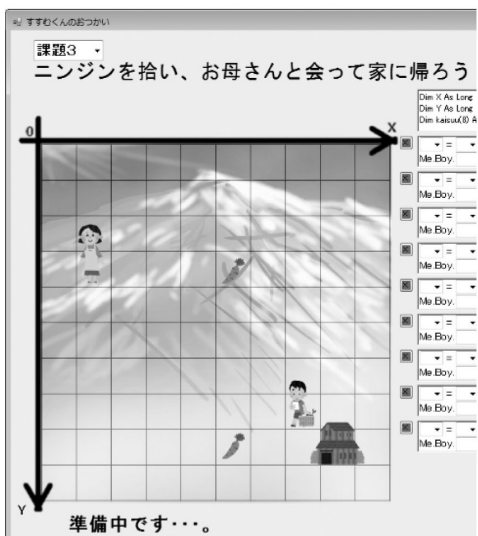
また、使うプログラミングツール機能や取り組む順番は指導者や学習者が決められるように、課題の順番は設定しない。



(a) 課題1



(b) 課題2



(c) 課題3

図4 課題機能

3. ソフトウェアの開発

3.1 各画面の機能

図5に本ソフトウェアの起動時の状態を示す。

- ①ブロック型プログラミングツール部
- ②繰り返す回数を変更することができるコントロールパネル
- ③実行ボタン
- ④課題の達成状況や男子の位置を元に戻すボタン
- ⑤ブロック型およびテキスト型プログラミングツールを転換させるボタン
- ⑥テキスト型プログラミングツール部
- ⑦作成したプログラムを一段分消去するボタン
- ⑧課題の変更を行うテキストボックス

3.2 ブロック型プログラミングツール部

図6に示すようにテキストボックスをクリックすることで、プログラム一覧を開くことができる。その中からプログラムをクリックすることで、プログラムを組むことが出来る。対応する2つのテキストボックス内には、テキスト型のプログラムが同時に表示させている。③実行ボタンをクリックすることで、作成したプログラムを上から順に処理をしていく。プログラムがない場合は、次のプログラムの処理を行う。

また、図7に示すように `kaisu()` の数値は、②繰り返す回数の変更することができるコントロールパネルを操作することで変えることができる。これもテキスト型のプログラムに反映される。

3.3 テキスト型プログラミングツール部

図8に示す⑦~⑧のテキストボックスをクリックすることで、その部分に入るコーディングに必要な変数や構文の一覧を開くことができる。その中からクリックして選択することで、プログラミングができるようになる。正しいコーディングがされていれると、図9に示すようにブロック型プログラミングツール部にそのコードが示す動きが同時に表示される。

また、⑤ブロック型およびテキスト型プログラミングツールを転換させるボタンを「テキスト型のプログラムを動作させる」にチェックした状態では、図10に示すようにコンパイルエラーを起こしているときにテキストボックスの内部が塗りつぶされる。これは、プログラムのデバックが完了したときに元に戻る。また、コンパイルエラーを起こした状態で③実行ボタンをクリックすると、そのプログラムは「プログラムなし」と見なして次のプログラムの処理に移る。

図8の⑧をクリックし数値を変えることで、 `kaisu`

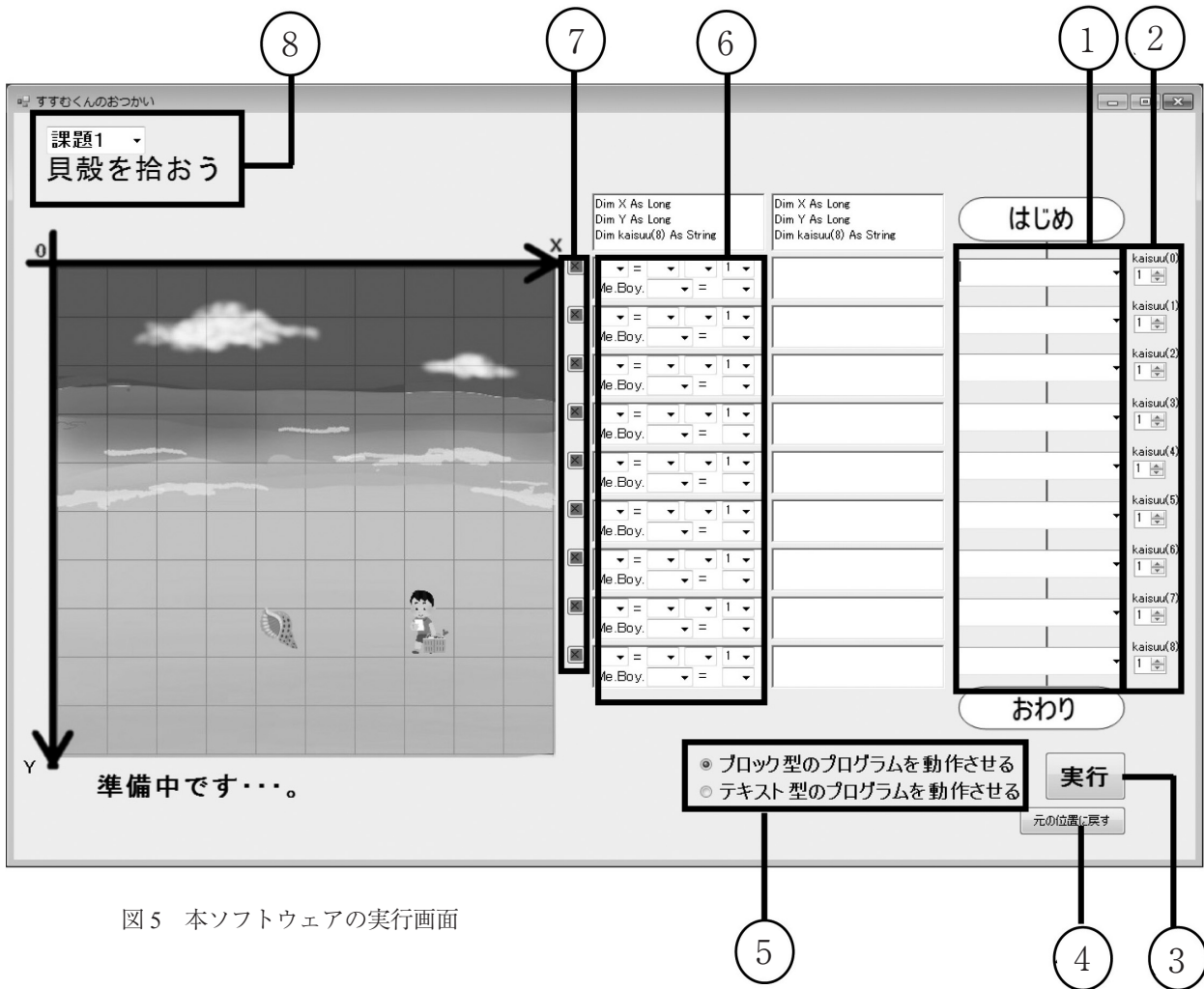


図5 本ソフトウェアの実行画面

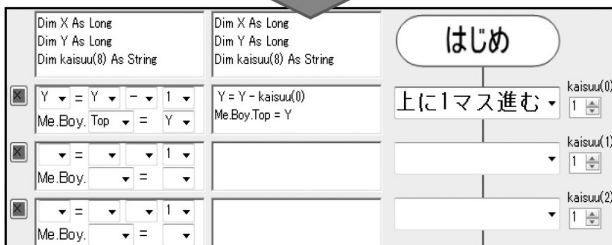
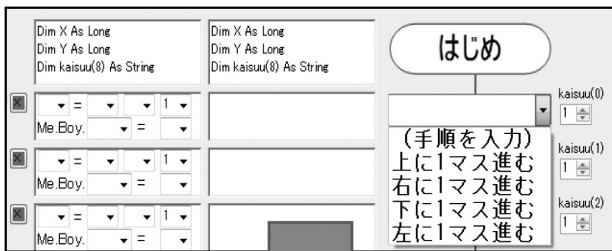


図6 テキストボックスを開いてプログラムを選択した状態

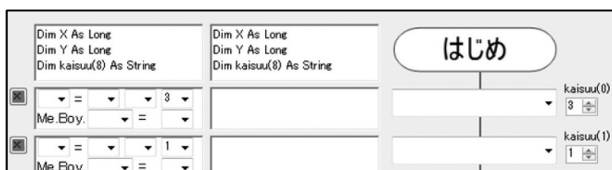


図7 kaisu()の数値を図6から変えた状態

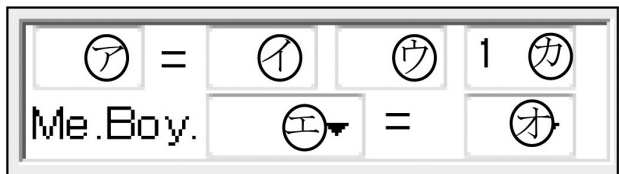


図8 テキスト型プログラミングツール部のテキストボックスの拡大図

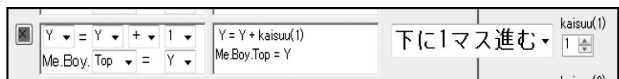


図9 正しくプログラミングされた状態



図10 コンパイルエラーを示している一例

() の数値を変えることができる。こちらもブロック型プログラミングツール部へ同時に反映される。

3.4 課題機能

図5の⑧課題の変更を行うテキストボックスをクリックし、図4に示す課題の一覧のなかからクリックすることで課題を変更することができる。③実行ボタンをクリックし、作成したプログラムを実行したとき、図11に示すように課題を達成した場合は「達成!」、失敗した場合は「失敗!」と表示される。

また、④課題の達成状況や男の子の位置を元に戻すボタンをクリックすることで、再度同じ課題に挑戦することができる。⑦作成したプログラムを一段分消去するボタンをクリックすることで、プログラミングを

はじめからやり直したり、プログラムを消したりことができる。

4. 開発したプログラミングツールの活用方法

4.1 小学生が利用する場合

小学校におけるプログラミング教育での活用では、まず、総合的な学習の時間で、プログラミング学習を行うときに1時限を通して使う方法が考えられる。プログラムとプログラミング言語についてコンピュータが処理すべき一連の手順と手順をコンピュータに伝達するために必要な言葉であることを説明した後に、本ソフトウェアを用いるものとする。

ブロック型プログラミングツールを搭載した本ソフトウェアは、初学者にとってプログラミングに触れやすく、指導者含め簡単に扱うことができる。そのため、簡単にプログラミングができ、プログラミングについて親しみを持たせることが期待できる。また、アクチュエータなどの機材不良の心配が無いため、問題の解決への必要な手順を考えることに集中でき、よりアルゴリズム思考能力を養うことができる。

本ソフトウェアに付属させている課題はプログラムの改善過程において、一つの明確な答えではなく、児童一人ひとりの最適解が求められるように設定している。つまり、児童一人ひとりが、自らの構想を持って課題に取り組むことができる。また、プログラミングの特性の一つである、様々な記述の仕方があることを早い段階かつ持っているレディネスの範囲で体感することができ、手順のブロックの組み合わせを否定されことなく改善する活動ができる。これはプログラミング的思考の育成に加え、学びに向かう力・人間性の育成につながる。

本ソフトウェアでは、流れ図を構成するフローチャート記号を処理のみにして簡素化している。これは、プログラミングツールで培ったレディネスを基に身の回りのコンピュータの活用事例を列举してプログラムを考える活動を想定した場合、フローチャートが複雑化・専門的になりすぎないようにするためである。児童や指導者が手順をフローチャートに変換しやすくなり、多くの活用事例を取り上げやすくなる。これは、より多くの児童が授業に参加しやすくなり、コンピュータやプログラミングの良さに気づきやすくなることにつながる。また、算数や音楽でのプログラミング教育を想定した場合、児童が理解しやすい流れ図を提示することができ、アルゴリズム的思考の育成に結びつけることができる。



(a) 達成した場合



(b) 失敗した場合

図11 課題を達成したか否かを評価する表示

4.2 中学校技術科内容「D 情報に関する技術」のガイダンスで利用する場合

本ソフトウェアは、プログラミングとプログラミング言語を簡易化・簡素化したものであり、プログラミングに対して不安のある生徒にとっても扱いやすいため、触れやすく親しみを持ちやすいと考えられる。また、プログラミング教育を既習の生徒にとっては小学校段階での学習の復習に足る内容となっている。これによりクラス全体の理解度を底上げすることに繋がり、授業を円滑に進めやすくなることが期待される。また、プログラミングに対しての不安を取り除くことができ、プログラムを作成することへの学習意欲の向上や動機付けにつながることを期待される。フローチャート形式で構成されたブロック型プログラミングツールやブロック型で選択したプログラムをプログラミング言語に変換・表示する機能を併設することで、コンピュータの5大機能やこれから学習する内容をアナウンスすることができる。

4.3 中学校技術科の「プログラムと計測・制御」の計測・制御メインの主題材への足がかりとする場合

ブロック型プログラミングツールとテキスト型プログラミングツールが隣接し、相互が同時に対応する本ソフトウェアは、フローチャートやプログラムとプログラミング言語の関係性を説明する上で例示しやすく、理解を促しやすい。

プログラミング教育での活用を目的とした本ソフトウェアは、同じプログラミング的思考の育成を目的とした現在中学校現場で多く取り入れられているビジュアルプログラミングツールや教材との類似性がある。機材不良による学習への支障が起こりにくく、クラス全体の短期間でのプログラミング的思考の底上げを図ることができる。また、簡素化したプログラミング言語を取り扱えるため、コーディングなどの発展的内容へ対応するためのレディネス形成ができる。

プログラミングの基本的知識の定着、クラス全体のプログラミング的思考の底上げを効率よく行えるこの活用方法は、主題材を高いレベルに設定している場合に特に有効である。

4.4 中学校技術科の「プログラムと計測・制御」の授業の中で利用する場合

ブロック型プログラミングツールとして、フローチャートに手順を書き込み、情報処理の手順を考える学習ができるよう設計された本ソフトウェアは、問題解決のためにフローチャートを活用して順序だてて考えようとする力を養うとともにプログラミング的思考

を培うことができる。本ソフトウェアはブロック型でプログラムした手順からプログラミング言語での記述を同時に入力されるように設計しているため、内部で行われている処理を意識し、コンピュータの働きの本質の理解につなげることができる。

小学校段階でブロック型プログラミングソフトやツールを用いたプログラミング学習を多く経験している生徒については、テキスト型プログラミングツールを使つてのプログラムを考える活動を発展的課題として行わせることができる。本ソフトウェアはプログラミング言語のスキルがなくてもコーディングができるように、穴埋め形式のコーディングスタイルで設計されている。また、同時入力機能や図9や図10のようにエラーを視覚的に知らせる機能を入れており、小学校段階での学習内容を活用することでコーディング体験をすることができる。

4.5 中学校技術科の「プログラムと計測・制御」のプログラム作成の補足・まとめに利用する場合

現在中学校現場で多く取り入れられているブロック型プログラミングツールを活用した学習では、プログラミング言語によって記述されたプログラムをコンピュータが情報処理やハードの制御を行っていることや5大機能が働いていることが見えづらい点があげられる。また、厳密に言えばブロック型プログラミングツールはプログラミング言語ではなく、プログラミング的思考の育成のみに傾いた指導では、これらがプログラミング言語であると誤認してしまう可能性がある。高等学校情報科では、プログラミング言語を扱う学習が多いため、プログラミング言語についての補足的説明をしなければ、生徒の学習に支障が起こることが予想される。本ソフトウェアは、テキスト型プログラミングツールを併設しており、プログラミング言語の一つである Visual Basic で記述されたコードを見せることができる。ブロック型プログラミングツールの内部で記述・処理されていたコードを示すことで、5大機能と内部で行われている処理に注目することができ、フローチャートやプログラムとプログラミング言語の関係について正しく理解することができる。

また、本ソフトウェアのテキスト型プログラミングツールは簡易化・簡素化して設計しており、ブロック型プログラミングツールと同じ感覚で扱うことができる。また、複数言語に対応したプログラミングツールとして相互が同時に対応するため、プログラミング言語に触れやすく、高等学校情報科で行うプログラミング学習のレディネスとなることを期待される。

5. まとめと今後の課題

本研究では、現在のプログラミング教育の問題点を考察し、それに基づいてマルチビジュアルプログラミングツールの開発を行った。本研究で開発されたソフトウェアは、プログラミング的思考の育成を促し、小中学生それぞれを対象としたプログラミングや言語の指導ができる可能性がある。本研究で開発したソフトウェアは、プログラミング教育にプログラミングの本質を組み込み、なおかつ初学者がプログラミングに取り組めること、プログラミングへの造詣がない指導者がプログラミング教育を行えるよう支援できることが期待される。

しかし、本研究では、教育実践が未着手であり、使用実験・質問紙調査等を行った上での教育効果の立証をする必要がある。本ソフトウェアでのより汎用的な教育実践を行うために、条件分岐、条件繰り返しの機能追加の検討および改良を行う必要がある。また、学習指導要領（平成 29 年 3 月公示）では、動的コンテンツに関するプログラミングやネットワークやデータを活用して処理するプログラミングとより幅広い内容を取り扱うことが求められている。技術科教員としてのより高い指導力や果たす役割の増大が求められ、教材・教具の開発も必要となることが予想される。その一助とするべく、授業者のニーズに応えた要素の追加や生徒のレディネスに応じた可能な限りの学習サポートなど、本ソフトウェアの改善をしていく必要がある。

参考文献

- 1) 高度情報通信ネットワーク社会推進戦略本部（IT 総合戦略本部）：第 8 回新戦略推進専門調査会，人材育成分科会議事次第：
http://www.kantei.go.jp/jp/singi/it2/senmon_bunka/jinzai/dai8/gijisidai.html (2015)
- 2) 文部科学省：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，小学校段階におけるプログラミング教育の在り方について（議論の取りまとめ）
http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.htm (2016)
- 3) 西田知博・原田章・中村亮太・宮本友介・松浦敏雄：初学者用プログラミング学習環境 PEN の実装と評価，情報処理学会論文誌，48 (8)，pp.2736-2747 (2007)
- 4) 例えば，スクラッチ：<https://scratch.mit.edu/> など
- 5) 江本鶴子・竹内章：プログラミング初心者にはトレースを指導するデバッグ支援システムの開発と評価，日本教育工学会論文誌，32 (4)，pp.369-381 (2009)
- 6) 森秀樹・杉澤学・張海・前迫孝憲：Scratch を用いた小学校プログラミング授業の実践：小学生を対象としたプログラミング教育の再考，日本教育工学会論文誌，34 (4)，pp.387-394 (2011)
- 7) 深谷和義・宮地晶子：小学生向けプログラミング授業のための「プログラミン」利用の検討，日本教育工学会論文誌，36 (Suppl.)，pp.9-12 (2012)
- 8) 若菜啓孝：小学生を対象としたプログラミング教育について，長崎大学大学教育イノベーションセンター紀要，(7)，pp.35-40 (2016)
- 9) 西正明：ArduBlock プログラミング環境とマイコンカー走行シミュレータの効果，日本産業技術教育学会第 22 回技術教育分科会講演要旨集，pp.35-36 (2016)
- 10) 文部科学省：プログラミング教育実践ガイド
http://jouhouka.mext.go.jp/school/programming_zirei/ (2015)
- 11) 室伏春樹・高木薫：タブレット端末を利用したプログラムによる計測・制御教材の開発，日本産業技術教育学会誌，57 (3)，pp.179-186 (2015)
- 12) 樋口大輔：汎用計測・制御基板を用いたプログラムによる計測・制御の学習の提案，日本産業技術教育学会誌，58 (2)，pp.119-124 (2016)
- 13) 村田育也・加藤慎司：中学生に対する 5 パズルを用いたプログラミング教育方法の開発，科学教育研究，40 (3)，pp.256-270 (2016)