

CAS ガイドラインに基づく小学校プログラミング教材の提案： mBot ロボットとアルゴリズム学習用ツールキットの活用

岩崎誠^{†1} 掛下哲郎^{†1}

概要：2020年から小学校にて新学習指導要領が実施され、プログラミングを通じて、論理的思考力を育成することが決まっている。しかし、文部科学省が本年3月に公表した「小学校プログラミング教育の手引き（第1版）」等を見ても、具体的な教育内容や教育レベルは明確とは言えない。我々は、論理的思考力を育成するために、mBot ロボットを用いたアルゴリズム学習用ツールキットを開発している。本ツールキットは、mBot の様々な機能をカバーする22個のプログラム群から構成されており、仕様、アルゴリズム、scratch プログラム及びハードウェアを関連づけて mBot プログラミングを学習できる。本論文では、イングランドの Computing At School (CAS) が策定した Computing Progression Pathways (CAS ガイドライン) に基づいて、本ツールキットを活用したプログラミング教材を提案する。Computing Progression Pathways では、6項目の学習項目と8つのレベルを用いて初中等段階における情報教育の全体像を定義している。本論文では、プログラミング教育との関連が深いアルゴリズム、プログラミングと開発、ハードウェアと制御の3項目について、小学校で標準とされるレベル1~5に対応するプログラミング教材を設計した。本教材では、使用する言語機能や、生徒に対する指示内容を適切に選ぶことでレベルに応じた教育を行える。

キーワード：プログラミング教育、初中等教育、論理的思考、mBot ロボット、CAS ガイドライン

A Proposal of Elementary Level Programming Education Material Based on CAS Computing Progression Pathways: Utilizing mBot Robot and a Toolkit to Learn Algorithmic Thinking

MAKOTO IWASAKI^{†1} TETSURO KAKESHITA^{†1}

Abstract: New curriculum will start at Japanese elementary schools in 2020. The curriculum contains education of algorithmic thinking through programming education. However education content and expected level are still unclear, although Japanese ministry of education published a leaflet of programming education at elementary schools (first edition). We have developed a toolkit to learn algorithmic thinking using mBot robot. The toolkit contains 22 programs that cover all functions of mBot hardware components. A toolkit user can learn computer programming by using correspondence among specification, algorithm, Scratch code and mBot hardware component. We propose programming education material utilizing the toolkit in this paper. The material is based on Computing Progression Pathways instituted by Computing At School (CAS Guidelines) in England. CAS guidelines define overview of information education at elementary and secondary school by 6 categories and 8 levels. We design education material about algorithm, programming & development, hardware & processing having strong relationship with programming education. The material corresponds to the education level 1 to 5 that is standard at elementary school. Education level of our material is controlled by selecting appropriate language feature and instruction to the students.

Keywords: Programming Education, Elementary and Secondary Education, Algorithmic Thinking, mBot Robot, CAS Guidelines

1. はじめに

2020年から小学校にて新学習指導要領が実施され、情報活用能力を言語能力と同様に「学習の基盤となる資質・能力」と位置づけている[1]。小学校においては、プログラミングを通じて、論理的思考力を育成することが決まっている。文部科学省によると、プログラミング教育で育成すべき資質・能力は、知識・技能、思考力・判断力・表現力等、学びに向かう力・人間性等の3つの柱で構成されている。

しかし、プログラミング教育を普通教科として単独では実施しておらず、総合的な学習の時間、理科、算数などの既存教科内での学習を行うことに留まっている。また、「小

学校プログラミング教育の手引き（第1版）」[2]を参照しても、上記の教科におけるプログラミング教育の内容について例示されているが、教育内容、教育レベル、学習時間等についての具体的な記述は見受けられず、学習教材や、授業計画を作成する際の根拠にするのは難しい。

これに対してイングランドでは、小中学生が学ぶべき情報教育の内容を学習項目ごとに具体的に示した Computing Progression Pathways が Computing At School (CAS) によって策定され、小中学校に広く普及している[3]。

我々は、教育用ロボット mBot (図1) を活用したプログラミング教育支援ツールキットを開発している[4]。本ツールキットは mBot の様々な機能をカバーする22個のプログラム群から構成されており、仕様、アルゴリズムおよび scratch プログラムを関連づけて mBot プログラミングを学

^{†1} 佐賀大学
Saga University

習できる。本ツールキットを活用することで、教員や児童に具体的なプログラム例を示すことができ、algorithmic thinking の立場から系統的なプログラミング教育を実施することが期待される。また、アルゴリズム中に含まれる基本操作を、操作対象のデバイス（モーター、各種センサー等）毎に分類することで、様々なアプリケーションを企画する際のヒントとしても活用できる。



図1：mBot ロボット

本研究では、我々が開発しているプログラミング教育支援ツールキットを用いて、CAS の「Computing Progression Pathways」(以下、CAS ガイドラインと略記する)に基づくプログラミング教材を提案する。

以下、2 節では、日本における小学校プログラミング教育を支援する取り組みとの関係を示す。3 節では、日本における検討状況と、プログラミング教育が進んでいるイングランドにおける小学校段階のプログラミング教育の現状を紹介する。4 節では、ツールキットで利用する mBot について紹介する。5 節では、mBot を利用したプログラミングツールキットおよびその特徴について紹介する。6 節では、Computing Progression Pathways に基づいて、ツールキットを活用したプログラミング教材を設計する。

2. 関連研究

文部科学省が公表している小学校プログラミング教育の指針を強化するための取り組みはいくつか行われている。

小池は、小学校第3学年の総合的な学習の時間におけるプログラミング教育のカリキュラムを開発した[5]。このカリキュラムは、第3学年の総合的な学習の時間に実施するものであり、ゲーミフィケーションの考え方を踏まえて、全7次40時間の構成になっている。2017年4月～10月に実践を通じて適用可能性が検証されている。

株式会社ベネッセコーポレーションと NPO 法人 CANVAS は、「プログラミングで育成する資質・能力の評価規準(試行版)」を公開して、プログラミング教育の目標を示している[6]。プログラミングに関する「知識・技能」については、逐次処理、繰り返し処理、条件分岐、センサーを使ったプログラミングが取り上げられているが、CAS ガイドラインに盛り込まれているルーチンやデータ構造の概念には触れられていない。

早稲田大学グローバルソフトウェアエンジニアリング研究所と株式会社フジテレビ KIDS 等は共同で、小学校プロ

グラミング教育において活用可能なルーブリック形式の評価基準(Rubric ProEEs)を提案している[7]。この評価基準の項目には、プログラミング概念の理解、プログラムの設計、プログラム作成、プログラムの読解・編集・評価等が含まれており、ソフトウェア工学的な立場がより強く反映されている。これに対して、本研究では、CAS ガイドラインを参照することで、センサー等のハードウェア制御も含む構成になっている。

3. 日本とイングランドにおける小学校段階のプログラミング教育

日本では、2020年から小学校にて新学習指導要領が実施され、情報活用能力を言語能力と同様に「学習の基盤となる資質・能力」と位置づけている。特に小学校においては、プログラミングを通じて、論理的思考力を育成することが決まっている。

しかし、プログラミング教育を普通教育として単独では実施しておらず、総合的な学習の時間、理科、算数などの教科での、プログラミングを用いた学習に留まっている。また、教育内容、教育レベル、評価についての具体的な記述は見受けられない。文部科学省が本年3月に公表した「小学校プログラミング教育の手引き(第1版)」についても同様である[2]。

また、2016年6月の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の有識者会議の議論とりまとめでは、プログラミング教育を実施するうえで、ICT環境の整備、教材の開発や指導事例集の整備、教員研修等の在り方、指導体制の充実や社会との連携・協働などが課題として挙げられている[8]。

2014年に文部科学省委託事業として、諸外国におけるプログラミング教育に関する調査研究が行われた[9]。これによると、イングランドでは、2013年9月のナショナルカリキュラムの改定において、教科「ICT」は「Computing」に代わり、従来のコンピュータの操作スキルやアプリケーションの使い方の学習に加えて、アルゴリズムの理解やプログラミング言語の学習を取り入れるなどコンピュータサイエンスの内容を充実させている。

「Computing」は独立教科として設置され、Key Stage 1-4(5-16歳)で必修である。具体的な指導内容や方法、指導時数については地区、学校、指導者に委ねられている。ヒアリングによると、プライマリースクールでは毎週1時間(毎年約30時間)が一般的である。「Computing」はCS(Computer Science)、IT(Information Technology)、DL(Digital Literacy)の3分野で構成され、プログラミング教育は主にCSに含まれる。

また、教員向けの教材の提供や研修を実施している Computing At School(CAS)によって小中学生が学ぶべき情報教育の内容を学習項目ごとに段階的に示した

Computing Progression Pathways (以後、CAS ガイドラインと表記する) が策定され、初中等学校に普及している。CAS ガイドラインは、学習時の目標設定や、次のステップで習得すべき内容の確認のほか、学習者の習熟度を評価するためにも活用されている。

CAS ガイドラインの学習項目は、アルゴリズム、プログラミング・開発、データ・データ表現、ハードウェア・処理、コミュニケーション・ネットワーク、情報技術の6項目に分類される。また、それぞれの具体的な到達目標は AB (Abstraction): 抽象化, DE (Decomposition): 分割, AL (Algorithmic Thinking): アルゴリズム的思考, EV (Evaluation): 評価, GE (Generalization): 一般化の5つの概念に分類される。到達レベルは8段階で定義されており、そのうち、小学校ではレベル1~5が標準とされている。レベル5における到達目標を以下に示す。

- アルゴリズム
 - 反復処理を理解している。(AL)
 - 問題を解くアルゴリズムは複数存在。(AL)(GE)
 - 構造化表記による解決法の記述 (AL)(AB)
 - 類似点や相違点を利用した問題解決 (GE)
- プログラミングと開発
 - プログラミング概念の理解 (AB)
 - 高水準言語プログラミングの経験 (AB)(AL)
 - 様々な種類の演算子や式の活用 (AL)
 - 適切なデータ型の選択 (AL)(AB)
- データとデータ表現
 - ビットパターンによる数値と画像の表現 (AB)
 - 二進形式でのデータ表現とデータ転送 (AB)
 - 二進データとファイルサイズの関係 (AB)
 - 実数と Boolean 型のデータ型の定義 (AB)
 - 問い合わせ言語を利用したデータ検索 (AB)
- ハードウェア・処理
 - コンピュータを構成する主な部品 (AB)
 - 命令フェッチ実行サイクルの概念 (AB)(AL)
 - OS とアプリケーションソフトウェア (AB)
- コミュニケーション・ネットワーク
 - 検索結果のランク付け (AL)
 - HTML と CSS の利用 (AL)(AB)
 - IP アドレスとパケット交換 (AL)(AB)
- 情報技術
 - インターネットサービス等の妥当性評価 (EV)
 - 情報技術の適用における倫理問題
 - 解決策の品質評価と改善 (EV)

本資料では、CAS ガイドラインに基づき、プログラミング教育と関連が深いアルゴリズム、プログラミングと開発、ハードウェア・処理の3項目について、各レベルにおける mBot ツールキットを用いたプログラミング教材を提案する。

4. mBot ロボットの概要

4.1 ハードウェア構成

図2に示すように、mBot のメインボードである mCore は、初期状態では以下に示す8種類のデバイスで構成される。これにより、CAS ガイドラインの「ハードウェア・処理」をカバーできる。

1. RJ25: 追加センサーを接続
2. RGB LED: RGB コードで指定された光を出力
3. ブザー: 指定された周波数でブザーを鳴らす。
4. IR 受信機: リモートコントローラから情報を受信
5. 光センサー: 周囲の明度を検出
6. IR 送信機: 数字と文字を受信機に送信
7. ボタン: ボタンから値を取得
8. モーターポート: mBot のモーター(車輪等)を制御

上記の電子機器に加えて、障害物を検出するための超音波センサー、床に書かれた線を検出するライントレースセンサー、無線通信のための Bluetooth などを利用できる。

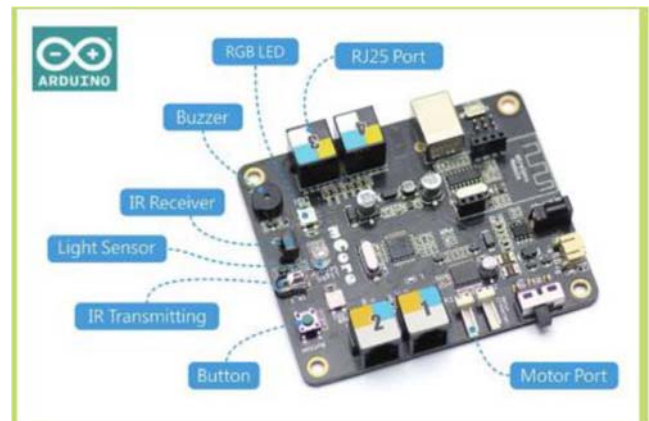


図2: mCore メインボード

4.2 プログラミング環境

mBlock は、mBot ロボットのプログラミング環境である。mBlock のユーザインタフェースを図3に示す。mBlock は Scratch コードを Arduino コードに自動変換する。開発者は、USB ケーブル、Bluetooth、または携帯無線(2.4G)を用いて PC、スマートフォンと接続することで、プログラムを用いて mBot を制御できる。また、開発したプログラムを mBot にアップロードすることで、mBot 単体でプログラムを実行できる。

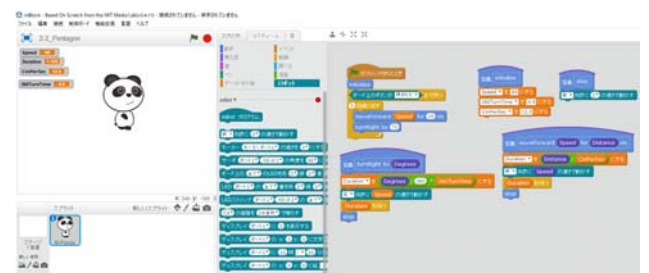


図3: mBlock のユーザインタフェース

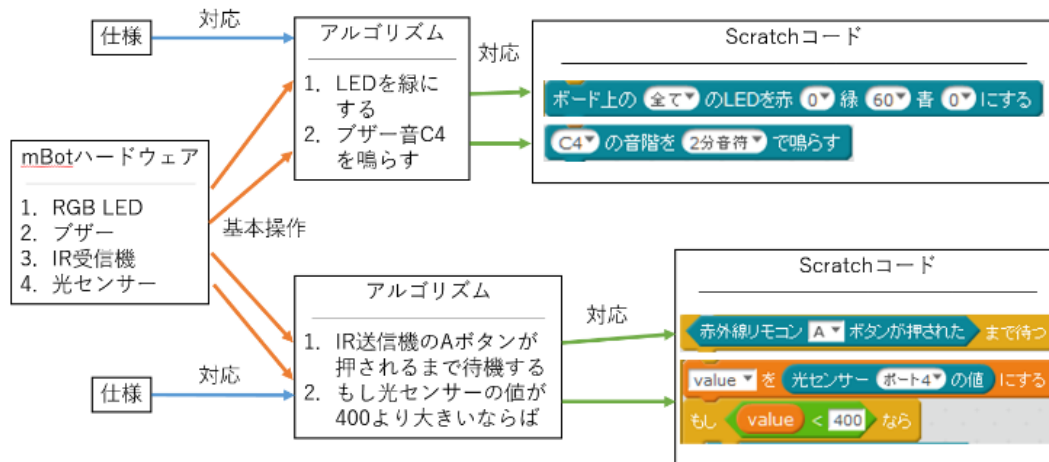


図4：ツールキットの構成

5. プログラミングツールキットの概要

5.1 ツールキットの構成

我々が提案しているツールキット[4]は仕様、mBot ハードウェア、アルゴリズム、Scratch コードの4つで構成される(図4)。図に示すように、仕様とアルゴリズム、ハードウェアとアルゴリズム、アルゴリズムと Scratch コードは相互に対応付けられている。

これらの要素によってツールキットを構成することにより、以下の利点が得られる。

- 仕様とアルゴリズムを対応付けることにより、目的とそれを実現するための(自然言語による)手段の記述を自然に対応付けることができる。
- アルゴリズムの各ステップと Scratch コードを対応付けることにより、自然言語による手段の記述と、プログラミング言語による記述の対応を具体的に表現できる。
- アルゴリズム中の各ステップは、基本操作に対応しているが、それらの基本操作を操作対象の mBot のデバイス毎にまとめることで、デバイスが提供している機能を自然に表現できる。その例を表1に示す。

表1：モーターと基本操作の対応の例

デバイス	基本操作
モーター	指定した速度で直進する
	指定した速度で後退する
	停止する
	右方向に指定した角度に曲がる
	右方向に指定した速度で曲がる
	右車輪を指定した速度で回転する

本ツールキットは、以下に示す 22 個のプログラムを含む。これらのプログラムは、授業を行う際の例としても、演習を行う時のテンプレートとしても活用することを意図

している。

- mBlock の制御構造等を含む基本コード (1 個)
- 図形の描画 (4 個)
- 入力データを用いたロボットの制御 (6 個)
- 6 本足パーツを活用した歩行ロボット (3 個)
- 各種センサーを活用する基本コード (8 個)

我々は mBot に対応して本ツールキットを開発しているが、mBot 以外のハードウェアに対応してツールキットを修正することも可能である。

5.2 ツールキットの具体例

5.2.1 「長方形の描画」の仕様

モーターを用いて mbot に 1 辺の長さが 10 cm の長方形を描画させる。なお、本プログラムは、mBot ロボットと mBlock ソフトウェアを接続した状態で動作させる。

5.2.2 アルゴリズム

- 1 緑のフラグがクリックされるまで待機する。
- 2 変数を初期化する。
- 3 mBot 上のボタンが押されるまで待機する。
- 4 以下の処理を 4 回繰り返す。
 - 4.1 速度 60, 距離 10 cm で前進する
 - 4.2 右方向に直角に曲がる

5.2.3 Scratch コード

アルゴリズムの各ステップは、Scratch コードに対応する。プログラム全体を図5に示す。青色の「定義」要素はサブルーチンの定義に利用される。' initialize " stop " turnRight ' ' moveForward ' は、モーターのパラメータ値の計算や変数の初期化をすることで、mBot のモーターを制御するサブルーチンである。



図 5 : Scratch コード

6. CAS ガイドラインに基づくプログラミング教材の設計方針

6.1 ツールキットで対応できる学習内容, レベルの検討

本論文では, CAS ガイドラインで定義されている 6 つの学習項目から, プログラミング教育との関連が深いアルゴリズム・プログラミングと開発, ハードウェアと制御の 3 項目を選んで 小学校で標準とされるレベル 1~5 に対応するプログラミング教材を開発する. その際に, 5 節で示したツールキットを活用する. 各レベルの難易度は, 条件分岐や反復, サブルーチンなどのプログラミングに関する学習内容と, 与えられたプログラムをそのまま実行する, アルゴリズムからプログラムを設計するなどの課題内容を変化させることで区別している. 表 2 に各学習項目, 各レベルの学習内容の違いを, 表 3 には生徒に対する指示内容の違いを示す.

学習内容の違いのレベル 1 では, レベル 2 以降で学習する項目を除いた, 単純なアルゴリズム, プログラムを用いて学習する.

レベル 2 では, ループや条件分岐について学習するが, それぞれの処理は組み合わせずに使用させる. 入出力機器

はセンサーを除いたディスプレイやリモコン, ボタンなどについて学習する.

表 2 : 各学習項目, 各レベルでの学習内容の違い

レベル	アルゴリズム	プログラミングと開発	ハードウェア・処理
1	単純なアルゴリズム	単純なプログラム	プログラムの必要性
2	ループ, 条件分岐	算術演算子, 条件分岐, ループ ループ, 条件分岐の片方のみを使用	入出力機器
3	入力に応じた出力	変数, アルゴリズムの実装 ループ, 条件分岐を組み合わせで使用 (入れ子)	センサーによるデータ収集
4	サブルーチン	サブルーチン	OS, ネットワーク
5	パターン認識	演算子, 式の活用	コンピュータアーキテクチャ, フェッチ実行サイクル

表 3 : 各レベルでの指示内容の違い

レベル	アルゴリズム	プログラミングと開発	ハードウェア・処理
1	具体的な指示を元に修正させる		
2	<ul style="list-style-type: none"> エラー修正 実行結果の予測 フローチャートでの表現 仕様の変更を指示し, 与えたアルゴリズム, プログラムを修正させる 		
3	不完全なアルゴリズム, プログラムを復元させる		
4	与えた仕様からアルゴリズムを設計させる	与えたアルゴリズムに基づいてプログラムを実装させる	
5	<ul style="list-style-type: none"> プログラムの企画 「2 台のロボットによるリレー」を設計する 		

レベル 3 では, ループと条件分岐を組み合わせ用い, センサーを利用した入力に応じて mBot を制御する.

レベル 4 では, サブルーチンについて学習し, レベル 5 ではサブルーチンの引数を変化させながら複数呼び出すような複雑なプログラムについて学習する.

課題内容の違いのレベル1では、具体的な指示を元にアルゴリズムやプログラムを修正させ、結果の変化を観察する。

レベル2では、段階的指標で示されたエラー修正や実行結果の予測、フローチャートでのアルゴリズムの表現を行う。また、仕様の変更だけ指示し、具体的な修正内容は児童に考えさせる。

レベル3では、仕様からアルゴリズムを、アルゴリズムからプログラムを設計させる。全体を設計させるのではなく、穴抜きしたアルゴリズム、プログラムを与え、仕様やアルゴリズムと対応付けながら復元させる。

レベル4では、穴抜きを復元させたレベル3の発展として、仕様からアルゴリズム、アルゴリズムからプログラムを設計させる。

レベル5では、仕様からアルゴリズム、プログラムを全て設計させる。「2台のロボットによるリレー」は、ツールキットのプログラム「入力データを用いたロボットの制御」の中の1つである。このプログラムはセンサーやループ、条件分岐などの要素を応用したプログラムなので、レベル5での学習に適している。また、新たなプログラムを企画・開発することで学習効果を高めることができる。

6.2 ツールキットでは対応できない項目の学習方法

段階的指標の学習項目の中には、mBotを活用したツールキットの仕様、アルゴリズム、プログラムを用いた学習では対応できない項目がある。それらの項目についての学習方法について述べる。

「アルゴリズム」のレベル1の「コンピュータを動作させるには正確な指示が必要なことを理解している。(AL)」については、あいまいな表現を含むアルゴリズム(例: Cookpadのレシピ)を意図的に示して、その解釈が複数通りあることを理解させる。

レベル4の「人間やコンピュータが得意とする仕事を認識している」については、人間やコンピュータが得意、不得意なことをグループワークで考えさせ、発表することで学習する。

レベル4,5の「同じ問題に対して、複数の解決策またはアルゴリズムがあることを理解している」については、迷路探索やソートアルゴリズム、多項式の計算など、複数のアルゴリズムがある問題を具体的に示すことによって学習する。

「ハードウェア・処理」のOS, モバイルネットワーク, コンピュータアーキテクチャ, フェッチ実行サイクルについては、他の課題を用いて学習する。

6.3 ツールキットを活用した教材の設計方針

本節では、CASガイドラインで示されている到達目標に対応する教材の設計方針を示す。紙数の制限があるため、ア

ルゴリズム、プログラミングと開発、ハードウェア・処理の3領域とレベル1~5の全ての組み合わせに関する説明は省略するが、それぞれの領域とレベルが少なくとも1つは出現するように配慮している。

6.3.1 アルゴリズム(レベル1)

アルゴリズム(レベル1)の到達目標はCASガイドラインにより以下のように定義されている。

1. アルゴリズムが何かを理解する。分岐を含まない単純なアルゴリズムを記述できる。(AL)
2. コンピュータを動作させるには正確な指示が必要なことを理解している。(AL)
3. エラーを避けるために、注意深く、かつ正確に行動できることを示す。(AL)

上述の到達目標のうち1については、「図形の描画」などの分岐を含まないアルゴリズムを用いて学習することで対応できる。低学年を対象としているので、ツールキットに含まれるアルゴリズム(分岐を含まない単純なもの)を提示し、具体的な指示に基づいてそれを修正させるような課題を出すのが適切だと考えられる。

到達目標2については、あいまいな表現を含むアルゴリズムを意図的に示して、その解釈が複数通りあることを理解させる方法が考えられる。ただし、mBotツールキットはそのようなアルゴリズムを含んではいないので、新たに作成する必要がある。例えばCookpadのレシピは料理をする人への指示なのでアルゴリズムの一種だが、曖昧な表現を多く含む(例: 適量, 味を整える)。人間は常識を元に判断できるが、コンピュータには通用しない。

到達目標3については、作成する際に曖昧な表現を避け、詳細に記述させることで対応できる。具体的には、ツールキットで実践しているように、アルゴリズムとプログラムを対応付けることで曖昧さを避けることができる。また、正しいアルゴリズム(それに基づくプログラム)を一部変更させ、プログラムが意図しない動作をするのを観察させる方法が考えられる。

6.3.2 プログラミングと開発(レベル2)

プログラム(レベル2)の到達目標はCASガイドラインにより以下のように定義されている。

1. プログラム内で算術演算子, 条件分岐およびループを活用できる。(AL)
2. 論理的に考えてプログラムの動作を予測できる。(AL)
3. プログラム内で、簡単な意味のエラーを検出し修正できる。つまり、デバッグできる。(AL)

到達目標1については、「図形の描画」等のツールキットに含まれるプログラムを用いて学習できる。ただし、プログラムを選定する際には、変数, サブルーチン, センサーからの入力を使わず、算術演算子, 条件分岐, ループお

よび mBot のボタンからの入力のみを含むものを用いる。条件分岐の際には mBot のボタンから入力した値を利用する。また、ループ内での条件分岐は使用しない。選定したプログラムを生徒に示し、それを入力・実行させた後、仕様の分岐の条件、ループの回数、ループする処理などを変更し、プログラムの修正は児童に考えさせる。

到達目標 2 については、選定したプログラムを生徒に示し、実行せずに mBot の動作を予測させることで学習できる。予測の正しさは、そのプログラムを実際に実行して見せることで確認できる。

到達目標 3 については、バグ（意味的エラー）を含むプログラムを用意し、アルゴリズム通りに動作するようにエラーを修正させることで対応できる。バグは教師側で意図的に混入させるが、レベル 2 では、ループと条件分岐の学習が主目的なので、その部分にバグを混入させるのが良い。

6.3.3 ハードウェア・処理（レベル 3）

ハードウェア（レベル 3）の到達目標は CAS ガイドラインにより以下のように定義されている。

1. コンピュータは、センサーやアプリケーションソフトウェアを含む様々な入力デバイスからデータを収集していることを知っている。(AB)
2. ハードウェアとアプリケーションソフトウェアの違い、およびコンピュータシステム内でのそれぞれの役割を理解している。(AB)

到達目標 1 については、「入力データを用いたロボットの制御」、「各種センサーの活用」などのプログラムを用いて学習できる。ただし、プログラムの選定に当たっては、変数、ループ内での条件分岐、およびセンサーからの入力を含んで良いが、サブルーチンは含まないものを選ぶ。

到達目標 2 については、これまでに述べたツールキットを用いたアルゴリズム、プログラミングの学習で理解できる。

6.3.4 アルゴリズム（レベル 4）

アルゴリズム（レベル 4）の到達目標は CAS ガイドラインにより以下のように定義されている。

1. 人間やコンピュータが得意とする仕事を認識している。(EV)
2. 問題を分割することで解決策を設計し、個々の部分問題を解決できる。(DE)(AL)(AB)
3. 同じ問題に対して複数の解決策があることを理解している。(AL)(AB)

到達目標 1 については、ツールキットのアルゴリズム、プログラムでは学習できない。人間とコンピュータがそれぞれ得意、不得意とすることについて、コンピュータが活用されている場面について、学習することで対応できる。具体的には、グループワークなどで考えさせた後に、各意

見をまとめ、補足する。または、コンピュータを活用した工場の見学等も考えられる。

到達目標 2 については、「図形の描画」や「入力データを用いたロボットの制御」などの、サブルーチンを用いるアルゴリズムを設計させることで対応できる。具体的には、既存の「図形の描画」のアルゴリズムを元に、任意の多角形を描画するアルゴリズムを設計させることが考えられる。または、サブルーチン呼び出すメイン関数の変更（サブルーチンの引数、呼び出す回数）、サブルーチンの変更（変数やアルゴリズム）なども考えられる。

到達目標 3 については、mBot を用いず、迷路探索やソートアルゴリズム、多項式の計算など、複数のアルゴリズムがある問題を具体的に示すことによって教育する方法が考えられる。mBot ツールキットを用いる必要は必ずしもない。

6.3.5 プログラミングと開発（レベル 5）

プログラム（レベル 5）の到達目標は CAS ガイドラインにより以下のように定義されている。

1. プログラミングは、アルゴリズム的な解決策とコンピュータの間を埋めることを理解している。(AB)
2. 標準的なライブラリを利用する高水準言語について実践的なプログラミング経験がある。(AB)(AL)
3. 様々な種類の演算子や式を活用できる。例えば、文の実行順を制御するために Boolean 型の式を活用できる。(AL)
4. 適切なデータ型を選択できる。(AL)(AB)

到達目標 1 については、これまでのアルゴリズムとプログラミング、mBot への実装などの学習を通じて、理解できると考えられる。

到達目標 2 については、mBot を制御する Scratch コードがライブラリなので、これまでの学習で経験できる。

到達目標 3 については、「2 台のロボットによるリレー」のアルゴリズムが、センサー、ループ、条件分岐、モーター制御など、多くの要素を応用したアルゴリズムなので、それらの要素を学習したうえで、設計したアルゴリズムをプログラムで実装することで対応できる。

到達目標 4 については、Scratch では変数を定義して値を代入できるが、厳密にはデータ型を定義できないので、データ型の選択は、アルゴリズムと並行して定義するデータ構造の定義の際に学習するのが適切である。

7. おわりに

本研究では、CAS が策定したガイドラインを元に、mBot ロボットとアルゴリズム学習用ツールキットを活用したプログラミング教材を開発した。この教材は仕様、アルゴリズム、プログラム、ハードウェアを関連付けながらアルゴリズム的思考を学習することができる。また、プログラミ

ングに関する学習内容と課題内容を変化させることで、児童の能力に応じた教材を提供できる。

今後は、提案した教材の実装を進め、プログラミングを指導する小学校の教員からのレビューおよび教材の評価実験を行いたい。また、本教材の趣旨を活かして教育を行うことのできる教員の養成にも取り組みたい。

謝辞 本研究は、JSPS 科研費 16K01022 の助成を受けたものです。

参考文献

- [1] 小学校学習指導要領，
http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/05/07/1384661_4_3_2.pdf
- [2] 小学校プログラミング教育の手引き（第一版），
http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/03/30/1403162_01.pdf
- [3] CAS Computing Progression Pathways KS1 (Y1) to KS3 (Y9) by topic Computing At School，
<http://community.computingschool.org.uk/resources/1692/single>
- [4] Inayanti Dwi Kusuma, Fitri Utamingrum, Tetsuro Kakeshita, “A Toolkit to Learn Algorithmic Thinking using mBot Robot”, 情報処理学会九州支部 火の国情報シンポジウム, 2018
- [5] 小池翔太, 小学校第3学年の総合的な学習の時間におけるプログラミング教育のカリキュラム開発の試み, 千葉大学大学院人文公共学府研究プロジェクト報告書, 324, pp. 23-32, 2018.
- [6] 株式会社ベネッセコーポレーション, NPO 法人 CANVAS, “プログラミングで育成する資質・能力の評価規準”,
<http://progstd.org/>
- [7] 佐々木 綾菜, 鷲崎 弘宜, 齋藤 大輔, 深澤 良彰, 武藤 優介, 西澤 利治, 小学校におけるプログラミング教育において活用可能なルーブリックの提案, 日本デジタル教科書学会第6回年次大会, 1T-4C-02, 2017
- [8] 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）”, 2016.
http://www.mext.go.jp/b_menu/shingi/chousa/shotou/122/attach/1372525.htm
- [9] 大日本印刷株式会社, “諸外国におけるプログラミング教育に関する調査研究”, 2015.
http://jouhouka.mext.go.jp/school/pdf/programming_syogaikoku_houkokusyo.pdf