

# ブロック言語とテキスト言語の双方で学習可能な 中学生を対象とするプログラミング教材の提案

小谷 悠介<sup>†1</sup> 望月 久稔<sup>†1</sup>

概要：平成 30 年改訂の学習指導要領において，中学校のプログラミング教育は 2021 年度から全面实施する．小学校はブロック言語を，高等学校はテキスト言語を学習することを念頭に，中学校のプログラミング教育においては，ブロック言語からテキスト言語への移行を踏まえ，双方の言語で扱えるプログラミング教材を提案する．

## 1. はじめに

平成 30 年改訂の学習指導要領 [1] において，中学校のプログラミング教育は 2021 年度から「技術・家庭」科目の「技術」分野で全面实施する．「技術」分野のうちプログラミングに関連する単元として「(D) 情報の技術」において「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」と「計測・制御によるプログラミング」がある [2]．さらに情報活用能力を系統的に育成できるよう，他教科等における情報教育に配慮することも定められた．

小学校においては新たにプログラミング的思考の育成 [3]，高等学校においては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 [4] について学習する．本論文はプログラミング言語において，小学校はブロック言語を，高等学校はテキスト言語による情報教育を念頭に考える．またテキスト言語のプログラミングを学習する際は，ブロック言語の学習を先に行った方が効果的だという研究結果 [5] がある．ゆえに小学校と高等学校の間である中学校のプログラミング教育は，ブロック言語からテキスト言語への移行期間であると考えられる．

以上のことから，本論文は中学生を対象とした，ブロック言語とテキスト言語の双方で学習できるプログラミング教材の提案を行う．提案内容としては，中学校プログラミング教育における学習テーマと，テーマに基づき作成したプログラミング教材と，ブロック言語とテキスト言語の対応表の 3 つである．

プログラミング教材を作成するにあたり，専用のブロック言語と様々なテキスト言語を扱うことができるレゴマイ

ンドストーム EV3<sup>\*1</sup> [6] (以下，マインドストーム EV3) を用いた．テキスト言語においては，データ解析や AI の分野で広く使われており，マインドストーム EV3 でも使用できる Python を用いた．

## 2. 新学習指導要領とプログラミング

### 2.1 中学校「技術」における改訂

平成 30 年改訂の学習指導要領 [2] において，中学校のプログラミング教育は 2021 年度から「技術・家庭」科目の「技術」分野で全面实施する．「技術」分野のうちプログラミングに関連する単元として「(D) 情報の技術」において「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」と「計測・制御によるプログラミング」がある．また情報活用能力を系統的に育成できるよう，他教科等における情報教育に配慮することも求められる．

### 2.2 中学校プログラミング教育の役割

新学習指導要領は，小学校においては新たにプログラミング的思考の育成 [3]，高等学校においては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 [4] について学習することを定めている．学習指導要領において，小学校のプログラミング教育は「プログラミング的思考」を育むことが示されており，プログラミングそのものの習得をねらいとしているわけではない [3]．そのため小学校の情報教育は，子ども向けプログラミング教育の取り組みで幅広く使われている [5] ブロック言語を用いに行くと効果的である考えることができる．一方高等学校のプログラミング教育はアルゴリズムやデータ構造を学習することが含まれており [4]，それらを実現するプログラム

<sup>†1</sup> 現在，大阪教育大学  
Presently with Osaka Kyoiku University

<sup>\*1</sup> レゴマインドストームは LEGO Group の商標

は複雑になる。また複雑なプログラムをブロック言語を用いて記述すると早い段階から長くなり、全体を把握することが難しくなると考えられる [5]。よって、高等学校の情報教育はテキスト言語を用いて行うと有効であると考えられることができる。

これらのことから、本論文は小学校と高等学校の間である中学校のプログラミング教育を、ブロック言語からテキスト言語への移行期間と考える。

また、プログラミング言語は大きく分けて「ブロック言語」と「テキスト言語」の2種類があり、それぞれメリットとデメリットがある。ブロック言語のメリットは、次の3つが考えられる。

- 感覚的に操作できる
- 初学時のハードルが低い
- 文法エラーが発生しにくい

ブロック言語のデメリットは、次の2つが考えられる。

- 少し複雑になるだけでプログラムが長くなる
- 小さな変更にも手間がかかる場合がある

テキスト言語のメリットは、次の2つが考えられる。

- 自由度が高い
- 小さな変更が容易

テキスト言語のデメリットは、初学時のハードルが高いことが考えられる。

またテキスト言語のプログラミングを学習する際は、ブロック言語の学習を先に行った方が効果的だという研究結果 [5] がある。本論文は、中学校のプログラミング教育の教材として、ブロック言語とテキスト言語の双方で扱えるようにする。

### 2.3 マインドストーム EV3 の仕様

マインドストーム EV3<sup>\*2</sup> [6] は EV3 ブロックを核とし、モータだけでなくカラーセンサ、ジャイロセンサ、タッチセンサ、超音波センサ、赤外線センサ、温度センサを制御できる教育用ロボット教材である。さらに、ブロック言語は教育版レゴマインドストーム EV3 ソフトウェアのブロック言語と Scratch、テキスト言語は C 言語、Java、Python、C#、mruby を制御に用いることができる [7]。

本論文でマインドストーム EV3 が中学校のプログラミング教材に適していると考えた理由は4つある。1つ目は、中学生を対象とするプログラミング教材として、ブロック言語とテキスト言語の双方で制御できるからである。2つ目は、ネットワークを用いた通信が可能 [7] だからである。新学習指導要領の「技術」分野「(D) 情報の技術」における内容の「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」に関して、「Wi-fi」「Bluetooth」「赤外線」を利用し、通信の学習ができる。3つ目は、セン

表 1 中学校プログラミング教育で求められる内容と学習テーマの一覧

求められる	学習テーマ
プログラミング的思考力の育成	現状把握と問題設定
	問題解決力、論理的思考力の育成
ロボットの制御	モータ制御のプログラム
	センサによる計測
	センサで計測した値での制御
ネットワーク	ネットワークの利用
	ネットワークを介した機器における入出力
データの計測とその利用	データの計測
	データの利用

サの種類が豊富 [7] だからである。新学習指導要領の「技術」分野「(D) 情報の技術」における内容の「計測・制御によるプログラミング」に関して、「センサを利用した制御」「モータの制御」の学習ができる。4つ目は、データロギングが可能である [7] ため、通常は計測しにくい量や変化を数値化あるいは視覚化して捉え、結果の分析や考察に用いられるからである。さらに、その際に生じるプログラムやデータの送受信においてネットワーク、センサやモータのプログラミングにおいて計測や制御の学習が可能である。

## 3. ブロック言語とテキスト言語の双方によるプログラミング教材

### 3.1 プログラミング教材の提案

前章までの内容から、中学校プログラミング教育をブロック言語からテキスト言語への移行期間と考え、その上でブロック言語とテキスト言語の双方で学習することが有効であると考えた。これらを踏まえ、本論文は中学生を対象とするプログラミング教材の提案を行う。

平成 30 年改訂の学習指導要領から、中学校におけるプログラミング教育で求められる内容は以下の4つが考えられる。

- プログラミング的思考力の育成
- ロボットの制御
- ネットワーク
- データの計測とその利用

各内容から、中学校プログラミング教育における学習テーマを提案する。求められる内容と学習テーマの一覧は表 1 である。続いて提案するテーマに基づき、本研究はブロック言語とテキスト言語の双方で学習できる 17 種類のプログラミング教材を作成した。各教材と教材番号の一覧は表 2 である。

各教材の作成にあたり本研究は、ブロック言語に教育版

\*2 レゴマインドストームは LEGO Group の商標

表 2 ブロック言語とテキスト言語の双方で学習できるプログラミング教材と教材番号の一覧

教材番号	プログラミング教材
1	モーターの制御
2	光の反射
3	音の性質と利用
4	地震
5	雷鳴
6	Python を利用した生物の分類
7	温度変化における物質の融点と沸点
8	プログラミングを用いた交配モデル実験
9	Python を利用したヒストグラムの作成
10	日照時間
11	ブロック言語を用いた等速直線運動の計測
12	車間距離維持
13	赤外線を利用したラジコン
14	門灯
15	ライトレースカー
16	アームを使ってみよう
17	自動運転

レゴマインドストーム EV3 ソフトウェアのブロック言語(以下, EV3 ブロック言語)を, テキスト言語に Python を用いた. テキスト言語に Python を用いた理由は, データ解析や AI の分野で広く使われており, 情報を専門とせずとも今後触れる機会は多くあると考えるからである.

さらに各教材でブロック言語からテキスト言語への容易な移行を実現するため, EV3 ブロック言語と Python の対応表(以下, 対応表)を提案する. 対応表は EV3 ブロック言語と Python の命令を横並びに配置しており, EV3 ブロック言語でのプログラミングを Python でのプログラミングへ機械的に変換するものである. 各対応表は表 5, 6, 7, 8 である.

また中学校の他教科等における情報教育も考慮し, 理科・数学との連携を試みた. 学習テーマと他教科の科目との連携は表 3, 科目ごとの単元とプログラミング教材の対応は表 4 である.

### 3.2 教材「車間距離維持」

本論文はプログラミング教材として「12 車間距離維持」を例にあげる.

「車間距離維持」の教材は, 理科「科学技術の発展」において車間距離維持システムの車両を題材に, 超音波センサとカラーセンサの測定値を用い, モーターの加減速を制御するプログラムを作成させる. 該当する学習テーマは「モーター制御のプログラム」と「センサによる計測」, 「センサで計測した値での制御」である. 各学習テーマの内容と目的

表 3 学習テーマと他教科との連携

学習テーマ	理科			数学
	中 1	中 2	中 3	
現状把握と問題設定	4,5		8	9
問題解決力, 論理的思考力の育成	4,5		8	9
モータ制御のプログラム		1	11	
センサによる計測	2,3		10,11	
センサで計測した値での制御			12,13,14,15,16,17	
ネットワークの利用	13			
ネットワークを介した機器における入出力	7,10			
データの計測	7		10	
データの利用	7		10	

表 4 科目ごとの単元とプログラミング教材の対応

科目	単元	対応する教材
理科	電流と磁界	1
	光の反射・屈折	2
	音の性質	3,5
	地震	4
	生物の特徴と分類の仕方	6
	物質の状態変化	7
	遺伝の規則性と遺伝子	8
	日周運動・年周運動	10
	運動とエネルギー	11
	科学技術の発展	12,13,14,15,16,17
数学	データの分布	9

は次の通りである.

「モータ制御のプログラム」は, モータをプログラムで制御する手法を学習させる. この学習を通してモータの制御を理解するとともに, 回転数や時間を利用して制御できることを理解させることが目的である. 「センサによる計測」は, センサで計測したデータを値で表現することや, 使用法を理解させることが目的である. 「センサで計測した値での制御」は, センサが計測した値を利用してモータやセンサを制御するプログラムを考えさせる. センサの値を反復処理を用いて常に計測させ, 出力された値によって分岐処理を用いて処理を変えるプログラムによって制御できることを学習させることが目的である.

本教材は, 超音波センサを取り付けた EV3 ブロックを用

いて車両を作成し、その車両と対象の距離を常に測定し、距離に応じて車両の速度を変えるプログラムを作成させることで、対象との距離を一定に保つ車両のプログラミングを実現したものである。

練習問題として「前方にタッチセンサを、下方にカラーセンサを取り付けたロボットを直進させ、壁に当たるか赤いラインに差し掛かれればロボットを停止させるプログラム」を考えると、EV3 ブロック言語と Python の双方におけるプログラムは次のようになる。

まず、EV3 ブロック言語で作成したプログラム例は図 1 である。次に、このプログラムを対応表を用いて Python に変換すると、図 2 になる。最後に、Python のみで作成したプログラム例は図 3 となる。このように、EV3 ブロック言語で作成したプログラムは対応表を用いることで機械的に Python のプログラムへ移行できるが、Python のみで作成したプログラムはより最適化した形になることがある。

#### 4. 終わりに

本論文は中学校プログラミング教育をブロック言語からテキスト言語への移行期間と考え、双方の言語で学習することが有効だと考えた。その上で中学生を対象とするプログラミング教材として、新学習指導要領に基づいた中学校プログラミング教育における学習テーマ、それに沿ったブロック言語とテキスト言語の双方で学習できるプログラミング教材、各教材で EV3 ブロック言語から Python への容易な移行を実現するための対応表の 3 つを提案した。

また EV3 ブロック言語から Python へ対応表を用いて変換した場合、Python のみでプログラミングした場合とは異なるプログラムになる可能性があるとした。

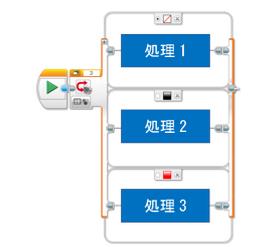
今後の課題としては、作成したプログラミング教材の検証が考えられる。本研究は作成したプログラミング教材を用いた授業実践が行われておらず、各教材の効果が検証できていない。実際に中学校で授業実践をし、生徒と教員からの教材に対する評価、プログラミングに対するイメージの調査を行うことで、より良いプログラミング教材を開発できると考える。

今後は小学校から高等学校にかけてのプログラミング教育において、他の期間における学習をサポートする教材が必要になると考える。

#### 参考文献

- [1] 文部科学省：平成 29・30 年改訂学習指導要領のくわしい内容, [https://www.mext.go.jp/a\\_menu/shotou/new-cs/1383986.htm#section4](https://www.mext.go.jp/a_menu/shotou/new-cs/1383986.htm#section4). (2020 年 1 月 8 日閲覧).
- [2] 文部科学省：【技術・家庭編】中学校学習指導要領(平成 29 年告示)解説, [https://www.mext.go.jp/component/a\\_menu/education/micro\\_detail/\\_icsFiles/afieldfile/2019/03/18/1387018\\_009.pdf](https://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2019/03/18/1387018_009.pdf). (2019 年 12 月 16 日閲覧).
- [3] 文部科学省：小学校プログラミング教育に関する概要

表 5 EV3 ブロック言語と Python の分岐, 反復処理における対応表

EV3 ブロック言語	Python
	if 条件式 : 処理
	if 条件式 : 処理 1 else : 処理 2
	if 条件式 A : 処理 1 elif 条件式 B : 処理 2 elif 条件式 C : 処理 3
	for i in range(回数) : 処理
	while True : 処理
	break
	time.sleep(時間)

- 資料, [https://www.mext.go.jp/component/a\\_menu/education/micro\\_detail/\\_icsFiles/afieldfile/2019/05/21/1416331\\_001.pdf](https://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2019/05/21/1416331_001.pdf). (2020 年 1 月 9 日閲覧).
- [4] 文部科学省：【情報編】高等学校学習指導要領(平成 30 年告示)解説, [https://www.mext.go.jp/content/1407073\\_11\\_1\\_2.pdf](https://www.mext.go.jp/content/1407073_11_1_2.pdf). (2020 年 1 月 8 日閲覧).
  - [5] ビジュアルプログラミング言語の利点、欠点は何ですか?, <https://jp.quora.com/bijuaru-puroguramingu-gengo-no-riten-ketten-ha-nani-desu-> (2019 年 12 月 9 日閲覧).
  - [6] 教育版 レゴ マインドストーム EV3, <https://education.lego.com/ja-jp/product/mindstorms-ev3>. (2020 年 1 月 22 日閲覧).
  - [7] ユーザーガイド(日本語), [https://le-www-live-s.legocdn.com/ev3/userguide/1.4.0/ev3\\_userguide\\_ja.pdf](https://le-www-live-s.legocdn.com/ev3/userguide/1.4.0/ev3_userguide_ja.pdf). (2019 年 12 月 6 日閲覧).

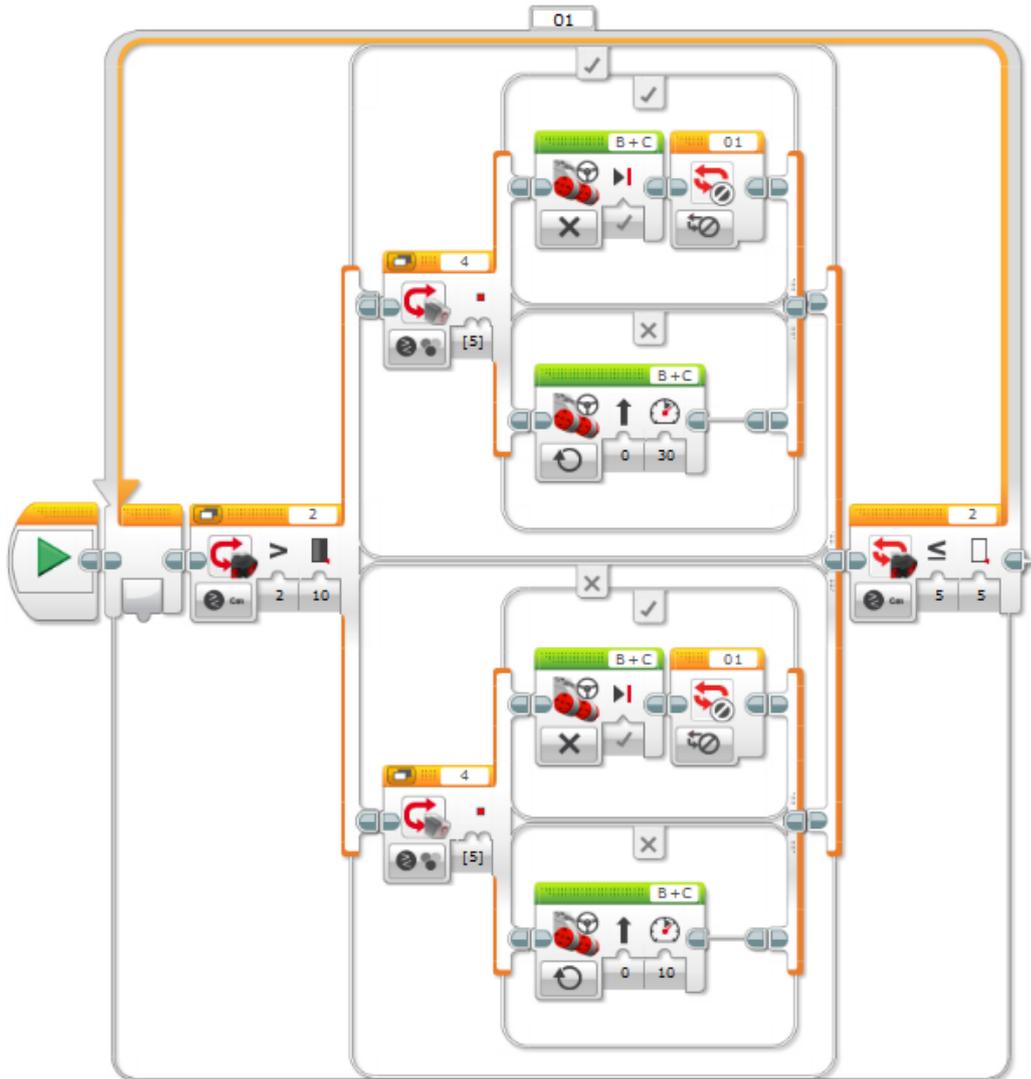


図 1 「車間距離維持」ブロック言語プログラム例

```

while us_sensor.distance() > 50 :
    if us_sensor.distance() > 100 :
        if color_sensor.color() == Color.RED :
            robot.stop()
            break
        else :
            robot.drive_time(100, 0, 100)
    else :
        if color_sensor.color() == Color.RED :
            robot.stop()
            break
        else :
            robot.drive_time(50, 0, 100)
    
```

図 2 対応表を用いた「車間距離維持」Python プログラム例

```

while (color_sensor.color()) != Color.RED
    and (us_sensor.distance() > 50) :
    if us_sensor.distance() > 100 :
        robot.drive_time(100, 0, 100)
    else :
        robot.drive_time(50, 0, 100)
    
```

図 3 「車間距離維持」Python プログラム例

表 6 EV3 ブロック言語と Python の条件式における対応表

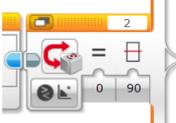
EV3 ブロック言語	Python
	A == B
	A != B
	A < B
	A <= B
	A > B
	A >= B

表 7 EV3 ブロック言語と Python の EV3 ブロックの発光における対応表

EV3 ブロック言語	Python
	brick.light(Color.RED)
	brick.light(Color.YELLOW)
	brick.light(Color.GREEN)

表 8 EV3 ブロック言語と Python の EV3 ブロックの挙動における対応表

EV3 ブロック言語	Python
	gyro_sensor.reset_angle(0)
	brick.sound.file( <span style="border: 1px solid black; padding: 2px;">ファイル名</span> )
	brick.display.text(color_sensor.color(), (60, 50))
	robot.drive_time( <span style="border: 1px solid black; padding: 2px;">速度</span> , <span style="border: 1px solid black; padding: 2px;">角速度</span> , <span style="border: 1px solid black; padding: 2px;">時間</span> )
	
	center_motor.run_target( <span style="border: 1px solid black; padding: 2px;">角速度</span> , <span style="border: 1px solid black; padding: 2px;">目標角度</span> )
	center_motor.run_until_stalled( <span style="border: 1px solid black; padding: 2px;">角速度</span> , Stop.HOLD )