

# 初等・中等教育におけるプログラミング言語の傾向と イベント処理プログラミングについて

## ～micro:bit を中心に～

喜家村 奨<sup>\*1</sup> 稲川 孝司<sup>\*1</sup> 西野 和典<sup>\*2</sup> 高橋 参吉<sup>\*3</sup>

<sup>\*1</sup> 帝塚山学院大学 <sup>\*2</sup> 太成学院大学 <sup>\*3</sup> NPO 法人学習開発研究所

<sup>\*1</sup>susumu@tezuka-gu.ac.jp, <sup>\*1</sup>t-inagawa@tezuka-gu.ac.jp, <sup>\*2</sup>k-nishino@tgu.ac.jp

<sup>\*3</sup>takahasi-san@u-manabi.org

本稿では、初等・中等教育のプログラミング指導で利用されているプログラミング言語の傾向と、その問題点について述べる。また Scratch から micro:bit のプログラミングの接続性を考慮して、開発したイベント処理のための拡張ブロックについて紹介する。

### 1. はじめに

GIGA スクール構想により、2020 年から小・中学校における一人一台端末及び高速通信環境下での学びがスタートしている。さらに今後は、高校でも、一人一台端末の導入が予想される。このような状況の今、プログラミング教育においても、それぞれの校種間の接続性を考慮した教材開発や効果的な指導法が求められている。

本研究の目的は、大きく変わった新学習指導要領の内容を中心に、小学校から中学校への接続、さらに高校への接続を考慮したプログラミング的思考および情報の科学的理解を深めるために教材や指導法を検討することにある<sup>(1)~(3)</sup>。

筆者らは、小学校から高校への接続性、STEAM 教育の重要性を念頭に micro:bit を用いたプログラミング教材を開発している。本稿では、初等教育および中等教育の教科書で用いられているプログラミング言語の傾向、およびその問題点について考察する。また、Scratch からの接続性を考慮して作成した micro:bit の開発環境 makecode エディタの拡張ブロックについても述べる。

### 2. 初等・中等教育の接続性を考慮したプログラミング言語の選択

ここでは、初等・中等教育で利用されているプログラミング言語の傾向とその問題点、および micro:bit の開発環境 MakeCode エディタ（以後、MakeCode）の有用性について紹介する。

図 1 は、初等・中等教育の教科書で用いられているプログラミング言語の傾向を示している。小学校では、Scratch などのブロック型プログラミング言語が多く利用されている。中学校では、ブロック型プログラミング言語に加え、なでしこ、ドリトルなどの日本語入力

型言語と一部 JavaScript も用いられている。高校では、Python, JavaScript, VBA などの実用性の高いテキスト型のプログラミング言語が多く利用されており、一部で Scratch なども用いられている。このような状況であるため、以下のようなことが懸念される。

(1) 高校では、テキスト型のプログラミング言語が多く使われているが、中学までは、テキスト型のプログラミング言語の利用は少ない。テキスト型のプログラミング言語は、文法を理解していないとミスをし、プログラムが動かないことが多く、習得にある程度の慣れが必要である。

(2) クラス内で生徒によって、それまでにならってきた言語がまちまちであり、1つの言語で、授業を進めることが難しい。

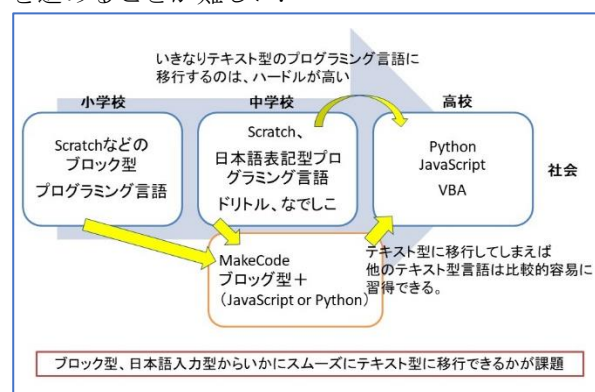


図 1 教科書で使用されている言語の傾向

このような懸念事項を払拭する1つの方法として micro:bit の開発環境である MakeCode の機能について述べる。

MakeCode は、ブロック型のプログラムとテキスト型のプログラミングの編集画面をボタン1つで切り替えることができる。この機能を利用することによって、学習者はブロック型からテキスト型

へスムーズに移行することができ、学習者によって、習得したプログラミング言語が異なる状況でも、学習者個々の習得状況にあった言語選択が可能であり、学習効果の低下を防ぐことができると思われる。

### 3. Scratch から micro:bit のへの接続性を考慮したイベント処理ブロックの開発

自動販売機などの順序機械のイベントを処理するプログラムについて考える。一般に、イベントを処理する方法は以下の2つの方法がある。

- (1) イベント駆動型：システムがイベントを検知して知らせてくれる。ユーザは発生したイベントに対応した処理をイベントハンドラに書けばよく、1つのイベント処理を他と独立して記述できるというメリットがある。反面、イベントハンドラの動作が並列になるため、同期や排他制御などの知識が必要となる。
- (2) イベント検知型：学習者のプログラム内でイベントの発生を検知し、処理を記述する。プログラム全体が逐次的に実行されるため、プログラムの流れを把握しやすいが、CPUの実行時間を浪費する傾向にある。

次に、図2に示したフローチャートとScratchのプログラムを例に、今回、開発したmicro:bitの拡張ブロックの利用と、その有用性について述べる。

図2のフローチャートに示した処理は、まず画像1、次に画像2と順番に表示した後、キー入力を待ち、その後、入力されたキーによって画像を選択表示する。

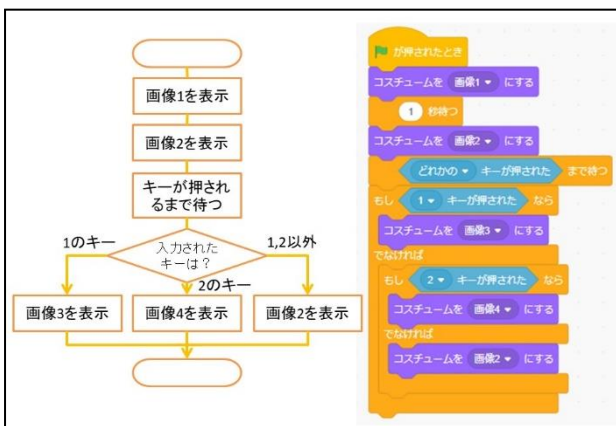


図2 イベントを処理するプログラムの例

Scratchでこのプログラムを作成した場合、図2に示すように、設計のもととなるフローチャートに対応したプログラムを書くことができる。

次にこのプログラムをmicro:bitで実装することを考える。画像表示に代わりにLEDへのマークの表示を使い、キー入力の代わりにボタン入力を

使用するとする。micro:bitにもイベントの発生を検知するブロックがあり、そのブロックを用いるとScratchのプログラム同様にボタン入力処理を記述できるが、イベントが発生するまでループして待つという処理を書く必要があり、フローチャートとの対応が崩れてしまい学習者の理解の妨げになる。そこで、今回、イベントの入力(8つ)を待つ拡張ブロックを開発した(図3)。このブロックを使うとmicro:bitでも図2のフローチャートに対応したプログラムがScratch同様に実装でき、Scratchからmicro:bitへの接続性を保つプログラミング指導ができると考える。

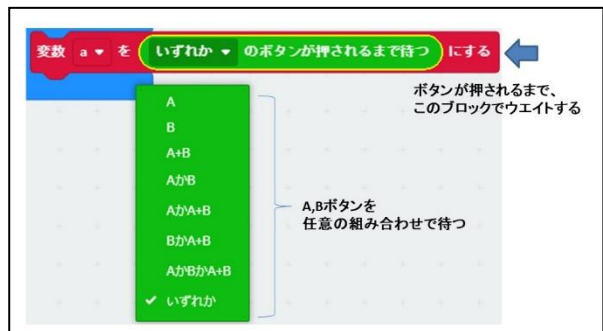


図3 開発したイベント処理拡張ブロック

### 4. おわりに

本稿では、初等教育から中等教育の接続性を考慮したプログラミング言語の選択について、その問題点と1つの解決策を述べた。また、Scratchからmicro:bitへの学習の接続性を考慮した場合に有用だと思われるmicro:bitの拡張ブロックの開発について述べた。

### 謝辞

本研究は2020年度よりJSPS科研費JP20K02528の助成を受けている。また、常日頃から、本研究に対して、的確で、心温まる助言をくださる、NPO法人学習開発研究所の三輪吉和氏に感謝します。

### 参考文献

- (1) 高橋参吉, 喜家村奨, 西野和典:「情報の科学」での「micro:bit」によるプログラミング教育の可能性～小学校から高校までの一貫したプログラミング教育～, 日本情報科教育学会第10回研究会報告書, pp. 10-15 (2018) .
- (2) 高橋参吉, 喜家村奨, 稲川孝司, 西野和典: 「micro:bit」プログラミングで学ぶ情報技術の教材開発, 教育システム情報学会第43回全国大会講演論文集, pp. 205-206 (2019) .
- (3) 喜家村奨, 西野和典, 稲川孝司, 三輪吉和, 高橋参吉: 初等・中等教育におけるプログラミングのための教材開発, 日本情報科教育学会, 第13回全国大会講演論文集, pp. 48-49 (2020) .