

ドリトルと BBCmicro:bit を用いた プログラミングの授業についての考察

岡本牧子^{†1} 日熊隆則^{†1} 大村基将^{†2} 兼宗進^{†2}
琉球大学^{†1} 大阪電気通信大学^{†2}

1. はじめに

2018年8月に石垣市で行われた中学校教員向けの免許更新講習において、ドリトル^[1]とBBC micro:bit^[2]を用いたプログラミングの授業を行った。受講者は、数学や理科、音楽の他高校の情報分野や農業高校担当など幅広い分野の教員11名であった。授業では、ドリトルの文法の学習、ボタンを用いたゲーム作成を行った後、BBC micro:bitの加速度センサーを用いたゲーム作成を行った。本発表は、その記録とこれからの課題をまとめたものである。

2. 授業のねらいと教材準備

2.1 授業のねらい

2020年度から小学校に導入されるプログラミング教育に向け、2018年3月に「小学校プログラミング教育の手引」が発表されたことを受け、全国各地では教材、授業例など様々な先行研究が行われている。沖縄県では各自治体でプログラミング教育への意識に差があり、何よりも、現場教員のプログラミング未経験が対応への消極性を助長している。また、「プログラムを組んでみたことがあるか?」という問いをすると、ほぼ全員の教員から「無い」あるいは「(スクラッチなどの)ブロック形式のソフトを触ったことはあるが、よくわかっていないと思う」という回答が得られる。これは、教員向けの講習などは時間の都合上1時間前後の講習となってしまう、プログラミングソフトの使い方などに終始してしまいがち、あるいはプログラムの説明をしたとしても受講者が自ら試行錯誤する時間がないため、受講者の印象に残りにくいことなどが考えられる。一方で、プログラムの学習に効果的な教材構成として、外部機器との連携が挙げられる。例えば、プログラムを実行すると、LEDが点灯したり、モーターが動いたり、センサーからの値がパソコンの画面に表示されたりする、といった構成である。教育用プログラミングソフトがほぼ無料で準備できるのに対し、これらを実現する外部機器を教育現場に導入するにはそのコストと導入するのに必要な知識や経験が必要となり、小中学校の教員にとって導入に踏み切るにはまだ容易ではない。

そこで、著者らはセンサーや信号の入出力が取り扱いや

すいBBC micro:bitに着目し、教育用プログラミングソフトドリトルと連携させることにより、より効果的なプログラミング学習ができないかと考えた。すなわち本授業では、BBCmicro:bitを利用することで外部機器を使用する際の、基盤とLEDやセンサーなどとの接続部にあまり時間を割かず、ドリトルを利用することで、自分でかいたプログラムの効果が実感できるよう授業を構成した。

2.2 BBC micro:bit を連携したドリトルプログラムの試作

ドリトルとBBC micro:bitの技術的な連携方法については、本稿では記述せず、学習者が必要な操作のみを説明する。図1(a)(b)にBBC micro:bitの加速度センサーの横の傾きを取り込みながら、ドリトルプログラムに反映させる二つのプログラムを示す。学習者は、BBC micro:bitの加速度センサーを常に取り込むため、図1(a)のプログラムを実行する必要がある。次に図1(b)のプログラムで、取り込んだ値をどのように操作するかを決定する。本講義では、自分の意思が反映しやすく、比較的容易なコマンドで作成できる、宝物拾いゲーム^[4]を参考に題材を作成した。シリアルポートの選択を行う操作などは、Arduino基盤の外部機器などを接続する場合と同様であり、学習者はUSB接続された外部機器からの信号を操作しているのを実感することができる。

```

プログラム [1/1]
1 //本プログラムをmicrobitに書き込み、
2 //serial_pc_flower.dtlをドリトルで実行し通信する
3 システム!"microbit"使う。
4
5 繰り返し実行=「
6     シリアル! (加速度センサ! 横の傾き?)出力。
7 」。
8
9 mb!転送。
10
    
```

(a) BBC micro:bit の加速度センサーの値を常に取り込むためのプログラム

```

プログラム [1/1]
1 //serial_mb_acciに対応したプログラム
2 mb=シリアルポート! 1024 115200 作る。
3 mb!(システム! シリアルポート選択)開く。
4
5 かめた=ターゲット! 作る。
6 花=ターゲット! 作る ペンなし (花) 変身する 100 50 位置。
7 花1=花! 作る -100 50 位置。
8 花2=花! 作る 100 -50 位置。
9 *値=フィールド! 作る。
10 かめた:衝突=「相手!相手!消える」。
11 「
12     かめた! 1 歩く。
13     x=(mb!100 読む)/10。
14     x値! (x)書く。
15     かめた! (x)右回り。
16 」。!1000 繰り返す。
17
18 mb!閉じる。
19
    
```

(b) (a)で得られた値の操作を決定するプログラム

図1 加速度センサーを用いて宝物拾いを行うプログラム

†1 MAKIKO OKAMOTO, University of the Ryukyus.
†1 TAKANORI HINOKUMA, University of the Ryukyus.
†2 MOTOMASA OOMURA, Osaka Electro-Communication University.
†2 SUSUMU KANEMUNE, Osaka Electro-Communication University.

3. 授業実践

受講者のほとんどは、プログラミングを行うこと自体が初めてであったため、授業ではオブジェクトの簡単な説明から行い、小学校プログラミング教育の手引にも記載されている四角形や三角形の作図を行った。次に作図を段階的に行うためのツールとしてボタン作成、ボタンを使わなくても自動的にオブジェクトを操作するタイマーの使い方、オブジェクトを複数作成する方法、オブジェクトの画像を変更(変身)させる方法、座標を用いて任意の場所にオブジェクトを移動させる方法などを一斉授業で行った。この過程で学習者はドリトルの文法構造や入力ミスを検索する方法やコツ、数値の変更方法などを学ぶことができる。さらに、習得したコマンドを使って、図2に示すようなりんご拾いゲームの作成と自己流の改良を行うことで、自分で書いたプログラムのデバッグをある程度できるようになることが確認できる。

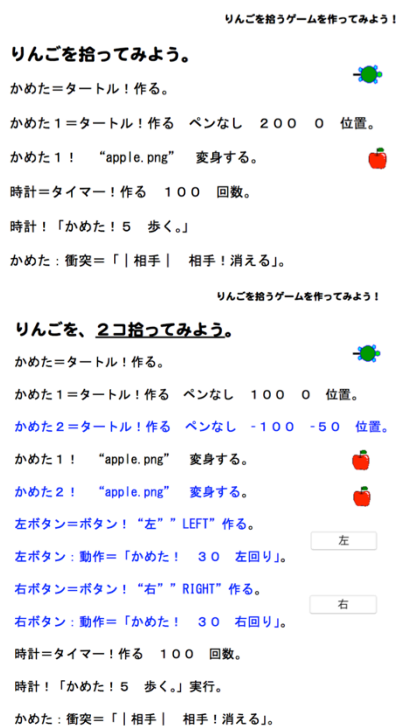


図2 ドリトルの文法やデバッグ練習の例題

上記のゲームが各自で作成できるようになったところで図1に示した例題プログラムを用いて、BBC Micro:bitの加

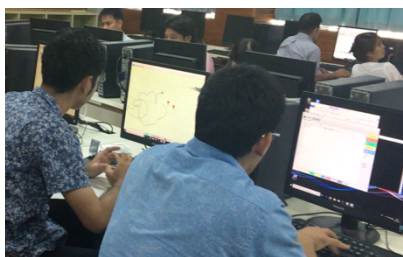


図3 加速度センサーを用いたゲーム作成の様子

速度センサーの値をゲームに取り込む学習を行った。ゲーム作成自体はすでに終了しているため、学習者は取り込んだ値の操作方法だけに集中することができた。

受講後の感想から、「ボタンを押したらモノが動くのが当たり前に感じていたが、プログラムの存在を改めて実感した」、「(micro:bitと一緒に使ってみて)これもできるのかな?とたくさんのアイデアが出てきたことに自分自身驚いた」、「日本語入力で簡単なゲームが作れることにとても驚きを感じた」、「マイクロビットを活用すれば、PC上で作成したプログラムを外に持ち出すことができ活用の幅が一気に広がる」、「(microbitは)通信が安定していたので扱いやすかった」など、生活や社会とプログラミングの関係をそれぞれのレベルで実感することができたようであった。

4. 課題

一方で、受講者の声からは「中学生でもブラインドタイプが定着していないので入力に時間がかかりそう」、「できるだけ円滑にエラーを修正させたい」などの懸念も上がっていたが、この点は現場教員がある程度プログラミングの経験を積み、デバッグ作業を楽しみながら行えるようになれば対応できると考える。また、今回の授業では行わなかったが、それぞれが作成したゲームで対戦させるなど、ゲーム作品を通じた学び合いシステムを授業の中に組み込む事も重要である。さらに、ドリトルとBBC micro:bitを利用すれば図4に示すような簡単な照度計を自分で製作する事も可能であるが、現在の学校現場では自治体によって情報端末のセキュリティやスペックなどに差があり、情報収集とその対応が必要である。

光量センサーを使って照度計を作ってみよう!

1. 端子0と端子1を使って、光量センサーの値が100以下なら端子0のLEDが、光量センサーの値が100以上なら端子1のLEDが光る。

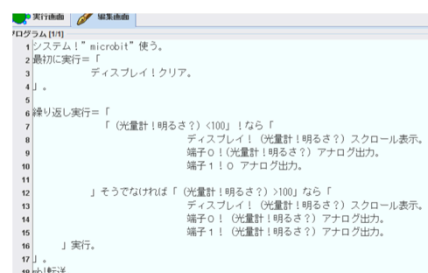


図4 ドリトルとBBC micro:bitを用いた照度計製作例

参考文献

[1] プログラミング言語「ドリトル」、大阪電気通信大学兼宗研究室。
<https://dolittle.eplang.jp>.
[2] The Micro:bit Education Foundation, .
<https://microbit.org/ja/about/>.
[3] 小学校プログラミング教育の手引(第一版), 文部科学省 .
http://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1410886.htm.
[4] 宝ひろいゲーム, 大阪電気通信大学兼宗研究室.
https://dolittle.eplang.jp/ch_drive.