

Scratch を用いた小学生向けプログラミング教育

山守 一徳
三重大学 教育学部

プログラミング的思考を小学校段階から育むように、文部科学省の有識者会議が今年6月に提言し、2020年からの授業実施を目指し、指導内容の検討が進められている。その中で、Microsoft や Apple などが新たなプログラミング教育用ツールを提供し始めている。

筆者は、MIT が開発したプログラミング教育用ツールである Scratch が一部小学校で使われ始めていることから、津市立一身田小学校や津市立西が丘小学校において、Scratch を用いた授業実践を行ってきている。現場の教員にとって、プログラミング的思考の2020年までの授業実施に向けての取り組みは始まったばかりで、指導内容や方法など、戸惑っているばかりである。Scratch は、プログラミングを簡単に組み立てられるようになっており、プログラミング初心者でも容易にチャレンジすることができるツールである。

本論文では、現場の教員の一助になるべく、筆者が行なった学年別の実践内容を紹介するとともに、さらなる活用方法について述べる。

1. はじめに

Scratch は、パソコンに専用ソフトをインストールしてから動かす Version1.4 と Web ブラウザ上で動かす Version2.0 がある。Version1.4 は動かす時にインターネットに接続している必要がないが、Scratch を動かすための専用ソフトをインストールしておく必要があるため、事前準備が大変である。Version2.0 は、インターネットに接続した状態で動かすのが基本であるが、Adobe Air (20 以上) と Scratch Offline Editor というソフトをインストールしておけば、インターネット接続無しで動作できるようになった。Adobe Air の動作要件と Scratch Offline Editor の動作要件の両方が必要であるが、Mac OS X v10.7 以上/Mac OS 10.5 以上/Windows7 以上/Linux (32 bit)等にインストールして動かすことができる。Version1.4 は、Windows 2000 以上/Mac OS X 10.4 以上/Ubuntu Linux 12.04 以上で動かすことができ、低スペックのハードウェアで動かすには、Version1.4 の方が良い。また、モーションカメラの Kinect との連携や、モーター制御・遠隔センサーとの連携は Version1.4 の方が得意である。Version1.4 と Version2.0 では、作成し保存したプログラムのファイル形式が異なり、2つの間に互換性はない。

筆者らは、小学校の PC 教室で Scratch を動かして授業実践するため、インストールが不要な Version2.0 を用いた。

2. Scratch2.0

Scratch.2.0は、https://scratch.mit.edu/projects/editor/?tip_bar=getStartedのURLにWEBブラウザからアクセスすると、図1のような画面が表示される。左側の猫の絵は、スプライトと呼ばれ、図1の右側のブロックの組み合わせのように見えるプログラムによってスプライトを動作させることができる。図1の中程にプログラムの部品となるブロックが置かれており、

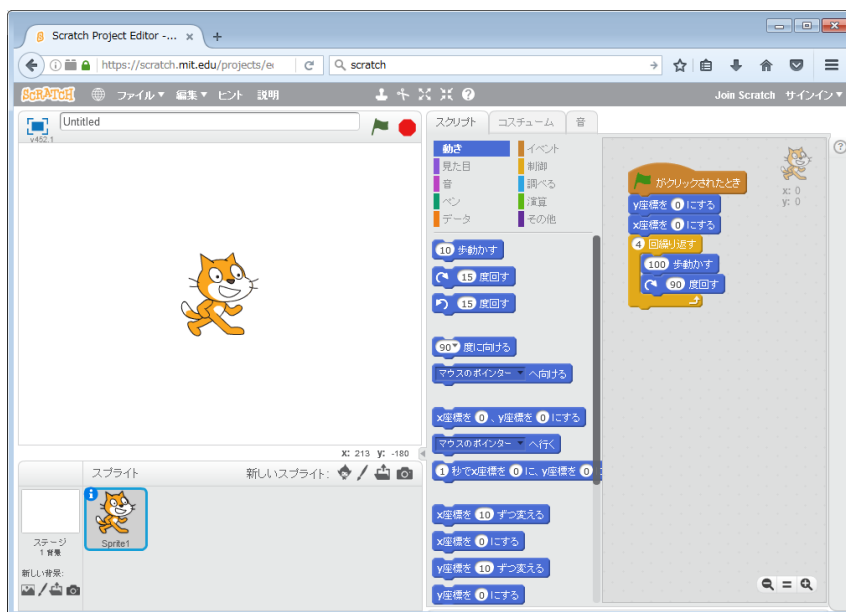


図1 Scratch2.0の初歩画面

その部品を右側のプログラム領域へドラッグ&ドロップしてプログラミングを行う。ドラッグ&ドロップをする時に、連結できる位置へブロックを持っていくと、自動的に適切と思われる位置にブロックが組み込まれた形になる。ブロックをつまんで移動させることでスプライトを動作させるプログラムを作成できるため、小学生でも充分、プログラミングが行なえるようになっている。このツールを使って、小学校において授業実践を行った。

3. 小学校1年生向け

小学校1年生は、パソコン操作に慣れさせることが主目的となる。パソコン教室で授業を行う時に電源を入れる方法から始まって、マウスやキーボードを使う方法を教えて、ログインという小学1年生にとっては不慣れな操作を行わせる。その後、マウスの操作に慣れさせるために、パソコンによるお絵描きを実施するのが通常である。この時、大学生が多数TA（ティーチングアシスタント）として援助すれば、小学1年生が引き起こすトラブルにすぐに対応できる。少人数のTAを参加させるだけでも大きな助けになる。その上で、小学1年生が絵を描く時の問題として、いい加減な絵を描くだけで真剣に最後まで完成させようとする児童がいることがある。パソコンを使ってお絵描きをすると、書き直しがすぐにできるが、マウスや電子ペンで思うように線が描けないことが起き、その点で児童が嫌気を起こしてしまう。TAがいると書き直しをサポートできるので、最後まで諦めずに書き続けさせることができる。

また、Scratchを使うと、絵を動かすプログラムを準備しておいて、描いた絵を動かすことができるので、絵の完成に漕ぎ着けようと努力をしてくれる。さらに、絵を集めるという操作が必要になるが、児童一人一人の描いた絵を先生機上でまとめて動かしてやれば、その中に自分の描いた絵を入れ込んで

もらおうと熱心に取り組んでくれるという利点がある。クラス全員によって一つの作品を完成させようと集団の中での自分の役割を体験させることができる。

(1)「水族館を作ろう！」

児童一人一人に魚など水の中の生き物を描かせて、先生機に集めて表示させることで、水族館のように見せることができる。描いた絵を保存させるには、TAに操作させて、スプライトファイルを先生機上の共有フォルダへ保存させる。保存する時のファイル名には、端末番号+児童の氏名の文字列を使うようにして、どの端末での保存作業が遅れているのかをすぐに把握できるようにする。先生機上では、共有フォルダ上のスプライトファイルをScratchの中に読み込んで、すぐに動かして見せる。この時、動かすためのプログラムは事前に用意しておき、スプライトの絵であるコスチュームが空になった状態のスプライトへプログラムだけを貼り付けた状態でスタンバイしていることを行う。そして、児童の絵のスプライトファイルは、「ファイルから新しいコスチュームをアップロード」のボタンを使うことで読み込まれて、コスチュームが後から追加されていくことを行う。プログラムを先にスタンバイしておくのは、スプライトごとにパラメータが異なったプログラムになっているからである。プログラムをコピーして別のスプライトのプログラムとして引き継がせた場合、スプライトごとにパラメータを直す手間が大変

であったため、プログラムを事前に用意しておき、コスチュームを読み込むという手順を取っている。スプライトの動きは、2パターン用意し、ランダムに動き回る方法と周回して動く方法の2種類用意している。ランダムに動く時は絵が重なることがあり、周回して動く時は絵が重なることがない。周回して見せる動きは図3の矢印で示す順番でスプライトを移動させている。旗印のボタンをクリックした時は、ランダムな動きになり、右下の青色の円形ボタンをクリックした時は、スプライトの並び順をランダムに並び直した後に、周回して移動する動きとなっている。

周回時に全児童の描いたスプライトが重ならないように見せるために、x座標の配列とy座標の配列と周回軌道中の何番目の場所かを示す位置番号配列の3種類の配列を用意している。位置番号配列は、添え字にスプライトの番号が入ると、その配列要素の値が周回軌道中の位置番号を示す作りになっている。x座標の配列とy座標の配列は、添え字に位置番号が入ると、その位置のx座標値、y座標値をそれぞれ示す作りになっている。右下の青色の円形ボタンをクリックした時のスプライトの並び順をランダムにさせるには、位置番号配列の値を入れ直すことで実現している。

また、位置番号によってスプライトの絵の向きを右向きか左向きかを変えており、次の位置番号へ移動する時に魚の頭が移動する方へ向くように見せている。そのため、先生役が児童の描いたコスチュー



図2 水族館を作ろう！の作品例



図3 周回の動作順

ムを読み込む時に魚の頭の向きを手動で統一して取り込んでいる。

スプライトの絵の大きさについては、児童が描く時の大きさは画面一杯に描くのが普通であるが、そのサイズでは水族館の中で表示させるには大きすぎるため、絵の大きさを自動で揃える機能をプログラムの中に入れている。それには、図4で示す縦線を作成しておき、スプライトと交わる縦線の数によってスプライトを大きくしたり小さくしたりしている。具体的には、3本以上の縦線と交わると小さくなるように、2本以下の縦線と交わると大きくなるように、スクラッチの Scale 値を変えている。交わる本数の判定を高速にするために、縦線を2本置きに3つのグループに分け、グループごとに1つのスプライトとして登録することにした。縦線から成るスプライトは3個から成り、図4で示す青色の縦線のスプライトと赤色の縦線のスプライトと緑色の縦線のスプライトの3つである。実際には線の色は見せていない。その上で、児童の描いた絵のスプライトが縦線のスプライトの何個と交わるかで判定する。3個のスプライトと交われば小さくさせ、2個以下のスプライトと交われば大きくさせる。ランダムな動きや周回させる動きの中に、その判定による Scale 値の変更させるプログラムを組み入れている。なお、周回して動く時には、右下あたりに来た時には意図的に絵を大きくして、動きも尻尾を振るような目立つ動きを追加している。児童に自分の絵がそこにやってくる時のワクワク感を持たせるためである。順番に絵が回ってくるので、自分の描いた絵の位置を追いかけるはずであり、ある箇所にくると動きが少し変わってサイズも大きくなるので、見ていて気持ちが昂るはずである。



図4 サイズ均一化のための縦線

(2) 「サバンナをかこう！」

小学校1年生にパソコン上で絵を描かせる授業として、2015年度は魚を描かせ、2016年度は動物を描かせる活動を行った。大草原にいる動物を描かせようとし、象、キリン、馬、牛などの動物のイラスト調の絵を印刷したものを黒板に貼り、大学生による完成作品例をプロジェクタ投影して、描かせるイメージを持たせることから授業を開始した。授業のタイトルは「サバンナをかこう！」と伝えるのであるが、小学校1年生には「サバンナ」の説明も必要である。大草原の中にある大きな生き物を書いてと要求するものの猫の絵を描いたりする児童もいるが、



図5 サバンナをかこう！の作品例

45分の授業の中で描かせる絵を変えさせるのは時間切れになる可能性が高く、「サバンナ」には存在しない動物が出てくるのは諦めざるを得ない。「水族館を作ろう！」の場合と比べると、頭と足と手の大きさや配置を整えるのは大変であり、何の動物を描いているのかがTAが見てわからない場合が多くなる。魚の場合は、形よりも色付けに凝って描くが、動物の場合は、目標の動物らしく見せるにはどのように

描いたら良いのか形が重要になり、色付けに凝ることは少なくなっている。形を整えるのに、何度も書き直し操作を行い、色付けは塗り潰し操作を行ったらそれで終わりという作業になっている。描く動物は1頭だけですと伝えてあっても、先生機の中に取り込まれて描いた動物が動き出すのを見ると、次の動物を描こうと2頭目、3頭目のお絵描きに取り組みようとする意欲的な行動がどのクラスでも起きていた。通常のお絵描きの授業では、時間のある限り一つの作品に手を加えることを続けようとするが、先生機の中で動かして見せると、何頭も多く描こうとするところが違って来る。これは書こうとする動物の種類がサバンナの場合は多くの種類があって、友達に描いた絵をたくさん見てもらおうとするからかもしれない。

事前に準備しておく内容としては、サバンナの場合、動物のいる場所が奥なのか手前なのか遠近感のある見た目になるので、水族館の場合の水槽の上の方か下の方かの見た目と違って、動物の大きさを意図的に変える必要がある、奥の動物は小さく、手前の動物は大きく見せる訳である。そのため、図6のようにサイズ判定のための縦棒を放射線状の配置にした。3本以上の縦棒に交われば小さく、2本以下の縦棒に交われば大きくさせるのは同じである。画面の下の方は縦棒が広い間隔になりスプライトは大きくなる。画面の上の方は縦棒が狭い間隔になりスプライトは小さくなる。



図6 放射状の縦棒

スプライトがランダムに動き回るか、周回を移動するように回るかの動きは、「水族館を作ろう！」と同じ動きである。周回して動く時に、右下あたりに来た時には意図的に絵を大きくして、動きも跳ねるような目立つ動きにしている。やはり、児童に自分の絵がそこにやってくる時のわくわく感を持たせるためである。

4. 小学校5年生向け

小学校5年生は、プログラミング初歩に取り組みさせることが主目的となる。初歩であるから、逐次処理と繰り返し処理を使って動くプログラムを作成させる。この時、プログラミングを独立した教材にするのではなく、ある既存の教科の中の内容と関連性を持ち、その教科内容の理解を深めることと、プログラミ的思考能力を向上させることを両立させる。そこで、算数の中の図形理解を深めことを目指し、プログラミングによって図形を描かせることを行った。

(1) 「星を描こう！」

正五角形の外角を理解させ、一筆書きで

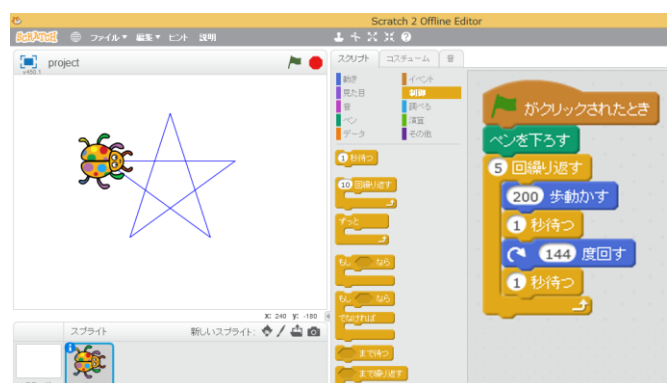


図7 星を描くプログラムと実行画面

星を描かせる時の筆の回転角度を考えさせるプログラム作成に取り組みさせた。実践した対象は、小学校5年生の4クラス（1クラス約33名）であり、各クラスとも45分の1回の授業で行った。

授業は以下の通りに進めた。

Step1: 事前に用意した正五角形を描くプログラムを提示し、実行時の動きを説明する。その中に使われているプログラミングブロックの機能も説明する。「ペンを下す」、「〇回繰り返す」、「〇歩進む」「〇度回す」、「〇秒待つ」、「旗印がクリックされたとき」のプログラミングブロックで正五角形を描くことができるが、「〇度に向ける」「〇座標を〇にする」といった基本的なプログラミングブロックも説明し、移動の仕方、削除の仕方、動作のさせ方などを説明する。

Step2: 正五角形を描くプログラムの組み方を実演して見せた後、同じプログラムを児童に組ませる。

Step3: 正五角形を描く時に何故72度ずつ回転させるのかを説明する。

正五角形を虫が移動しながら描いていくと、虫は書き始めから書き終えるまでに顔の向きを1周回ってきていることに気付かせる。顔の向きを変える箇所は5回あり、 $360 \div 5$ の計算から72度となることを学習させる。

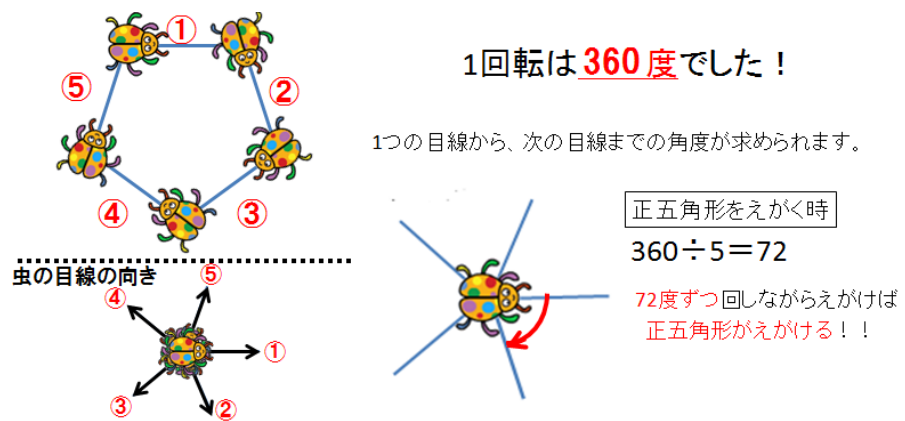


図8 正五角形を描く時の回転角度の説明

Step4: 正六角形、正八角形、一筆書きの星を描くプログラムを作らせる。

パワーポイントを使ってヒントの画像を提示する。

繰り返し回数を変えるのと、回転する角度を変えることで、正六角形、正八角形、一筆書きの星を描くことができる中で、回転する角度は、正六角形は $360 \div 6$ 、正八角形は $360 \div 8$ という計算式で値を求めさせる。

Step5: 星を描く時に何故144度ずつ回転させることになるのかを考えさせる。

一筆書きの星は、虫は書き始めから書き終えるまでに顔の向きを2周回ってきていることに気付かせる必要がある。そして、 $720 \div 5$ の計算式で144の値を算出させる。

Step6: 星を描くことが早くできた児童は、TAが作成した図形の手本を見せて、各自で綺麗な幾何学図形を描くプログラムの作成に挑戦させる。

Step7: 解説、振り返りをする。

星をえがく時は・・・

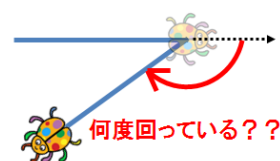


図9 星を描く課題のヒント

授業実践をした結果、Scratch の使い方については、児童はすぐに理解できることがわかった。45 分の授業の中でプログラミングブロックの機能を理解させてプログラミングに着手させるには、サンプルプログラムを提示することが大変有効であり、真っ白な場面からプログラミングブロックを触らせるよりも、手本のプログラミングブロックから触らせた方が、授業の最初の段階がスムーズに入っている。まず簡単なところから始めてすぐに動かせるようになるので、授業実践中、夢中になって取り組む子どもたちの姿をたくさん見ることができた。

図形の内角を学習していない小学校 5 年生にとって、虫の顔の向きを変える角度（外角）は、正五角形の時は 72 度になることを教えるには、 $360 \div 5$ の考え方で教えた方がわかりやすい。通常は、正五角形の中にその中心を三角形頂点とする 5 個の二等辺三角形を当てはめて、二等辺三角形の頂点の角度が、 $360 \div 5$ から 72 度となり、三角形の内角の和 180 度から、二等辺三角形の頂点以外の角度が、 $(180 - 72) \div 2$ の式より 54 度と教えた後に、正五角形の内角の 1 つが 54×2 より 108 度を求めた後に、虫の顔の向きを変える角度は $180 - 108$ の式より 72 度となるという順番で教えるのが普通であろう。（もちろん、正五角形の中を 3 つの三角形で区切って、正五角形の内角の総和は、 180×3 より 540 度を求め、5 か所の角があるから、正五角形の内角の 1 つが、 $540 \div 5$ より 108 度を求めるという方法もある。）しかし、虫の顔の向きを変える方法を使って教えると、内角を教える必要もなく、72 度が算出できる。これを使えば、一筆書きの星を描く時の角度も、同じテクニックで考えることができる。一筆書きの星を描く時に、内角から考えさせると大変であり、小学校 5 年生には 144 度を算出できない児童が多く現れると思われるが、それよりも虫の顔の向きの動きを捉えさせた方が容易である。このような図形の捉え方を経験することにより、図形理解が深まると思われる。

その次に、TA が作成した図形の手本を見せて、各自で綺麗な幾何学図形を描く段階になってから戸惑いが見られる児童が多く現れる。偶発的に綺麗な幾何学図形ができた時や近くの友だちの作品を見て触発された時に、児童間で尋ね合う場面が増えて、とても盛り上がりを見せた。友だちから教えられてから自分でも再現でき、その時にプログラムの中身まで理解できているのかが気に掛かる場所である。

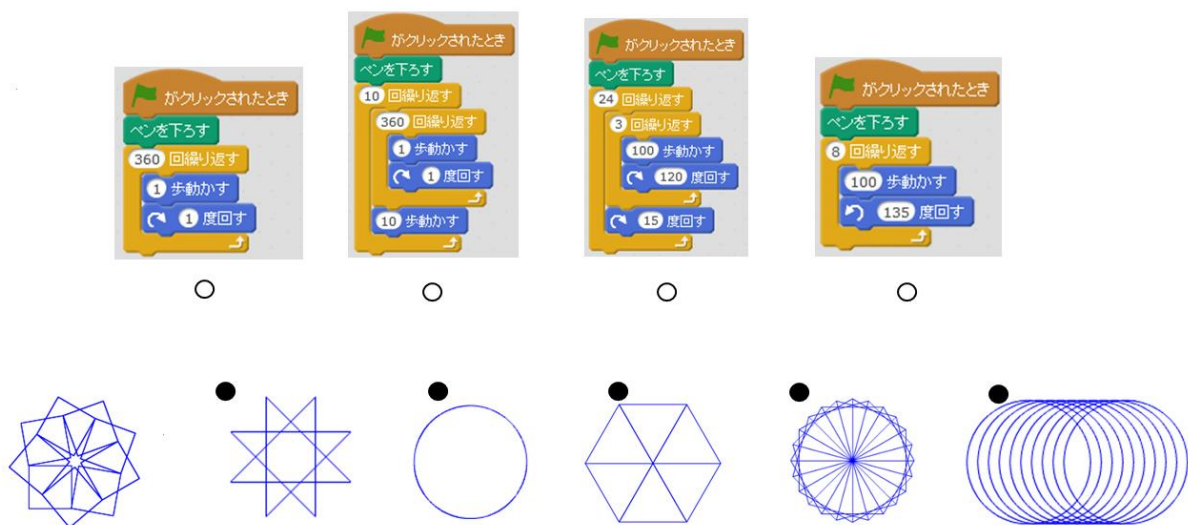


図 10 プログラムの理解度を調べる質問

綺麗な幾何学図形を描くにはどのようなアルゴリズムにしたら良いのか、繰り返し処理と逐次処理の組み合わせだけのプログラムであるが、アルゴリズムが導き出せないということは、まだプログラミング的思考はできていないようである。

そこで、本授業実践後にアンケートを取った。「星を描くプログラムでどうして144度回すのか分かりましたか」という項目のアンケートでは、4段階評価で78.0%の児童が分かった側の回答をした。プログラム自体に対する理解を確認するために、質問紙には図10のように児童が作成した内から図形を描く4つのプログラムを用意し、2つのダミーを含んだ描かれる図形の候補を6つ用意した。プログラムとそれで描かれると思う図形を線で結ばせる形で回答を求めた。その結果、左のプログラムから順に「円(図10下部左から3番目の図形)」、「多数の円(図10下部左から6番目の図形)」、「多数の回転三角形(図10下部左から5番目の図形)」、「星形の八角形(図10下部左から2番目の図形)」が正解であるが、正答率を算出(標本数=121、小数第二位を四捨五入)すると、「円」に関しては80.2%、「多数の円」に関しては74.4%、「多数の回転三角形」に関しては40.5%、「星形の八角形」に関しては62.0%であった。4つの中で比較的簡単な「円」や「多数の円」での正答率は4つの中では高くなっているが、「多数の回転三角形」や「星形の八角形」の正答率は4つの中では低く、プログラムの読み取り能力は、初めてScratchに触れる児童が多かったことも考えられるが正多角形や五芒星の図形理解と比べると、そこまで身に付いてはいないように考えられる。

なお「Scratchを使った授業をまたやりたいか」では、4段階評価で80.8%の児童がまたやりたい側の回答をした。各クラス45分授業の中で、かなり詰め込まれた内容になってしまったが、喜んでもらったように思える。偶発的に綺麗な幾何学図形ができた時の盛り上がりや、またやりたいという気を起こしていると思える。

5. 小学校3年生向け

小学校3年生向けは、パソコンのキーボード操作に慣れさせることを目指し、**QWERTY**配列のキーボードでローマ字入力させることを行う。授業実践はまだ行われていないが、ローマ字表記を小学校3年生の国語科の授業で習うため、その時に、ローマ字の学習と、パソコンキーボードの**QWERTY**配列の習得を兼ねた授業実践を行うと良い。パソコンのキーボード入力に慣れることは、4年生、5年生と年次進行するに従って必要であり、ICT教材を使った演習を行う時に、文章をキーボードで素早く入力する能力が求められる。ローマ字を習うと同時に、ローマ字習得の必要性が感じられる演習を行うことは重要である。小学校3年生向けは、**Scratch**を使って問題提示しキーボードを打たせて正解判定するアプリを提供することが考えられる。

6. 小学校6年生向け

小学校6年生向けは、プログラミングを必要とする演習に取り組みたいところである。よく見られるのがレゴロボットを使うプログラミング演習であるが、レゴロボットは高価である。最近、レゴロボットに代わる「オズロボット」や「ファブロボット」も出てきている。安価なので導入はしやすいが、**Scratch**

はソフトウェアだけで済み、ロボットの動きを画面上で見せることができる。小学校6年生向けは、Scratch のスプライトをロボットのように動かすようにプログラミングさせるのが良いと思われる。

7. まとめ

Scratch を活用する方法について、授業実践を交えて紹介した。小学校1年生向けと5年生向けは実施体験済みのものであり、小学校3年生と6年生向けはまだ未実施のものである。その他にも小学校2年生、4年生向けにも考えられる。塗り絵が動くものとして、「daub だーぶ」という iOS または Android 向けのアプリがある。他にも塗り絵が動くアプリは、「colAR mix」など多数出てきている。授業で使う時に、クラス内の全児童の絵を集めて動かすということがしたい場合には、Scratch は有効である。絵を集める時に、絵のサイズを統一させる機能が必要であり、sprite ファイルは、zip 形式であるため、解凍して中身のファイルを変更し、圧縮し直すことで絵のサイズを整えることができる。sprite ファイルを drag&drop すればスクリプトの絵のサイズや位置を整えるツールを JavaScript で開発もした。このツールを使えば、図4や図6の縦棒を導入する必要はなく、Scratch プログラムの動作負荷を下げることができる。

Scratch を用いるこれらの活動は、現状では、大学生を TA として動員しているのでスムーズに授業が実施できているが、今後は、TA がいない場合でも実施できるようにしていかないと、他の学校現場への普及が難しいと思われる。今回紹介した5年生向けの実践は、1回の授業で納めるために、TA が活躍したが、TA がいない場合には、授業の回数を増やせば、同様のことは実施できると思われる。高学年になると、パソコン操作にも慣れて、各自が対応できるようになってくるからである。低学年の場合は、操作中にデスクトップに突然現れる、ポップアップウィンドウに対応できず、そこで先生の対応待ちに陥り、授業が進まなくなる。TA がゼロでは困るが、数名だけでもいれば、対応できると思われる。

参考文献

- (1) 山守一徳, 吉原健人, 大原敦子: Scratch を用いたお絵描きの授業実践, 情報処理学会第78回全国大会, 1F-04 (2016.3)
- (2) K. Yamamori and K. Yoshihara: Classroom Practices to First Graders by Using Scratch, ISEPST 2016, ISEPST-744 (2016.3)
- (3) K. Yoshihara and K. Yamamori: A Technique for Resizing of Sprite in Scratch, ISTE2016, ISSN2409-1855 Vol.02 No.02 (2016.7)
- (4) 山守一徳, 吉原健人, 稲垣 諒: Scratch スプライトのサイズ均一化ツールの開発, 平成28年度電気関係学会東海支部連合大会, F2-4 (2016.9)
- (5) 吉原健人, 山守一徳, 葛原孝紀: Scratch を用いた図形理解を深める授業実践, 情報処理学会第79回全国大会, 1ZC-04 (2017.3)

- (6) 稲垣 諒, 山守一徳: Scratch コスチューム画像のサイズ均一化ツールの開発, 情報処理学会第 79 回全国大会, 1ZC-05 (2017.3)
- (7) 山守一徳, 稲垣 諒, 中川真里: Scratch を用いる小学校 1 年生のお絵描き実践, 2017 年電子情報通信学会総合大会, D-15-1 (2017.3)
- (8) オズロボット: <https://ozobot.jp/> (2017 年 3 月現在)
- (9) ファブロボット: <http://fabbot.info/> (2017 年 3 月現在)
- (10) daub だーぶ: <http://daub.knowledge-works.co.jp/> (2017 年 3 月現在)
- (11) colAR mix: <http://www.quivervision.com/> (2017 年 3 月現在)