

1 はじめに

新学習指導要領ではプログラミング教育に関し、「目的に応じたアルゴリズムを考え表現し、プログラミングによりコンピュータや情報通信ネットワークを活用する力を育むこと」を目標としている。また新学習指導要領解説によると、その手段の一つとして「プログラムを作成する力、作成したプログラムの動作を確認したり、不具合の修正をしたりする力などを養い、その際、プログラミングによって問題を解決したり、コンピュータの能力を踏まえて活用したりする力などを養うこと」を挙げている。そこで今回は、昨年度の研究で紹介した『G o o g l e B l o c k l y』を題材として、具体的な授業教材の研究および実践的な教材作成をすることを目標とした。

2 概要

G o o g l e社がウェブ上で提供している『G o o g l e B l o c k l y』は教育用ビジュアルプログラミングソフトであり、スクラッチのようにブロックを組み合わせて様々な演算を可能にする。また、ブロックで作られた演算をJ a v aなどの言語に変換させることもでき、具体的なプログラミングの記述との関連を学ぶこともできる。それらの特長を生かして、プログラミングの基礎を学ぶことのできる授業教材について研究したい。また高校数学で学習する内容をB l o c k l yによって実装することで、教科横断的な課題についてJ a v a S c r i p tによる数的処理プログラミングの例を紹介し、課題解決のためのプログラミングであることを意識づけられるようにしたい。なお今回の研究はこれまでの研究を引き継ぐという意味から昨年までの資料を引用している部分があるため、ご了承いただきたい。

3 研究内容

(1) スクラッチについて

スクラッチとは、マサチューセッツ工科大学（M I T）のメディアラボが開発したプログラミング学習用ソフトである。様々な言語に対応しており、世界中のプログラミング教育で最も使われているソフトと言われている。NHK教育テレビでスクラッチの番組が放送されたり、小学生がスクラッチを利用したゲームを制作して発表するイベントが開催されたりと、教育の場で使われることも多い。実際に小学校の授業で使われたという声を聞いたこともあり、信頼性は高い。

特徴は「ビジュアルプログラミング」と呼ばれる、視覚的なオブジェクトで構成されたプログラミング言語であることで、命令の書かれたパーツをパズルのように組み合わせることで簡単に作品を作ることができる。わかりやすさや、命令によって可能な制御の幅広さ、エラーに対する融通の効き方は評価が高い。

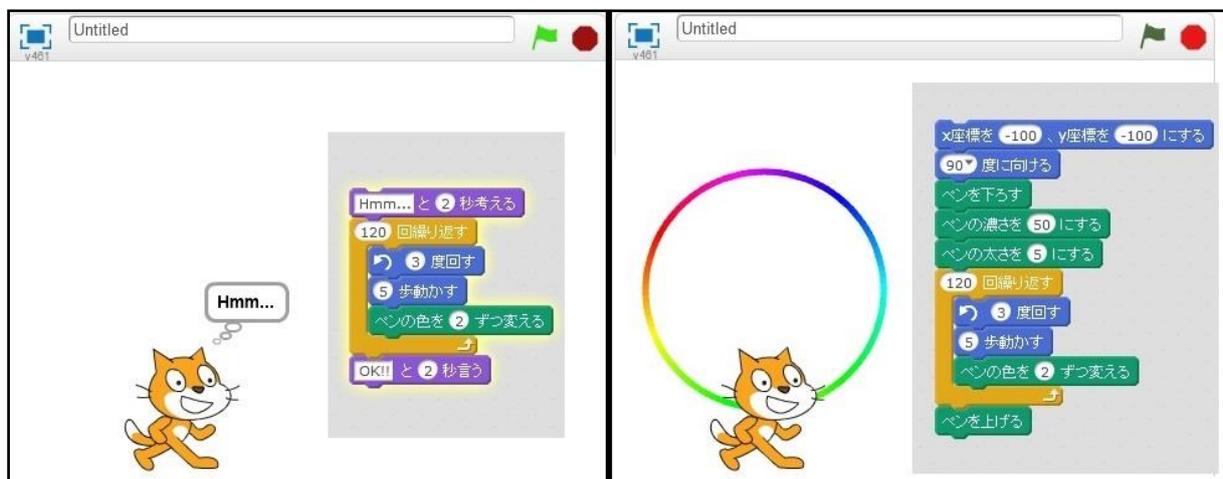
授業としては、ビジュアルプログラミングの導入として利用すると良いと思われる。

例えば（図1-a）は、オブジェクトの初期位置を指定した上で一回転させる命令文である。

（図1-b）はそれに加えてオブジェクトに吹き出しを表示させたり、軌跡をカラフルな線でなぞらせたりしたものである。



（右上：図1-a 右下と左下：図1-b）



上から順番に命令を実行していくことや、ブロックで挟んだり繋げたりして一連の命令を作ることなど、ビジュアルプログラミングがどんなものかを直感的に理解することが出来る。

(2) Google Blocklyによる授業教材

Google BlocklyとはGoogle社が提供するビジュアルプログラミング言語である。各種プログラミング言語に対応した命令が、シンプルなブロックで視覚化されており、それらを組み合わせることでプログラミングを行うことが出来る。Blockly Gamesのページには、Blocklyを用いたゲームがいくつか紹介されており、中でも「迷路」はパズルのような感覚で反復や条件分岐を、「タートル」は以前ドリトルで紹介したような、オブジェクト移動による軌跡の描写をおこなうことができる。どちらも全10ステージ用意されているが、すべてを解くためには繰り返しや条件分岐など様々な考え方を組み合わせる必要であるため、それらを考えさせるだけでも十分な論理的思考力の訓練になる。

ア Google Blockly インストールおよび使用手順

(ア) ファイル（オープンソース）をダウンロード

GitHub で公開されており、下記の画面から master.zip をダウンロードする。

<https://github.com/google/blockly>



(イ) 解凍

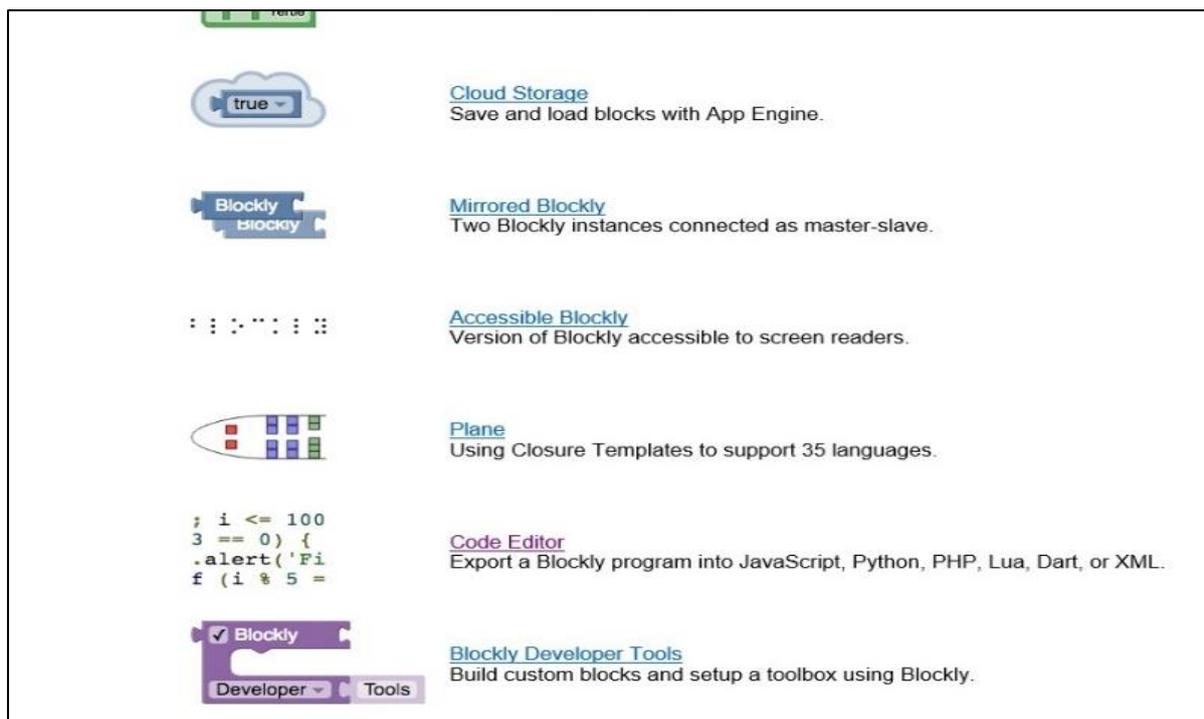
解凍時にアドレスの定義づけを行っているためか、解凍後にディレクトリを移動させると正常に動作しないことがあるので注意が必要である。

(ウ) demos ディレクトリ内にある index.html を実行

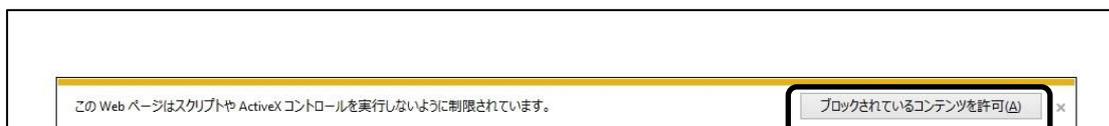
デフォルトソフトが Microsoft Edge になっている場合が多いので、右クリックからプログラムを指定して開くで、Internet Explorer を使用した方がよい。

(エ) 図のような画面が出るため、その中から「Code Editor」を選択

(オ) スクリプトおよび ActiveX コントロールのコンテンツを許可



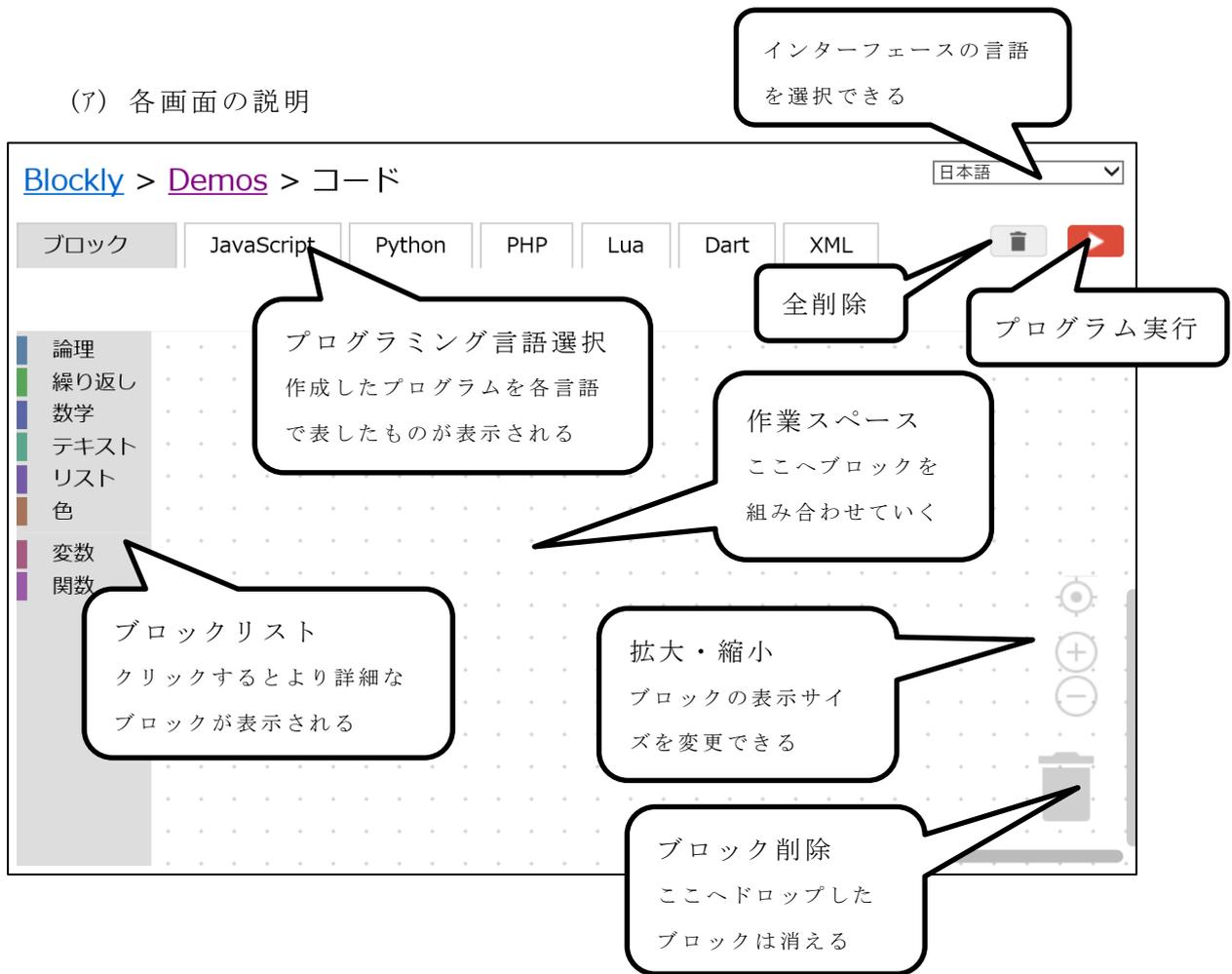
実際の index.html の画面は上図のようである



動作の際にはスクリプトを許可するようポップアップが表示される

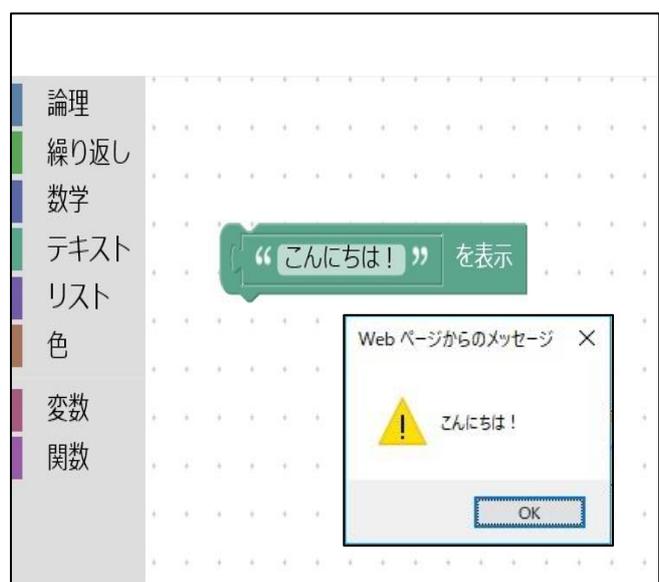
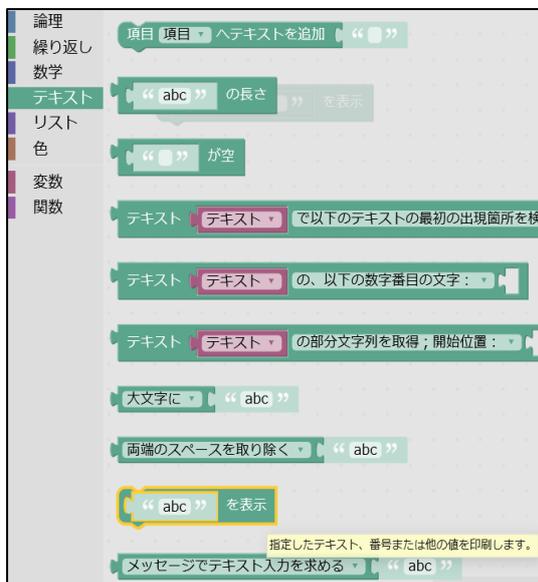
イ GoogleBlockly「Code Editor」について

(ア) 各画面の説明

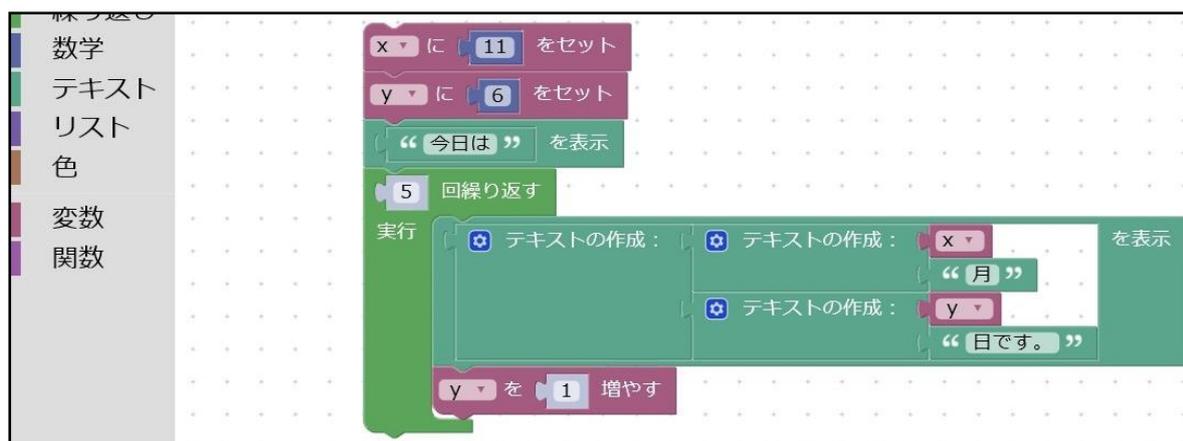


(イ) 使用方法

- ・ リストから命令を選び、作業スペースにドラッグ&ドロップする。
- ・ ブロックを組み合わせる命令を作成する。
- ・ 実行ボタンを押すと、プログラムが実行される。



(ウ) 授業実践例 入力した日付を表示するプログラム



時間がかかる生徒はいたが、周りと協力しながらどのようにブロックを組めば良いか考え、ほとんどの生徒が一時間の間に実装までたどり着くことができた。

(エ) 授業後のアンケート結果

1	プログラミング はどうですか？	とても 難しかった	難しいが できる	意外と 簡単だった	超余裕
		5人	27人	5人	0人
2	もっと高度なこ とをしたいと思いますか？	とても思う	少し思う	あまり 思わない	まったく 思わない
		5人	23人	8人	1人
3	プログラミング のイメージはつ かめましたか？	はい	少し	あまり	ぜんぜん
		17人	18人	1人	1人

感想（一部を抜粋）

- ・ プログラミングはブロックにしたら簡単だけど文字で表すととても難しいので、改めてパソコンは凄いなと思った。
- ・ コンピュータは内部で複雑な命令を出されて動いていることを実感した。
- ・ たくさんある選択肢の中で、正解となるブロックを探すのは難しかったけれど、成功したときの達成感はとてもよかった
- ・ ブロックをたくさん使ってしまったので、もっと簡単に作成する方法があると知りうまく使えるようになりたいと思いました。

多くの生徒が、ビジュアル化によって簡単に実装できる事へ関心を強く抱いていた。また自分なりに工夫しようとする生徒も見られ、個々の理解度に応じて実習できていた点は良かった。しかし一時間の授業では、プログラミングによって問題解決をさせたいという到達目標へは遠く及ばないため、これを複数時間かけてどのように計画立てられるかが今後の課題であると感じた。

イ JavaScriptによるプログラミング例

問 『repeat』ブロックを利用して、以下のルールに沿ったプログラムを作ってください。

- ① 50より大きい変数まで、変数作成および50以上かの判定を繰り返す
- ② 50より大きい変数が出たら処理を終了する

答



注意点は、以下の3点です。

- 初めに変数へ0をセットする
- 『繰り返し』ブロックの反復条件を、50以下に設定する
- 『繰り返し』ブロックの反復処理させる条件に気を付ける

この問題で作ったプログラムをJavaScriptでプログラムすると以下の通りであり、Blocklyがいかにかかりやすく視覚化されているかわかる。

```
var Count;
function mathRandomInt(a, b) {
  if (a > b) {
    var c = a; a = b; b = c;
  }
  return Math.floor(Math.random() * (b - a + 1) + a);
}
while (Count <= 50) {
  Count = mathRandomInt(1, 100);
  window.alert(String(Count) + String('を作りました。'));
  window.alert(Count<=50?'50より小さい!やり直し':'50より大きい!やったね');
}
window.alert('処理を終わります。');
play_arrow
```

ウ 数学的な処理とプログラミングの関連

整数の性質（数A）で学習するユークリッドの互除法を実装すると、以下のよう
なプログラミングとなる。他のプログラミング言語で一から実装しようとする
と高度な技術を要するが、ブロックならば視覚的にもわかりやすく、関心の高い
高校生ならば十分実装可能であると思われる。

【ブロック】

The diagram shows a Scratch-style block-based program for calculating the Greatest Common Divisor (GCD) of two numbers, 91 and 33, using the Euclidean algorithm. The program starts by setting variables: 'firstA' to 91, 'firstB' to 33, 'A' to firstA, and 'B' to firstB. A loop is initiated with a condition 'R = 1 or R = 0'. Inside the loop, 'R' is set to the remainder of 'A' divided by 'B'. Text blocks are used to display the current values of 'A', 'B', and 'R', along with the division symbol and the word 'remainder'. Then, 'A' is set to 'B' and 'B' is set to 'R'. The loop continues until 'R' becomes 0. Two conditional blocks handle the final result: if 'R' is 0, it displays 'firstA と firstB の最大公約数は A'; if 'R' is 1, it displays 'firstA と firstB は、互いに素'.

【プログラムのアルゴリズム】

「 $A \div B = \text{〇〇}$ 余り R 」という計算をして、次の割られる数として B を、
次の割る数として R を変数に代入する操作を行う。これを R が 0 か 1 になるまで繰
り返し、最終的に R が 0 となったら割り切れたと判断して、最後に割った数を最大
公約数として表示する。また R が 1 となった場合は割り切れなかったと判断して、
最初の 2 数は互いに素であったと表示する。

【対応する JavaScript】

```
var firstA, firstB, A, B, R;

firstA = 91;
firstB = 33;
A = firstA;
B = firstB;
while (!(R == 1 || R == 0)) {
    R = A % B;
    window.alert(String(String(A) + String(' ÷ ' + String(B))) + String(' ' +
String(' あまり ' + String(R))));
    A = B;
    B = R;
}
if (R == 0) {
    window.alert(String(String(firstA) + ' と ' + String(String(firstB) +
String(' の 最小公倍数は' + String(A))));
}
if (R == 1) {
    window.alert(String(String(firstA) + ' と ' + String(String(firstB) + '
は、互いに素')));
}

0 == 0;
```

4 まとめ

プログラミング教育に関しては、高校段階でどの程度の内容が求められるのかわからない要素も多く、またそういった分野について知識面や技術面で不安を抱えている教員も多いと思われる。そういった中でGoogle Blocklyを初めとするビジュアルプログラミングや、様々な場面で紹介されている簡易的なプログラミング実装モデルは、児童生徒はもちろん、教員にとっても良い研修の材料になると考えられる。今回のような内容を、課題解決能力育成を図る授業まで昇華させるにはかなりの時間と労力がかかると思われるが、まずは少しずつでも経験と研さんを積まなくてはならないと、今回の研究を通じて感じる事が出来た。今後も様々な情報を得つつ、生徒のためになる授業となるよう研究を深めていきたい。