

# 小学校でのプログラミング必修化にともなうプログラミング指導法の提唱 —Scratchを使用して—

Programming classes are compulsory in elementary school.  
So I advocate programming teaching methods.  
- Using Scratch -

小野 功一郎  
ONO Koichiro

## 要 旨

平成32年度（2020年度）から実施される小学校新学習指導要領では、「プログラミングを体験しながら論理的思考力を身に付ける」<sup>1)</sup>と明記され、プログラミング教育が必修化となる。

児童にわかりやすく教えるにはどうすれば良いのか？どのようなプログラミングシステムがあるのか？

本著ではプログラミングシステムを比較研究し、最も代表的なScratchを取り上げ、実際にプログラミンしながら図工を例として授業をおこなう教育内容・教育教材を提案する。

## Abstract

In the elementary school curriculum guidelines for new study to be implemented from fiscal 2020, "To acquire logical thinking skills while experiencing programming". It is Compulsory as "programming education".

Educate easy to understand for children. How can we do it? What kind of programming system is there?

I selected the most representative scratch. I propose educational contents / teaching materials for doing classes of art and craft as an example.

キーワード：プログラミング教育 ICT教科指導法 小学校新学習指導要領 デジタル教材 教育工学

keywords : Programming Education ,ICT Subject Elementary school curriculum guidelines for new study, Digital Teaching Materials, Educational Engineering

## I. はじめに

### 1. プログラミングと新学習指導要領

小学校でのプログラミング必修化が平成32年度（2020年度）から実施される小学校新学習指導要領では、「プログラミングを体験しながら論理的思考力を身に付ける」<sup>1)</sup>と明記され、そのための学習活動として「算数」「理科」「総合的な学習の時間」で取り上げる際の内容や扱いについて一部例示され、プログラミングに関する新教科は設けず、あくまで既存の教科の中で、プログラミングを教育することになっている。

具体的には各学校・教員が決めて教育をすることになっている。しかし、プログラミング教育には多くの教育ノウハウが必要であり、本を読めば、講義を聴けば、それで教育ができるというものではない。

プログラミングを通じて本当に教えるべきことは何なのか？どのように教えればよいのか？本著では図工を例として授業をおこなう教育内容・教育教材を提唱する。

### 2. プログラミング教育の必要性

なぜプログラミング教育が必修化されたのか？それは21世紀型スキルといわれる「生きていくための教養」であるからと考える。これからの社会を生き抜くには自分で問題を発見し情報を活用し他の人と協力しながら、答えのない問題を解決して行く力が必要だ。この能力とプログラミングは関連性が高い。決してプログラマーを養成しようというのではなく、プログラムを学ぶことに大きな意味がある。どのような意味があるのかを次に具体的に述べる。

## II. 研究方法

### 1. プログラミング教育の意義

#### 1) 楽しい・面白い・感動

プログラミングをすることは、遊ぶことと同じく楽しい・面白いことである。自分で考えたことやイメージしたことが、画面の中で動く、初めてプログラミングをし

\*大和大学教育学部教育学科（初等幼児教育専攻）

平成29年12月12日受理

た児童が感動し笑顔になる瞬間だ。ネットに繋がったコンピュータかタブレット（型端末）が1台あれば、プログラミングはできるのだ。

### 2) 想像力を育成する

プログラミングは創造力を刺激する。「つくる楽しさ」「動かす感動」「理解する喜び」「新しい挑戦」がおこなえるからだ。プログラミングを通じ遊びの中に学びがある、これが想像力を育成するのだ。

これからの時代、どんどん提案ができて実行できる人が価値ある人材である。新しい価値を生み出す訓練として、プログラミングはとて有用である。

### 3) 問題解決能力の育成

プログラミングの基礎を学べば、自分の作りたいモノを作ることができる。与えられた問題を解くというのではなく、自分の作りたいモノを作るというプロセスの中で、問題解決型の取り組みを行うことになる。図1に示すようにまさしくPlan（計画）→Do（実行）→Check（評価）→Act（改善）→のPDCAサイクルだ。自分で計画し問題と向き合い一つ一つ可決をしていく、出来上がれば更に良いモノに改良するといった具合だ。



図1 PDCA サイクルについて

### 4) 論理的思考力の育成

プログラミングを行う中で、手順が一貫していて筋が通っていなければ、プログラムを実行しても正しく動かすことはできない。一貫していて筋が通っている考え方が必要である。なぜ正しくない動きをするのか？現状の原因と結果を考えてプログラムを改善する。この現状の原因と結果を考えて、理想の状態に持つていくための改善策を考えられる思考こそが論理的思考である。

これからの時代、誰もが経験をしたことのない問題を解決していかなければならなくなる。どのように解決していくかという訓練として、プログラミングはとて有用である。

### 5) 自己肯定感の育成

内閣府の行った調査では日本の若者は諸外国と比べて、自己を肯定的に捉えている者の割合が低く（図2）、自分に誇りを持っている者の割合も低いと判明した。

日本の若者のうち、自分自身に満足している者の割合は5割弱（図2）、自分には長所があると思っている者の割合は7割弱（図3）で、いずれも諸外国と比べて日

本が最も低い（図2）。年齢階級別にみると、特に10代後半から20代前半にかけて、諸外国との差が大きい（図5）。<sup>3)</sup>

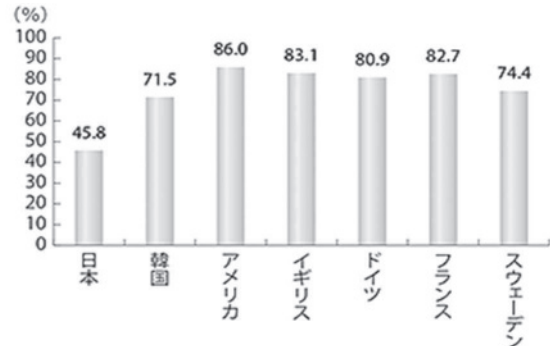


図2 自分自身に満足している（全体）<sup>3)</sup>

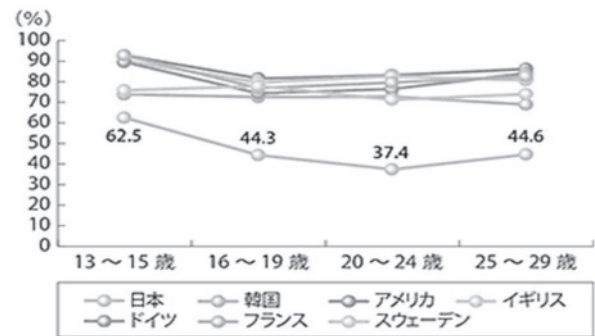


図3 自分自身に満足している（年齢階級別）<sup>3)</sup>

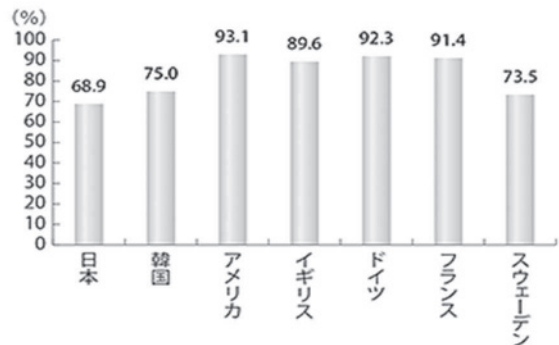


図4 自分に長所がある（全体別）<sup>3)</sup>

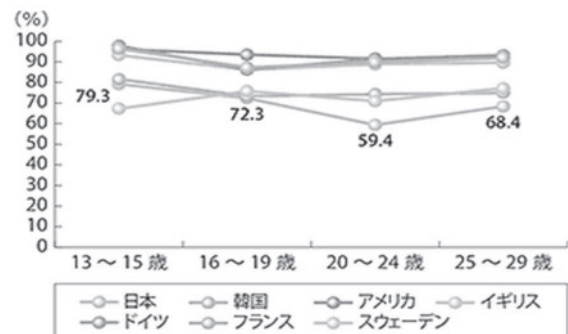


図5 自分に長所がある（年齢階級別）<sup>3)</sup>

プログラミングは「作りたい」から「できた！」までの過程が短く、成功した達成感から自己肯定感の育成の育成になる。

6) 将来の選択肢の幅を広げる

野村総合研究所は2016年1月12日の研究報告講演会と記者説明会で「日本の労働人口の49%が人工知能やロボット等で代替可能に」と題するニュースリリースを出した(表1)。国内601種類の職業について、それぞれ人工知能やロボット等で代替される確率を試算したところ「10~20年後に、日本の労働人口の約49%が就いている職業において、それらに代替することが可能」との推計結果が得られた。

表1は人工知能やロボット等による代替可能性が高い100種の職業を示したものである。

表1 人工知能やロボット等による代替可能性が高い100種の職業<sup>2)</sup>

(50音順、並びは代替可能性確率とは無関係)

I C生産オペレーター	製粉工
一般事務員	製本作業員清涼飲料ルート
鋳物工	セールス員
医療事務員	石油精製オペレーター
受付係	セメント生産オペレーター
A V・通信機器組立・修理工	繊維製品検査工
駅務員	倉庫作業員
N C研削盤工	惣菜製造工
N C旋盤工	測量士
会計監査係員	宝くじ販売人
加工紙製造工	タクシー運転者
貸付係事務員	宅配便配達員
学校事務員	鍛造工
カメラ組立工	駐車場管理人
機械木工	通関士
寄宿舍・寮・マンション管理人	通信販売受付事務員
C A Dオペレーター	積卸作業員
給食調理人	データ入力係
教育・研修事務員	電気通信技術者
行政事務員(国)	電算写植オペレーター
行政事務員(区市町村)	電子計算機保守員(I T保守員)
銀行窓口係	電子部品製造工
金属加工・金属製品検査工	電車運転士
金属研磨工	道路パトロール隊員
金属材料製造検査工	日用品修理ショップ店員

金属熱処理工	バイク便配達員
金属プレス工	発電員
クリーニング取次店員	非破壊検査員
計器組立工	ビル施設管理技術者
警備員	ビル清掃員
経理事務員	物品購買事務員
検収・検品係員	プラスチック製品成形工
検針員	プロセス製版オペレーター
建設作業員	ボイラーオペレーター
ゴム製品成形工(タイヤ成形を除く)	貿易事務員
こん包工	包装作業員
サッシ工	保管・管理係員
産業廃棄物収集運搬作業員	保険事務員
紙器製造工	ホテル客室係
自動車組立工	マシニングセンター・オペレーター
自動車塗装工	ミシン縫製工
出荷・発送係員	めっき工
じんかい収集作業員	めん類製造工
人事係事務員	郵便外務員
新聞配達員	郵便事務員
診療情報管理士	有料道路料金収受員
水産ねり製品製造工	レジ係
スーパー店員	列車清掃員
生産現場事務員	レンタカー営業所員
製パン工	路線バス運転者

※職業名は、労働政策研究・研修機構「職務構造に関する研究」に対応

また、文部科学省の「教職員等の指導体制の在り方に関する懇談会提言」では「今の子供たちの65%は、大学卒業時に今は存在していない職業に就く」という提言している。

先が見えない社会でどんな仕事にも役立つ技術として考えられるのがプログラミンだ、プログラミングを学べば児童たちの将来の選択肢の幅を広げることになる。

7) コミュニケーション能力の育成

プログラミングは一人で画面に向かいこつこつ一人ですること、コミュニケーション能力の低い人が向いている、という人がいますが、これは大きな間違いだと声を大にして述べる。プログラム開発には人と人のコミュニケーションが大切であり、コミュニケーション能力も優れている必要があり、Scratchというプログラミングシステムではネットを活用し作品を通じて仲間と繋がることができるのだ。

制作した作品は公開ができ、公開すると世界中の仲間

との交流が生まれ、仲間の反応を得ながら新たな作品づくりのアイデアにもつながる。

## 2.プログラミング教育の世界情勢

表2に示すように、海外ではすでに小学校での本格的にプログラミング教育が始まっている。

表2 小学校でプログラミング教育を実施している国

	開始年	国名
必修	2003	ハンガリー
	2005	インド
	2009	ロシア
	2014	イギリス
	2016	フィンランド
一部で実施	開始年	国名
	2004	イタリア
	2010	スウェーデン
	2012	エストニア
	—	シンガポール

「諸外国におけるプログラミング教育に関する調査研究」(文部科学省平成26年度・情報教育指導力向上支援事業) 報告書を基に著者作成

あの「Skype」発祥の国エストニアでは、世界で初めて一部の小学校で小学1年生からのプログラミング教育を行った。エストニアでは、IT教育が国家戦力として位置付けられている。

フィンランドでは小学1～2年生からゲームなどを作るプログラミング教育を行っている。

イギリスではコンピュータをいかに使うかという教育を行っていたが、2014年に方向転換して小学1年生からプログラミング教育を必修化「コンピュータで作る・考える」を重視した教育を行っている。

表3 イギリスでのプログラミング教育実施状況

【学年】の縦付け部分は義務教育期間

年齢	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	
学年			1	2	3	4	5	6	7	8	9	10	11	12	13							
教育制度			初等教育				中等教育						高等教育									
教育機関			プライマリースクール(6年間)				セカンダリースクール(5年間)					シックスフォーム(2年間)		大学								
Key Stage			1		2		3			4												
実施科目名			「Computing」																			
位置付け			必修																			
内容			「プログラミング教育」含む																			
学級担任か教科担任か			学級担任										教科担任									
備考																					KS4は事実上選択	

「諸外国におけるプログラミング教育に関する調査研究」(文部科学省平成26年度・情報教育指導力向上支援事業) 報告書<sup>4)</sup>

表3にイギリスでのプログラミング教育実施状況を示した。では実際にイギリスの小学校でのコンピュータ教育(プログラミング教育を含む)に使われている教科書の内容を見てみることにする。

表4 イギリスでのコンピュータ教育(プログラミング教育を含む)に使われている教科書

No.17

- タイトル: International Primary Computing 1
- ISBNコード: 9780198309970
- 出版年: 2015
- 対象学年: 1年
- 対象者: 学習者
- 科目名: Computing
- 概要: コンピュータの使い方の基本(安全な利用方法、ペイント・Microsoft Excel・Webブラウザの使い方)、Scratchの紹介。

No.18

- タイトル: International Primary Computing 2
- ISBNコード: 9780198309987
- 出版年: 2015
- 対象学年: 2年
- 対象者: 学習者
- 科目名: Computing
- 概要: パソコンの基本的な操作方法、アイコンの意味の説明、インターネットの使い方、社会での情報技術の利用方法、Scratchでの簡単なプログラミングの紹介。

No.19

- タイトル: International Primary Computing 3
- ISBNコード: 9780198309994
- 出版年: 2015
- 対象学年: 3年
- 対象者: 学習者
- 科目名: Computing
- 概要: Microsoft PowerPoint、Excelの使い方、Scratchでのプログラミング(繰り返し処理)、インターネットを利用したコミュニケーション、画像情報表現の紹介。

No.20

- タイトル: International Primary Computing 4
- ISBNコード: 9780198310006
- 出版年: 2015
- 対象学年: 4年
- 対象者: 学習者
- 科目名: Computing
- 概要: Microsoft Word、Excelの使い方、画像の管理・編集方法、Scratchでのプログラミング(条件分岐)、インターネットを使った情報検索方法、著作権の解説。

No.21	<ul style="list-style-type: none"> <li>●タイトル: International Primary Computing 5</li> <li>●ISBNコード: 9780198310013</li> <li>●出版年: 2015</li> <li>●対象学年: 5年</li> <li>●対象者: 学習者</li> <li>●科目名: Computing</li> <li>●概要: Microsoft Word, Excelの使い方、マルチメディアの利用・編集方法、ブログの使い方、Scratchでのプログラミング(ゲーム作成)、ソートアルゴリズムの紹介。</li> </ul>
No.22	<ul style="list-style-type: none"> <li>●タイトル: International Primary Computing 6</li> <li>●ISBNコード: 9780198310020</li> <li>●出版年: 2015</li> <li>●対象学年: 6年</li> <li>●対象者: 学習者</li> <li>●科目名: Computing</li> <li>●概要: Microsoft Word, Excelの使い方、マルチメディアの利用(プレゼンテーション)、コンピュータゲーム設計、Scratchでのプログラミング(迷路作成)、Webページ作成の解説。</li> </ul>

「諸外国におけるプログラミング教育に関する調査研究」(文部科学省平成26年度・情報教育指導力向上支援事業) 報告書<sup>4)</sup>

イギリスでのコンピュータ教育(プログラミング教育を含む)に使われている教科書の例を表4に示した。この資料から、小学校で実際に行われているプログラミング教育の内容を学年ごとにまとめると次のようになる。

表5 小学校でのプログラミング教育事例

学年	内 容
1	Scratchの紹介
2	Scratchでの簡単なプログラミングの紹介
3	Scratchでのプログラミング(繰り返し処理)
4	Scratchでのプログラミング(条件分岐)
5	Scratchでのプログラミング(ゲーム作成)、ソートアルゴリズムの紹介。
6	コンピュータゲーム設計、Scratchでのプログラミング(迷路作成)

この内容は、日本の小学校でのプログラミング教育の教育内容に大いに参考になるだろう。

### 3. ビジュアルプログラミング言語の比較

どのようなプログラミングシステムがあり、どのようにプログラミング教育を行えば良いのかについて考える。

まずプログラミングシステムとしてビジュアルプログラミング言語を提唱する。ビジュアルプログラミング言語とはプログラムコードを記述するのではなく、視覚的な操作でプログラミングが可能なプログラミング言語である。代表的なものにScratch・MOONBlock・プログラミン・VISCUIT・Google・Bliclyなどがある。

#### 1) Scratch

開発者: MIT (Massachusetts Institute of Technology: マサチューセッツ工科大学) メディアラボが開発した。  
特 徴: 視覚表現でキャラクターを動かす直感的にプログラミングができる。世界で最も多く使われているビジュアルプログラミングシステム。初心者からかなりの上級者まで対応ができ、コミュニティが充実している。

#### 2) MOONBlock

開発者: 秋葉原リサーチセンターが開発した。  
特 徴: ブロックを組み合わせて並べるだけで直感的にプログラミングができる。ブロックを動かしながらコーディングの内容をチェックできるので、次のステップであるコーディングにも進みたい者にも対応している。

#### 3) プログラミン

開発者: 文部科学省が開発された。  
特 徴: 「絵を組み合わせる」ことでプログラムができる。命令の数が少なくできることが単純である分、初心者が表現を楽しみながらプログラミングができるようになっている。

#### 4) VISCUIT

開発者: NTTの研究で開発された、開発者は原田康徳氏。  
特 徴: 自分で描いた絵を動かしながらプログラミングを学ぶ形式。見たままのプログラミングができるので初心者向きである。

#### 5) Google・Blicly

開発者: Googleが提供されている。  
特 徴: ブロックを組み合わせることで、プログラムを作ることができる。プログラムになれるための問題集的な内容。

### 4. 図工を例とした教育を提唱

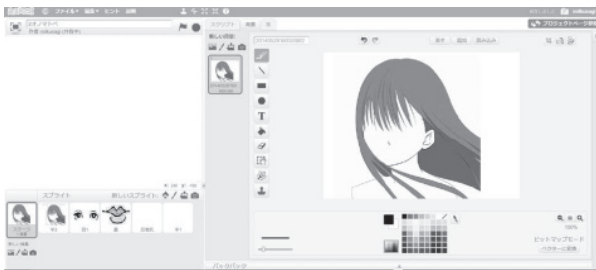
図工を例として児童にわかりやすく教える教育内容及び教育教材を提唱する。いくつかの例をここに示す。

例1) 自分自身のキャラクターを動かそう。

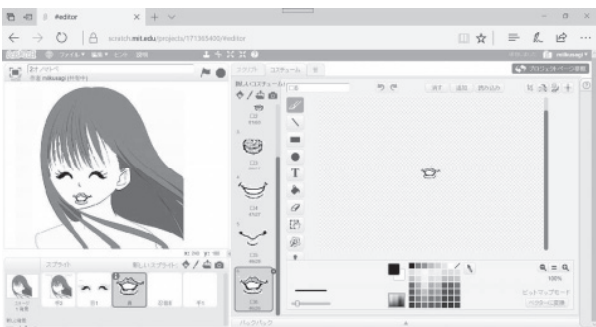
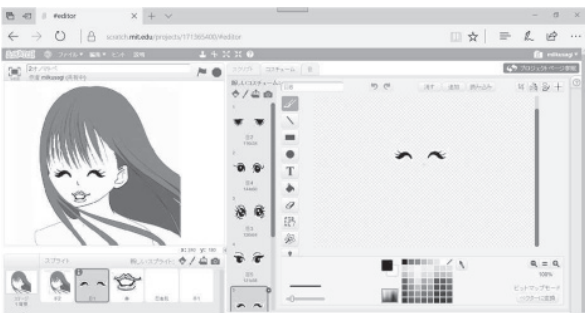
自分の似顔絵を描き日本語の様々な言葉（舞い上がる気持ち・がっかりした・頭が真っ白など）やオノマトペ（ワクワク・シクシク・ガタガタなど）で表現される状態を目鼻口などのパーツを使って個別に表現しその組み合わせで自分自身の感情の変化をキャラクターで表す。

形容詞・犠牲音を通してことばの表現を考える。以下、手続きを①から④まで示した。

①まずScratch上で顔を描く。



②同様にScratch上で目と口を幾つか描く。



③目と口を入れ替えるプログラミングを行なう。



### Scratchでプログラミングの詳細

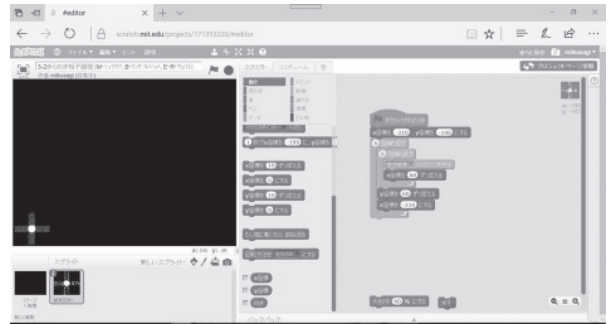
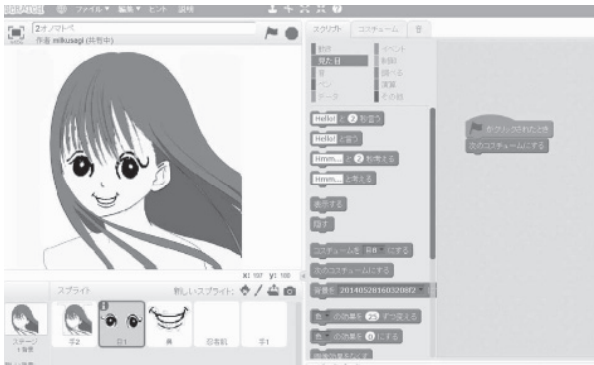


プログラミング詳細

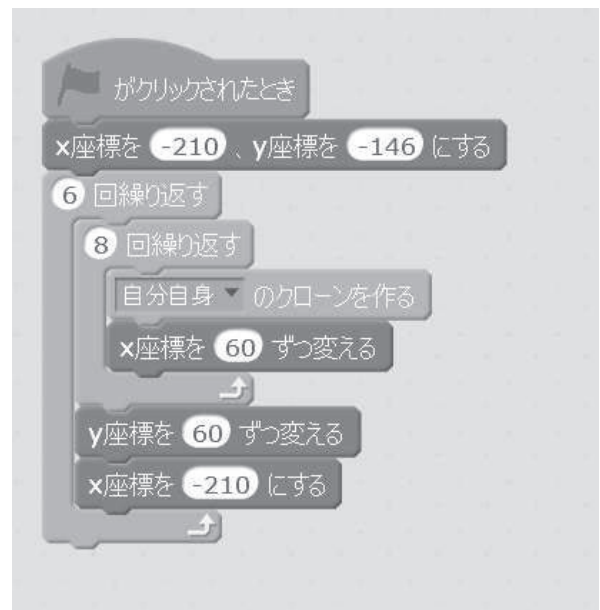
- i 「▶がクリックされたとき」に実行する。
- ii 用意した目や口を次々に入れ替えて表示する。「次のコスチュームにする」

④実行すると目と口が、次々に入れ替わる。





### Scratchでプログラミングの詳細



例2) 錯覚図形（トリックアート）を創ろう。

画像の白線の交差部分に黒色の点が現れたり消えたりして、まるで点滅しているように見えるイリュージョンを体験する。「きらめき格子錯視（M・シュラウフ、B・リンゲルバッハ、E・R・ウィスト）」以下、手続きを①から③まで示した。

①錯覚図形（トリックアート）の素材を描く。Scratchでも描けるが、今回はペイントを使用した。



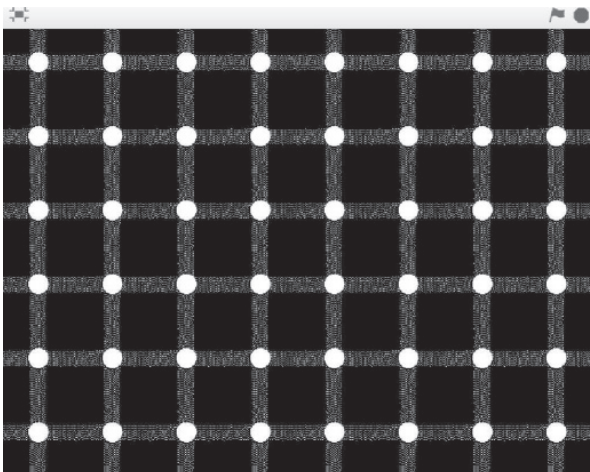
②描いた素材をScratchに取り込み、その素材を定期的に繰り返し表示するプログラミングをする。

- i 素材を左端に設置する。「x座標を-210、y座標を146にする」
- ii 素材をコピー「自分自身のクローンを作る」して表示し右へ60動かす。「x座標を60ずつ変える」。
- iii これを8回繰り返す。「8回繰り返す」（素材が横に8個作られる）
- IV 次に上へ60動かす「y座標を60ずつ変える」、左位置を元の位置に戻す。「x座標を-210にする」
- V さらに ii～IVを6回繰り返す「6回繰り返す」。横に8個の素材が縦に6段作られる）

③黒い点がいくつ見えるか数えてみよう。



完成画面を拡大



完成した画面を見つめていると、白い丸の中に黒い点が見れる。この目の錯覚は全てが同じ大きさ、同じ距離で並ばれている必要がある。手作業ではとても時間がかかり授業内で行えない内容がプログラミングでは可能となる。

例3) カラフルな図形を描いてみよう、出来上がった図形を使ったコマ廻し。<sup>注)</sup>

円や多角形など図形をカラフルな色彩で表現する。その図形をプリントアウトして手作りのコマに貼り付け、そのコマが回ること(視覚混合)によって自分がデザインした色彩図形の色が変化することを楽しみ、加算混合及び減算混合の違いを学ぶ。尚、加算混合及び減算混合を下記に示す。

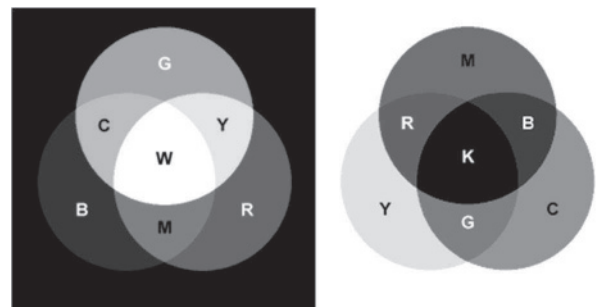
色の3原色加算混合(RBG)の混合

- Red (赤)とGreen (緑)との混合からYellow (イエロー)
- Green (緑)とBlue (青)との混合からCyan (シアン)
- Blue (青)とRed (赤)との混合からMagenta (マゼンタ)
- Red (赤)とGreen (緑)とBlue (青)との混合からWhite (白)

色の3原色減算混合(CYM)の混合

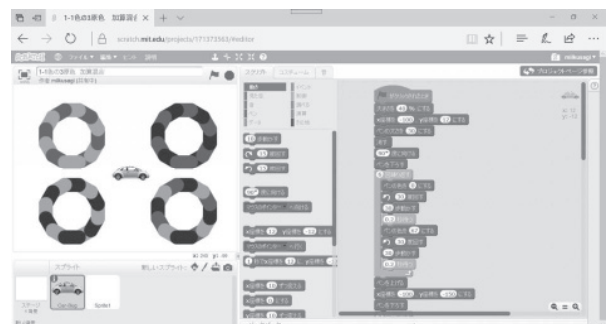
- Cyan (シアン)とYellow (イエロー)との混合からGreen (緑)
- Yellow (イエロー)とMagenta (マゼンタ)との混合からRed (赤)
- Magenta (マゼンタ)とCyan (シアン)との混合からBlue (青)
- Cyan (シアン)とYellow (イエロー)とMagenta (マゼンタ)との混合からKey (黒)

図6 加算混合(RBG)及び減算混合(CYM)



以下、手続きを①から⑤まで示した。

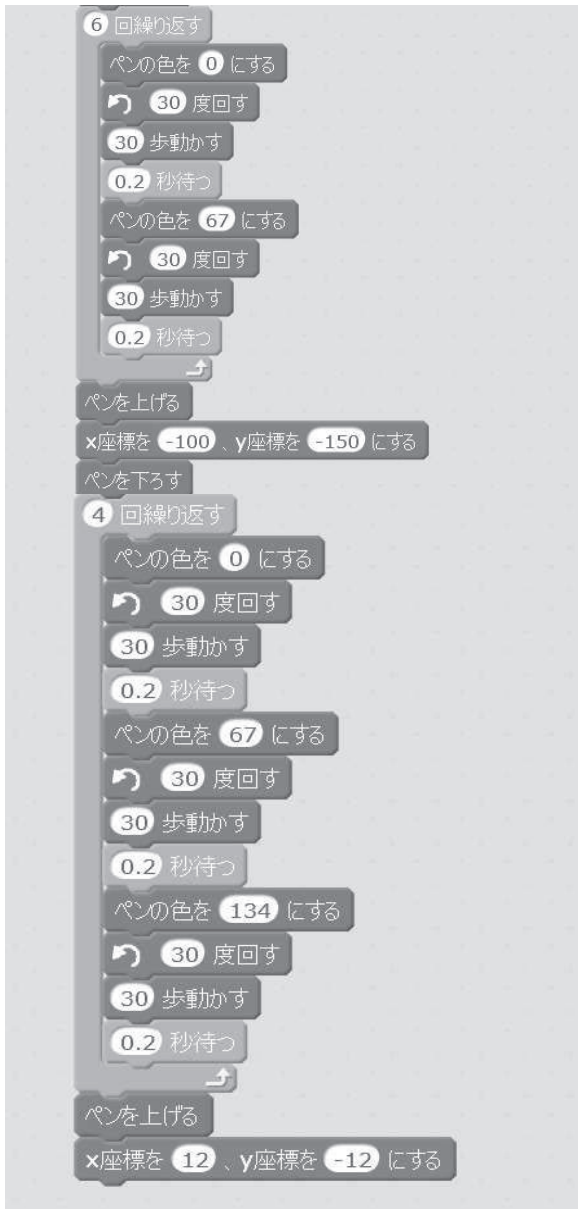
1) コマによる色の加算混合(RBG)を実験するための教材をScratchでプログラミングする。混合する色を輪にして描く。



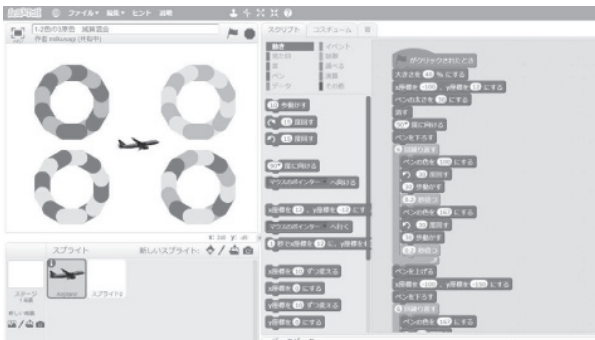
Scratchでプログラミングの詳細



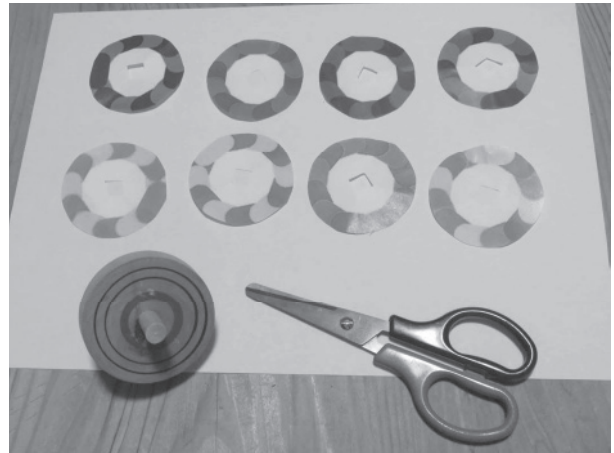




2) 同様にコマによる減算混合 (C Y M) を実験するための教材をScratchでプログラミングする。混合する色を輪にして描く。

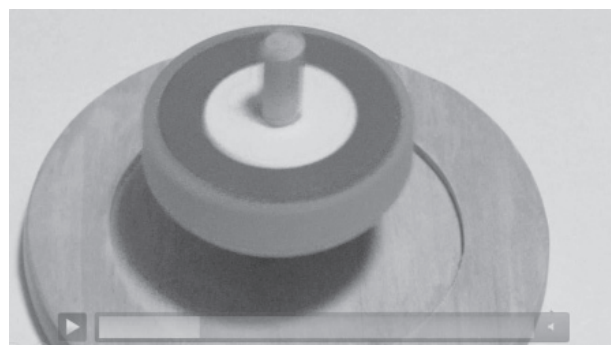
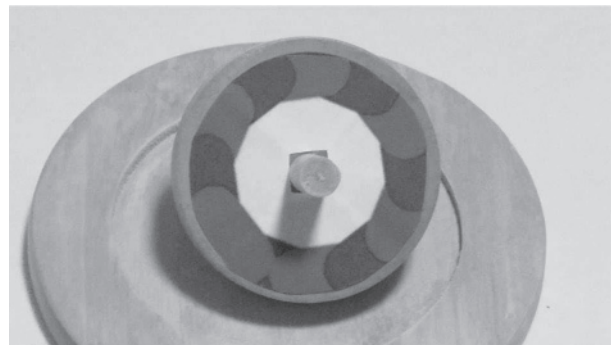


3) 描かれた輪を印刷して、ハサミで切り取る。



4) 切り取った描かれた輪をコマの上に乗せて、コマを廻す。

Blue (青)とRed (赤)との混合からMagenta (マゼンタ)



5) 同様にして合計8パターンのコマによる色の混合を行う。尚、本著はモノクロ印刷で製本されるため、残りの7パターンを含めた合計8パターンのコマによる混合カラー動画を次のURLに掲載した。

[http://ono.from.tv/academic\\_paper](http://ono.from.tv/academic_paper)

### Ⅲ. 結果

小学校新学習指導要領に明記されているように、既存の教科の中で、「プログラミングを体験しながら論理的思考力を身に付ける」<sup>1)</sup>を実践すると、「例1)自分自身のキャラクターを動かそう。」や「例2)錯覚図形(トリックアート)を創ろう。」にあるように、論理的思考力を身に付けながら手作業はできなかった新たな結果をプログラミングで得ることができる。また、「例3)カラフルな図形を描いてみよう、出来上がった図形を使ったコマ廻し」にあるように、プログラムの中で結果を出さずに、プログラミングを新しいツールとしてとらえる授業手法も自然体で良いだろう。

### Ⅳ. おわりに

読み書きを学ぶのは作家になるためではない、プログラミングを学ぶのはプログラマーになるためではない。生きていくために様々なことを可能にするために学ぶのである。

日本はかつて経験をしたことのない高齢化社会を迎える、また産業も変革期である。ITをどのように活用し価値あるものを創造するのか、そのことができる人材をいかに多く育てられるかが日本の将来にかかっている。日本の将来を担う児童たちがプログラミングを小学校で学ぶことは必要なことである。

この教育内容および教材をさらに発展させてe-ラーニング型デジタル教科書へと繋げていく所存である。

### 注

本著はモノクロ印刷で製本されるため、残りの5パターンを含めた合計6パターンのコマによる混合カラー動画を次のURLに掲載した。

[http://ono.from.tv/academic\\_paper](http://ono.from.tv/academic_paper)

2017年10月10日閲覧

### 引用文献

1) 小学校学習指導要領

[http://www.mext.go.jp/component/a\\_menu/education/micro\\_detail/\\_icsFiles/afieldfile/2017/05/12/1384661\\_4\\_2.pdf](http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2017/05/12/1384661_4_2.pdf) 文部科学省 平成29年3月

2) NRI 未来開発 NewsRelease

[http://www.nri.com/~media/PDF/jp/news/2015/151202\\_1.pdf](http://www.nri.com/~media/PDF/jp/news/2015/151202_1.pdf) 株式会社野村総合研究所 平成27年12月02日

3) 子ども・若者白書(旧青少年白書)特集 1 自己認識

[http://www8.cao.go.jp/youth/whitepaper/h26honpen/tokushu\\_02.html](http://www8.cao.go.jp/youth/whitepaper/h26honpen/tokushu_02.html) 内閣府 平成26年6月

4) 「諸外国におけるプログラミング教育に関する調査研究」(平成26年度・情報教育指導力向上支援事業)報告書

[http://jouhouka.mext.go.jp/school/pdf/programming\\_syogaikoku\\_houkokusyo.pdf](http://jouhouka.mext.go.jp/school/pdf/programming_syogaikoku_houkokusyo.pdf) 文部科学省 平成27年3月

### 参考文献

NRI 未来開発 NewsRelease

[http://www.nri.com/~media/PDF/jp/news/2015/151202\\_1.pdf](http://www.nri.com/~media/PDF/jp/news/2015/151202_1.pdf) 株式会社野村総合研究所 平成27年12月02日 教職員等の指導体制の在り方に関する懇談会提言

[http://www.mext.go.jp/a\\_menu/shotou/hensei/003/\\_icsFiles/afieldfile/2015/09/11/1361243\\_1.pdf](http://www.mext.go.jp/a_menu/shotou/hensei/003/_icsFiles/afieldfile/2015/09/11/1361243_1.pdf) 文部科学省 平成27年8月26日

子ども・若者白書(旧青少年白書)特集 1 自己認識  
[http://www8.cao.go.jp/youth/whitepaper/h26honpen/tokushu\\_02.html](http://www8.cao.go.jp/youth/whitepaper/h26honpen/tokushu_02.html) 内閣府 平成26年6月

「諸外国におけるプログラミング教育に関する調査研究」(平成26年度・情報教育指導力向上支援事業)報告書  
[http://jouhouka.mext.go.jp/school/pdf/programming\\_syogaikoku\\_houkokusyo.pdf](http://jouhouka.mext.go.jp/school/pdf/programming_syogaikoku_houkokusyo.pdf) 文部科学省 平成27年3月