

# Scratch によるプログラミング教育

## Programming Education Using Scratch

小林 大 輔\*

Daisuke Kobayashi

### Abstract:

As a programming environment, Scratch become popular. I report the result of introducing Scratch into the class. Compared to C and Java, students' understanding was improved.

### キーワード：

Scratch C、Java、プログラミング学習、ゲーム

### I. はじめに

これまで筆者は、短大でのプログラミング教育で C 言語と Java を利用していたが、文法や構文の学習とデバッグに多くの時間を取られ、本質的なプログラミング学習まで学生を導くことが難しかった。対策として主に子供向けのプログラミング教育環境として評価の高い Scratch を 2018 年度の授業から導入し、一定の成果が得られたので報告する。

### II. Scratch 導入の経緯

一般に、コンピュータプログラミングは英語をベースとした C 言語や Java などのプログラミング言語によりコードを記述（以下コーディング）し、コンパイラによりコードを機械語に翻訳し動作させる。プログラミングの初学者は、プログラミング言語の「文法」や「構文」と、機械語への「翻訳」の手続きを学んだ後に、プログラムによる問題解決を学ぶ。理系の学生であれば、数学的な問題を様々なアルゴリズムを用いた

プログラムで解くという問題に自然にかかわることになり、学習に適した例題も豊富にあるが、文系の、特に短大で初めてプログラミングを学ぶ学生に適した例題があまりない。「ゲームを作る」を目的とした各種入門書もあり、これまで筆者も学生向けに試したことはあるが、プログラミング言語の文法の学習から、動きのあるゲームを作るまでには時間がかかり、学生のモチベーションを維持するのが困難であった。

Scratch は MIT メディアラボで開発されたプログラミング学習のプロジェクトで、ブラウザ上で動作し、無償で提供されている。したがって、インターネットに接続できる PC があれば、誰でも利用できる<sup>1)</sup>。サイトの説明では「Scratch は、特に 8 歳から 16 歳向けにデザインされていますが、すべての年代の人々に使われています」とあり、子供でも手軽にプログラミングを始められることを謳っており、実際に小学生向けの入門書も書店に多数並んでいる<sup>2)</sup>。

2018 年度の夏のオープンキャンパスで高

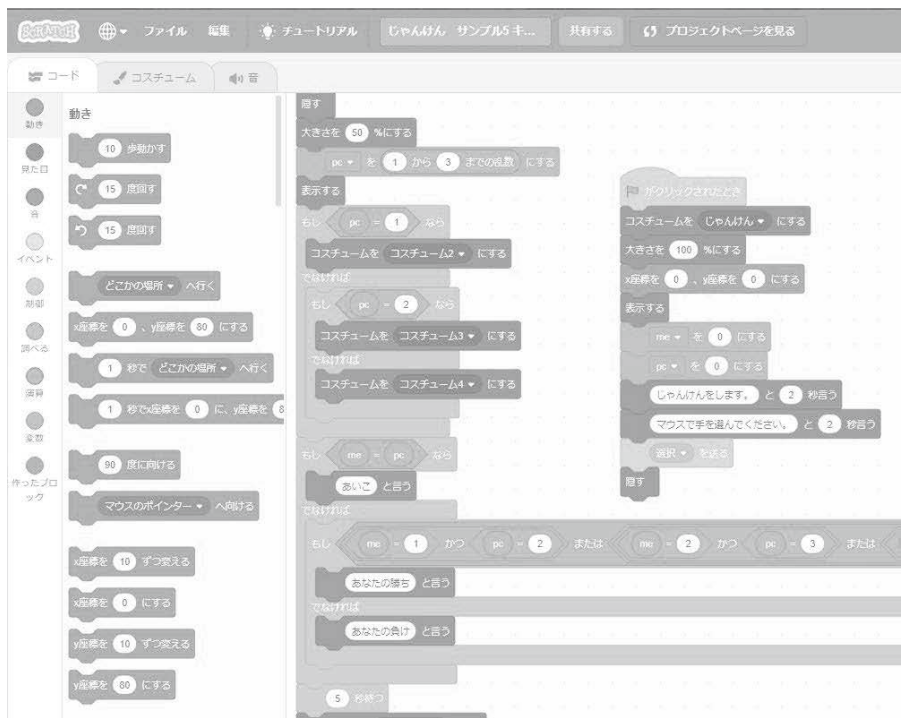
\*佐野日本大学短期大学 総合キャリア教育学科 Sano Nihon University College Professor

```

1 import java.util.Random;
2 import java.util.Scanner;
3
4 public class LuckyNo {
5
6     public static void main(String[] args) {
7
8         Random rand = new Random();
9         Scanner stdin = new Scanner(System.in);
10
11         int lucky = rand.nextInt(10);
12
13         System.out.print("入力してください 0:グー 1:チョキ 2:パー ");
14         int hand = stdin.nextInt();
15
16         System.out.println(); //改行
17         System.out.print("あなたは ");
18         switch (hand){ //0:グー,1:チョキ,2:パー それ以外は全部グー
19             case 0 : System.out.println("グー"); break;
20             case 1 : System.out.println("チョキ"); break;
21             case 2 : System.out.println("パー"); break;
22             default : System.out.println("グー");
23                 hand=0; break;
24         }
25

```

(a) Java コード



(b) Scratch のブロック

図1 Java コードと Scratch ブロック

校生及び保護者とプログラミングの体験学習をした結果、概ね好評であったため後期の Java プログラミングの授業で、予定していたシラバスから変更することを学生に了解を得たうえで Scratch によるプログラミング教育を実施した。

### III. Scratch の特徴

#### 1. 環境の整備が（ほぼ）不要

C 言語や Java のプログラムを開発するためには、使用する PC でプログラミング環境を整備する必要がある。例えば Java の場合、Java 開発キット（Java Development Kit、以下 JDK と略す）をインストールする必要がある。無料なので費用は掛からないが、アプリケーションの利用とは異なり、

相応の PC の知識が必要であり、かつインストールに半日以上の時間がかかる。また、JDK を利用して正しくコンパイルできるかの検証も初心者には難しく、Java プログラミングを履修する学生が自宅の PC に JDK をインストールするには敷居が高い。

Scratch は PC のインターネットブラウザ上で動作し、インストールも不要である。必要なのはインターネットへの接続とユーザー登録のみなので、ユーザー登録を授業で済ませてしまえば自宅で自習できる。Scratch2.0 までは FLASH を使用していたため、Windows 以外のタブレット端末では基本的に利用できなかった（一部の FLASH 対応のブラウザをインストールすれば利用可能）が、2019 年から Scratch3.0 ヘバージョ

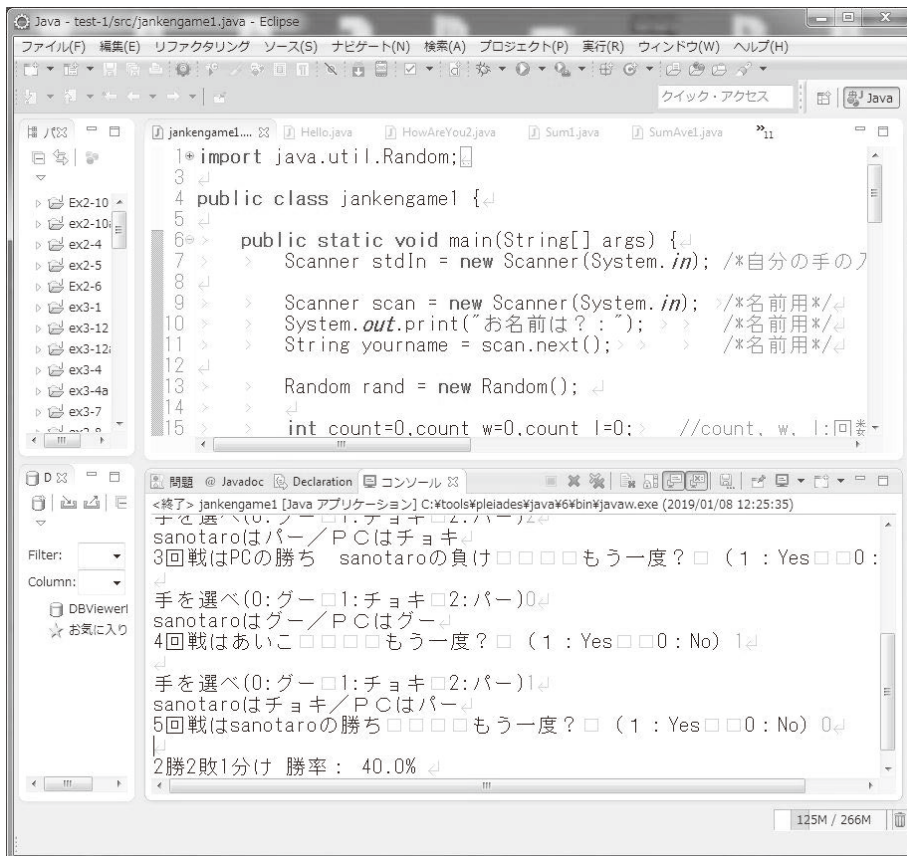


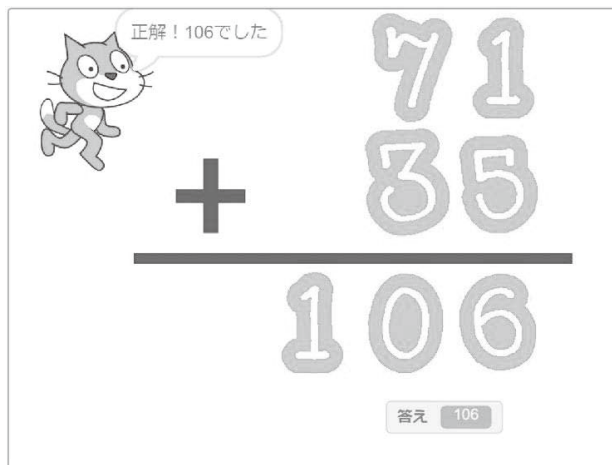
図2 じゃんけんゲーム (Java テキストベース)



(a) シューティングゲーム



(b) じゃんけんゲーム



(c) たし算引き算ゲーム

図3 授業で作成したプログラム

ンアップし、HTML5.0 対応となったため、PC で作成したプログラムをタブレット端末の Chrome でも利用できるようになった。

## 2. コーディングしない

Scratch ではコーディングの代わりにブロックの組み合わせでプログラムを記述する (図 1 (a)(b))。ブロックは一般的なプログラミングの文法を視覚的にわかりやすく表したもので、後述するとおり、文法ミスによるデバッグが原理的に発生しないため、効率の良い学習ができる。ブロックは日本語による表記が可能 (40 以上の言語に対応) で、英語を学んでいない小学生や英語に抵抗感のある大人でも利用できる。

## 3. ゲーム作成が基本

Scratch はキャラクターを動かすことから始まるので、視覚的に理解できる。簡単なシューティングゲームであれば 1 回目の 90 分の授業内で作成可能である。

# IV. 授業への導入結果

## 1. 授業の進捗

プログラム作成は (a) コーディング、(b) コンパイルに伴うデバッグ、(c) 実行、(d) 動作のデバッグを繰り返す。ここで (b) のコンパイルに伴うデバッグはコーディングのミス (多くは文法のミス) を訂正するもので、これを完了しないと機械語への「翻訳」ができず (c) の実行ができない。文法や構文のミスがあればコンパイラがエラーメッセージを出す、多くの場合エラーメッセージが英語のため、慣れるまではこのエラーメッセージすら理解できない。(b) のデバッグに時間がかかることが授業の進度低下と学生のモチベーションの低下を招いていたと考える。Ⅲ-2 で述べた通り、Scratch では (b) のコンパイルに伴うデバッグがないので、学生は視覚的に動作の不具合を見な

がら (d) の動作のデバッグができ、効率よくプログラム作成ができた。

## 2. 作成したプログラム

多くのテキストで採用しているプログラミング学習は、

- (1) 任意の文字の表示
- (2) 条件分岐
- (3) 繰り返し処理
- (4) 配列

といった順序で学習する。この後はオブジェクト指向に関する内容となりクラスやメソッドなど高度な内容となる。ここ数年、プログラミングの授業では (1) ~ (4) についてテキストの例題と練習問題をこなした後、学生が作るオリジナルのプログラムはキャラクターベースのじゃんけんゲーム (図 2) のみであった。

今年度の授業では、初回の授業で簡単なシューティングゲームを作成したのち、最終回までにじゃんけんゲーム、たし算ひき算学習用ゲームを作成できた (図 3 (a)(b)(c))。それぞれのゲームでは、基本的な動作を確認したのち、ユーザーインターフェイスを考慮して細かい改良を加えることもできた。

(1) ~ (3) は作成したすべてのゲームに必要な要素であるので、最初の 90 分で作成する簡単なシューティングゲームで一通りの説明が終わり、更に他のゲームを作るとに理解が深まっていった。(4) の配列や、様々なアルゴリズムの学習に充てる時間が持てなかったが、これらは 4 年制大学への編入用の科目での対応でよいと考える。

## V. おわりに

文系の短大生が初めて学ぶプログラミングとして、Scratch による学習が有効であるとの感触を得た。授業以外にも、子供向け・社会人向け講座も実施してみたい。

注

- 1) <https://scratch.mit.edu/about>
- 2) 例えば 竹林ら, ”できるキッズ 子どもと学ぶ Scratch プログラミング入門”, 2017