

Scratchで遊んでわかる！ 中学数学 数学をプログラミングでハックする

五十嵐康伸 博士(理学)
滋賀大学 データサイエンス・AIイノベーション研究推進センター
名古屋工業大学 コミュニティ創成教育研究センター
yasunobu.igarashi@gmail.com

岡田 延昭
光が丘オープンソースクラブ
nokada07@gmail.com

1 はじめに(民間の活動)

対面型ワークショップにおける
探究学習の授業実践事例



ICTを活用した
中学数学の体系的な教材

Scratchで遊んでわかる! 岡田 延昭、五十嵐 康伸 著

中 + 学 + 数 学

数学を
プログラミングで
ハックする



発表内容

1 はじめに

2 授業実践事例

2.1 対面型ワークショップの内容

2.2 対面型ワークショップの運営

3 教育教材

3.1 書籍執筆の経緯

3.2 書籍とワークショップの対応

4 おわりに

2.1 対面型ワークショップの内容

- 都立高校入試の数学問題
 - 中学校で学習する基本的な理解を問う問題
 - 紙と鉛筆で解くことを想定
 - 奇跡的にScratchのステージにぴったりおさまる問題設定

2.1 対面型ワークショップの内容

- 都立高校入試の数学問題
 - 紙と鉛筆を用いて観る数学の一面
 - プログラミングを用いて観る数学の一面

2.1 対面型ワークショップの内容

- 都立高校入試の数学問題
 - 紙と鉛筆を用いて観る数学の一面
 - 象徴的な入試問題
 - プログラミングを用いて観る数学の一面
 - 学校で学んだ公式や数式を把握しなおし、遊んでみる
- 都立高校入試ハック(hack)
 - ある対象を別の角度から見て、従来とは違った方法で遊んだり面白がったりする、客観的で喜劇的な行為
 - 紙と鉛筆が中心の試験をちょっと遠くから俯瞰し、相対化する

1 **ブロックパレット**：Scratchで使える命令が集められています。動き  や見た目  などのアイコンをクリックすると、それに関する命令のブロックが表示されます。

● 画面の説明

The screenshot shows the Scratch web interface with several callouts:

- 1**: Points to the **ブロックパレット** (Block Palette) on the left side of the screen.
- 2**: Points to the **バックパク** (Undo) button at the bottom center.
- 3**: Points to the **持ってくる** (Pick up) button, which is a blue block with a hand icon, located in the workspace.
- 4**: Points to the **スプライト** (Sprite) panel at the bottom right, which shows the selected sprite (Scratch Cat) and its properties.
- 5**: Points to the **ステージ** (Stage) panel at the bottom right, which shows the current scene and background.

Additional annotations in the workspace include:

- 捨てる** (Drop): A red arrow pointing from the 'どこかの場所へ行く' block to the '捨てる' button.
- 持ってくる** (Pick up): A red arrow pointing from the '捨てる' button to the 'どこかの場所へ行く' block.

2 コードエリア：ここにブロックパレットからブロックをドラッグ&ドロップして、命令を組み立てていきます（反対にブロックを削除したいときは**2**から**1**にドラッグ&ドロップします）。

●画面の説明

The screenshot shows the Scratch code editor interface with several numbered callouts:

- 1**: Points to the left sidebar containing the block palette, categorized by function (Motion, Look, Sound, Events, Control, Cycles, Variables, Pointers, Block Definition).
- 2**: Points to the central workspace where code blocks are assembled. A red box highlights a 'Go to some location and say for 2 sec' block. Red arrows labeled '持ってくる' (bring) and '捨てる' (discard) show the drag-and-drop interaction between the block palette and the workspace.
- 3**: Points to the top navigation bar, which includes options like '共有する' (Share), 'プロジェクトページを見る' (View project page), 'チュートリアル' (Tutorial), and '直ちに保存' (Save immediately).
- 4**: Points to the 'スプライト' (Sprite) panel at the bottom right, which shows the current sprite (Scratch Cat) and its properties like position (x=0, y=0), size (100), and direction (90).
- 5**: Points to the 'ステージ' (Stage) panel at the bottom right, which shows the current scene (Background 1).

Additional interface elements include the 'コード' (Code) tab, 'バックバック' (Undo) button at the bottom, and the Scratch logo in the top right corner.

3 ステージ：命令された結果が表示されます。真ん中にあるネコはスプライトと呼ばれます。ステージに立っている役者と考えるといいかもしれません。役目を与えられたスプライト（役者）にコード（台本）を書いてあげて、表現したいもの（舞台作品）をステージ上に実現します。

The image shows the Scratch 3.0 interface with several numbered annotations:

- 1**: Points to the block palette on the left side of the interface.
- 2**: Points to the 'バックパク' (Backspace) key at the bottom center.
- 3**: Points to a 'どこかの場所へ行く' (Go to some location and say for 2 secs) block in the script area.
- 4**: Points to the 'スプライト' (Sprite) area at the bottom right, which contains a cat sprite.
- 5**: Points to the 'ステージ' (Stage) area at the bottom right, which is currently empty.

Red arrows indicate the movement of the block:

- An arrow labeled **持ってくる** (bring) points from the block in the script area to the stage area.
- An arrow labeled **捨てる** (discard) points from the block in the stage area back to the script area.

4 スプライトエリア：どのようなスプライトがいるのか、また、そのスプライト情報（プロパティ）が表示されています。例えば、名前や座標、大きさ、向きの情報などが示されています。

● 画面の説明

1

2

3

4

5

バックバック

持ってくる

捨てる

スプライト

スプライト1

表示する

大きさ

向き

90

100

90

ステージ

背景

1

5 背景エリア：背景の情報が示されています。舞台に例えるなら、舞台装置に関する変更などを設定する場所と考えるとよいかもしれません。

● 画面の説明

1

2

3

4

5

バックパク

持ってくる

捨てる

2.1 対面型ワークショップの内容

- 数学の動的モデルをScratchで表現
 - 動かせるグラフ
 - 動かせる図形

発表内容

1 はじめに

2 授業実践事例

2.1 対面型ワークショップの内容

2.2 対面型ワークショップの運営

3 教育教材

3.1 書籍執筆の経緯

3.2 書籍とワークショップの対応

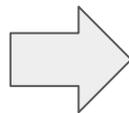
4 おわりに

2.2 対面型ワークショップの運営(場所)



2.2 対面型ワークショップの運営

施設: 光が丘図書館(洋室)
プロジェクターの投影先: スクリーン



光が丘公民館(和室)
壁



2.2 対面型ワークショップの運営

2016年12月

2018年10月

2020年2月



対面型ワークショップ(3年2ヶ月)

2.2 対面型ワークショップの運営

合計回数：約50回

頻度：月1回→週2回

曜日と時間：水19:00-21:00と日19:00-21:00

参加者：のべ100人

PC：基本参加者が持参、予備のchrome bookを2台と運営用のPC1台を運営が持参

2.2 対面型ワークショップの運営

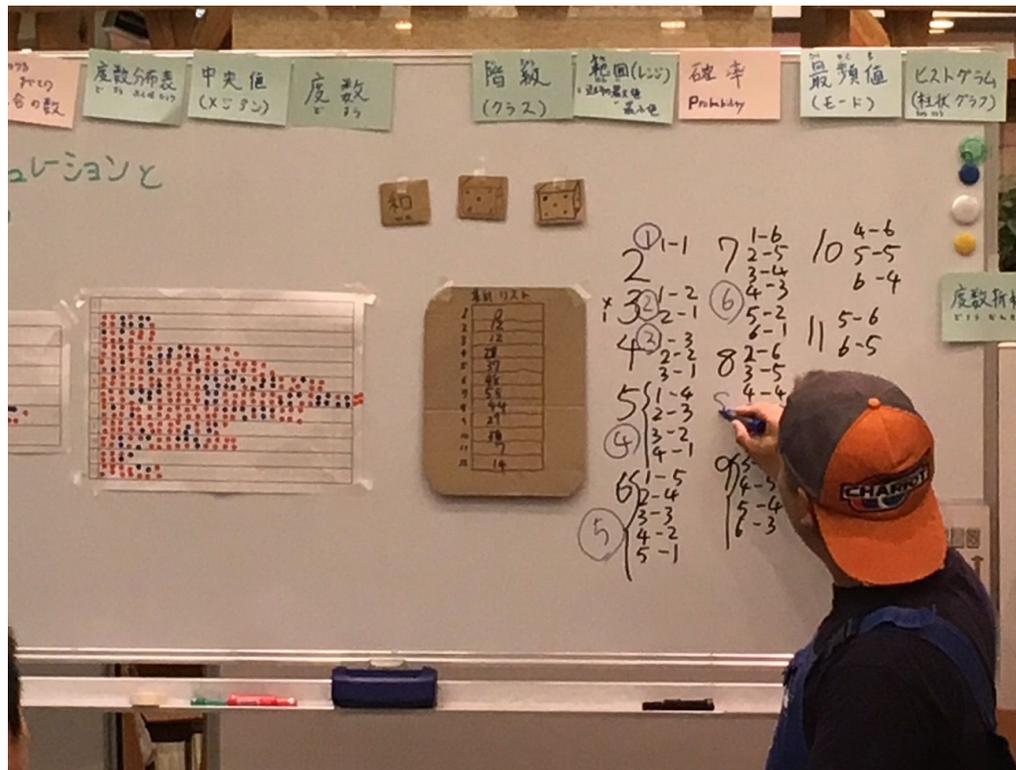
参加者の属性

当初の想定：公立中学生、光が丘第3中学校

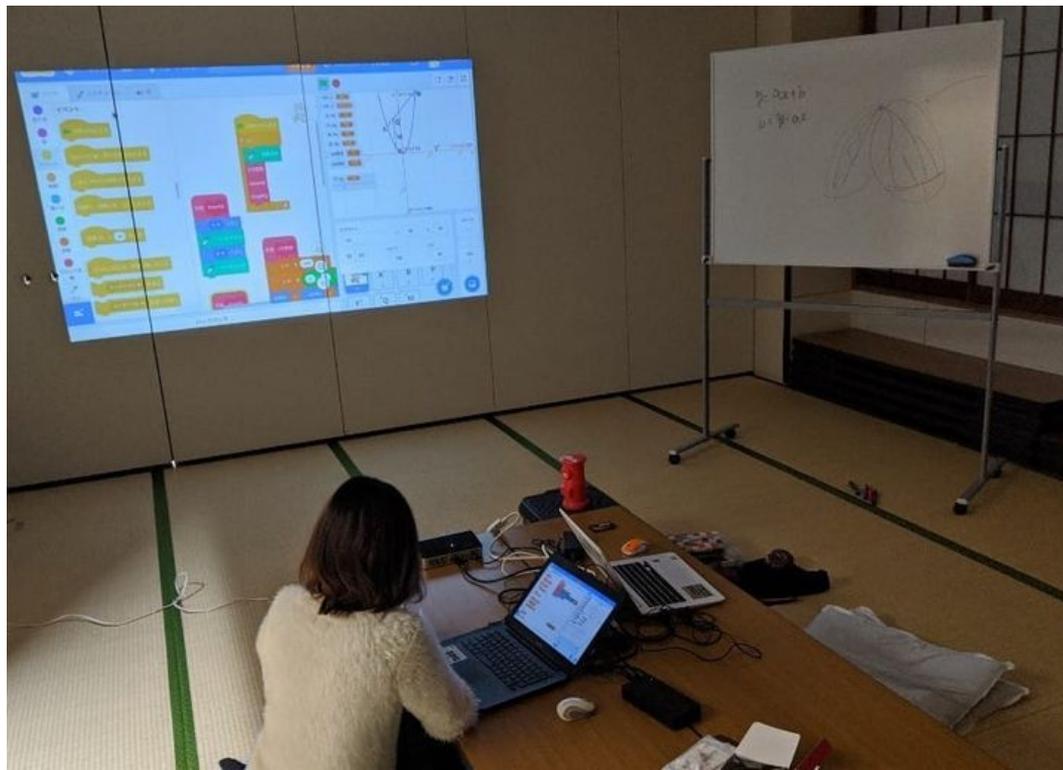
興味を持って継続して参加：私立中学生、城北中学校

その他：小学生や塾の先生

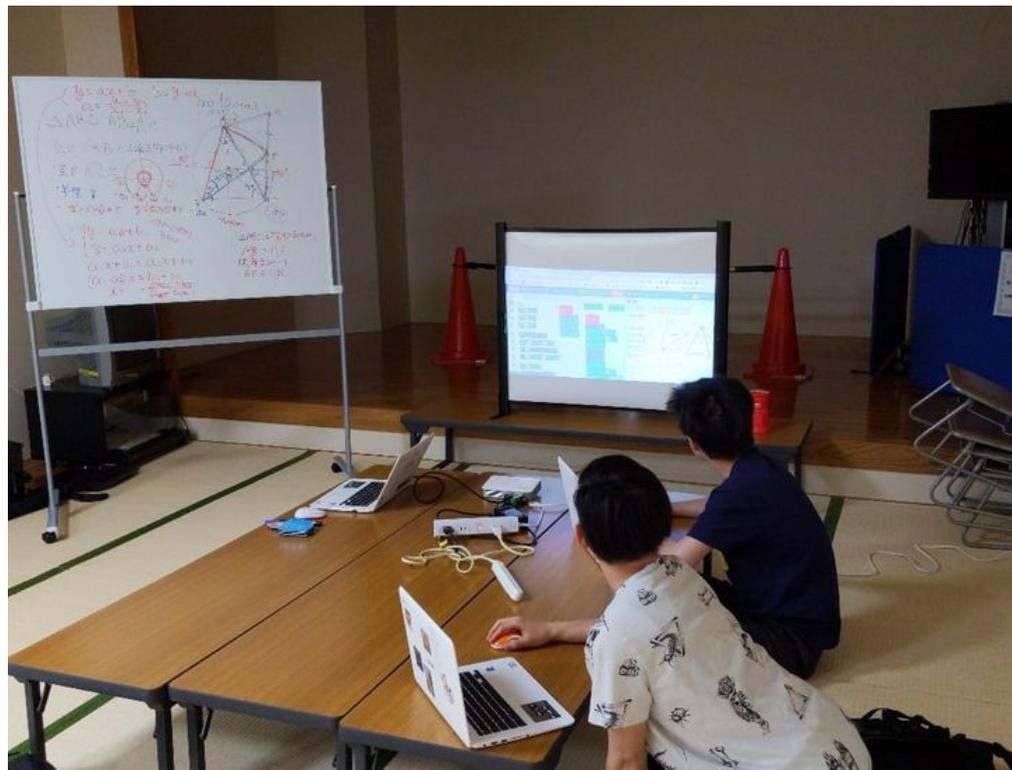
統計の問題のワークショップ風景



公民館における平成30年度の都立高校入試数学の設問3(2次関数)のワークショップ風景



公民館における平成27年度都立高校数学入試問題の問4(円周上の点を結んでできる図形)のワークショップ風景



発表内容

1 はじめに

2 授業実践事例

2.1 対面型ワークショップの内容

2.2 対面型ワークショップの運営

3 教育教材

3.1 書籍執筆の経緯

3.2 書籍とワークショップの対応

4 おわりに

コードの出力結果

twitterにて #都立高校入試ハックのハッシュタグをつけて発信

光が丘オープンソースクラブ @hkr_osc

これは、都立高校入試でもよく出題される2つのサイコロの和の頻度分布のシミュレーション。実際にサイコロを振って観察したあと、参加者で意見を出しながら、Scratchでモデル構築しました。リストをソートアルゴリズムで並べ替え、中央値や最頻値、標準偏差まで扱えます。

サイコロ	和
dice1	1
dice2	5
goukei	6
trial_num	1000
i	13
j	1
temp	165
total	5506
mode	7
median	85
average_t	5.506
average_d	90.909091

午後8:46 · 2020年1月21日

光が丘オープンソースクラブ @hkr_osc

モンテカルロ法による円周率の計算onScratch. プログラミング学習のよくある課題ですが、Scratchだと動的な可視化が低い労力でできるのでいいですね。近似値にちょっとずれがあるのは、Scratchの乱数発生に偏りがあるのかしら。

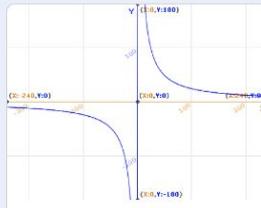
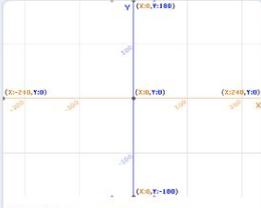
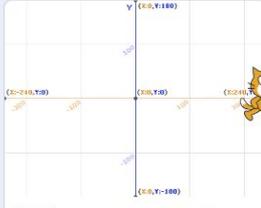
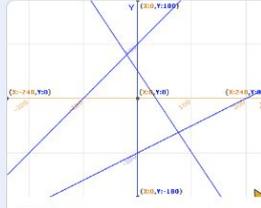
変数	値
trial_num	300000
pi	3.10988
x	93
y	-90
d	129.417928
cnt	233241
p	0.77747

モンテカルロ法による円周率計算

午後4:15 · 2020年12月23日

書籍中で使われたコード(28例)

scratchのプロジェクトとして公開 <https://scratch.mit.edu/studios/29433278>

 <p>1-1-1_準備編_Scratchの基本... nokada07</p>	 <p>1-1-2_準備編_数式を組み立... nokada07</p>	 <p>1-1-3_準備編_グラフ描画に... nokada07</p>
 <p>2-1-1_関数_線対称と点対称... nokada07</p>	 <p>2-1-2_関数_比例・反比例の... nokada07</p>	 <p>2-1-3_関数_一次関数を描く nokada07</p>
 <p>2-1-4_関数_2点間の傾き、切... nokada07</p>	 <p>2-1-5_関数_2点間を線分比... nokada07</p>	 <p>2-1-6_関数_垂直に交わる直... nokada07</p>

発表内容

1 はじめに

2 授業実践事例

2.1 対面型ワークショップの内容

2.2 対面型ワークショップの運営

3 教育教材

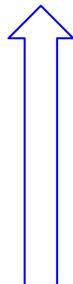
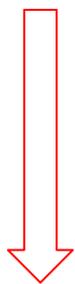
3.1 書籍執筆の経緯

3.2 書籍とワークショップの対応

4 おわりに

3.2 書籍とワークショップの対応(順番)

書籍の読者に向けた説明順



- PART 1 準備編: Scratchでのグラフの描き方、数式の表し方の基本
- PART 2 トレーニング編: 中学数学で習った内容をScratchで表現しよう
- PART 3 応用編: 都立高校入試問題をScratchでプログラムしてみよう
- エピローグ 数式を使った表現で遊ぶ

ワークショップにおけるコードと説明の開発順

3.2 書籍とワークショップの対応(目次)

PART 1 準備編: Scratchでのグラフの描き方、数式の表し方の基本

1-1-1 Scratchの基本的な操作

1-1-2 数式を組み立てる

1-1-3 グラフ描画における計算値の補正

3.2 書籍とワークショップの対応(目次)

PART 2 トレーニング編: 中学数学で習った内容をScratchで表現しよう

2-1 関数

2-1-1 線対称と点対称の図形を描く

2-1-2 比例・反比例のグラフを描く

2-1-3 一次関数を描く

2-1-4 2点間の傾き、切片、距離を求める

2-1-5 2点間を線分比で分ける

2-1-6 垂直に交わる直線を描く

2-1-7 2直線の交点を求める

2-1-8 二次関数を描く

2-1-9 点 P を動かす

2-1-10 解の公式・判別式を表現する

平面図形の項に入る前に

3.2 書籍とワークショップの対応(目次)

2-2 平面図形

2-2-1 円を描く

2-2-2 三角、四角、 n 角形を描く

2-2-3 扇型を描く

2-2-4 角度と面積を求める

2-2-5 円周角の定理を動かす

2-2-6 平行四辺形を動かす

2-3 確率

2-3-1 サイコロをシミュレーションする

3.2 書籍とワークショップの対応(目次)

PART 3 応用編: 都立高校入試問題をScratchでプログラムしてみよう

3-1-1 関数問題1

3-1-2 関数問題2

3-2-1 平面図形問題 1

3-2-2 平面図形問題 2

3.2 書籍とワークショップの対応(目次)

エピローグ 数式を使った表現で遊ぶ

点 P のパレード/原点 O の扇ダンス/点 P の紙吹雪

おわりに

3.2 書籍とワークショップの対応（イメージ）

ワークショップを登山にたとえて、「ふだん学校で習うのとは違う側面から山の頂上(≡解)を目指します。みなさんが登りやすいと思われるルートやステップで私は案内しますが、Scratchに慣れている人は自分がやりやすい歩幅や登り方があるかもしれません。また、ちょっと寄り道したところに発見があるかもしれません。全く別のルートを思いついたら、ぜひ挑戦してみてください。

いっしょに、
数学の冒険に出かけよう!



私がみなさんの
案内役をつとめますよ



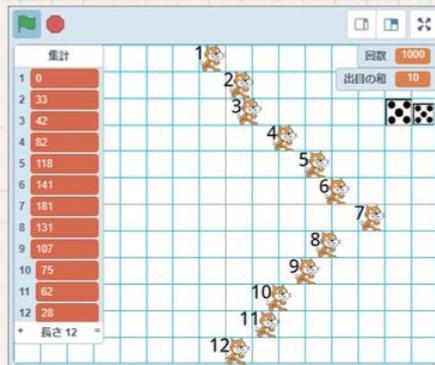
統計の問題

問 1 から 6 までの目の出る大小 1 つずつのさいころを同時に 1 回投げるとき、出る目の数の和が 10 以下になる確率を求めよ。
ただし、大小 2 つのさいころはともに、1 から 6 までのどの目が出ることも同様に確からしいものとする。

2-3-1

サイコロをシミュレーションする

大小2つのサイコロを振った出目の和を集計するシミュレーションを作ります。また、出目の和ごとの組み合わせの数も集計し、シミュレーションでの集計結果と比較します。シミュレーションの過程がわかりやすくなるよう、可視化にもチャレンジします。



サイコロを表現してみますよ。

乱数ブロックでサイコロを表現できそう。出目の和の出現回数をカウントする必要もありますね。



リストを使うと、リストの番号も利用できて便利です。

出目の和ごとにサイコロの組み合わせの数も集計して、シミュレーションの結果と比較してみましょう。



サイコロや集計結果を可視化してみてもいいかな？

出目の和の最大値は12なので、12個のスプライトを作って、集計の数値をx座標に反映させればいいですね。



✓ 作戦決定!

- 乱数ブロックを使って大小2個のサイコロを表現し、出目の和を計算する。
- 出目の和の出現回数を集計リストにカウントアップする。
- 2つのサイコロの組み合わせを生成する処理を作る。
- 出目の和ごとにリストに集計して各出目の場合の数とその総数を求める。
- サイコロを可視化する。
- 集計リストを参照して座標が変化するクローン(複製)を作り、可視化する。

動き

- 10 歩動かす
- 15 度回す
- 15 度回す
- どこかの場所へ行く
- x座標を 0、y座標を 0 にする
- 1 秒で どこかの場所へ行く
- 1 秒で x座標を 0 に、y座標を 0 にする
- 90 度に向ける
- マウスのポインターへ向ける
- x座標を 10 ずつ変える
- x座標を 0 にする
- y座標を 10 ずつ変える
- y座標を 0 にする
- もし隣に着いたら、跳ね返る
- 方法を 左右のみ にする
- x座標

```

定義 扇形を描く 半径 原点x座標 原点y座標 開始角 中心角
  θ を 開始角 にする
  x座標を 原点x座標、y座標を 原点y座標 にする
  ペンを下ろす
  x座標を 半径 * θ の sin + 原点x座標、y座標を 半径 * θ の cos + 原点y座標 にする
  θ > 開始角 + 中心角 まで繰り返す
  x座標を 半径 * θ の sin + 原点x座標、y座標を 半径 * θ の cos + 原点y座標 にする
  θ を 1 ずつ変える
  x座標を 原点x座標、y座標を 原点y座標 にする
  ペンを上げる
  
```



スプライト

draw x: -120 y: -6

表示する 大きさ 向き

100 90

draw O P P2 P3

発表内容

1 はじめに

2 授業実践事例

2.1 対面型ワークショップの内容

2.2 対面型ワークショップの運営

3 教育教材

3.1 書籍執筆の経緯

3.2 書籍とワークショップの対応

4 おわりに

4 おわりに: 著者らへのワークショップ希望(例: 夏休み都立高校入試ハック)はメールで相談

対面型ワークショップにおける 探究学習の授業実践事例



ICT を活用した 中学数学の体系的な教材

Scratchで遊んでわかる! 岡田 延昭、五十嵐 康伸 著

中 + 学 + 数 学

数学を
プログラミングで
ハックする



参考文献

1. <https://www.facebook.com/hkrosc/>
2. https://twitter.com/hkr_osc
3. <https://twitter.com/hika3cyu>
4. https://www.kyoiku.metro.tokyo.lg.jp/admission/high_school/ability_test/problem_and_answer/index.html
5. <https://scratch.mit.edu/studios/29433278/>
6. <http://www.plaything.jp/?p=6891>
7. <https://www.oreilly.co.jp/books/9784814400355/>
8. https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1416408.htm
9. https://www.mext.go.jp/content/20200403-mxt_jogai01-000006333_001.pdf