

Scratch で学ぶプログラミング

伊藤泰子

NDEC Web Magazine 3

2023.2.10

視覚的にプログラミングを学べる開発環境である Scratch を活用したプログラミングの実際について解説しています。マサチューセッツ工科大学メディアラボのライフロンギンダーガーデングループで開発されたもので、ブロックを組み合わせるプログラミングによって、キャラクターなどを思うように操ることができるように構成しました。

京都ノートルダム女子大学教育センター

<https://www.notredame.ac.jp/ndec/index.html>

e-mail to: vsadmin(a)ml.notredame.ac.jp

Scratch(スクラッチ)

マサチューセッツ工科大学メディアラボのライフロンギンダーガーデングループで開発された視覚的にプログラミングを学べる開発環境です。

ブロックを組み合わせて(=プログラミング)、キャラクターなどを思うように操ることができます。

Scratch – Imagine, Program, Share

<https://scratch.mit.edu/>

Japanese Scratch-Wiki

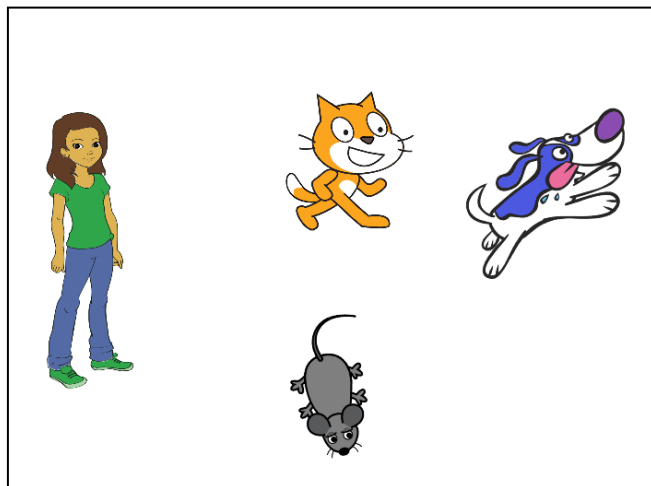
<https://ja.scratch-wiki.info/wiki/%E3%83%A1%E3%82%A4%E3%83%B3%E3%83%9A%E3%83%BC%E3%82%B8>

プロジェクト	Scratch の作品。
ステージ	プロジェクトの背景の部分。すべてのスプライトはステージの上に存在する。
スプライト	プロジェクト内で操作ができるオブジェクト(ねこ、りんご、ピアノなど)。
ブロック	プログラムを組み立てるために使う部品。つなげて使用する。複数のブロックを組み合わせてひとまとめにしたものを、スクリプトという。
ブロックパレット	使用できるブロックの一覧。 カテゴリ:動き、見た目、音、ペン、変数、リスト、イベント、制御、調べる、演算
スクリプト	スプライト、ステージに対する操作。ブロックの組み合わせ。
スクリプトエリア	スクリプトを記述するための場所。
スプライトリスト	プロジェクトで使用するスプライトの一覧。

演習1

「動き」のブロックを使ってスプライトを動かしてみましよう。

ファイル名: Exe1



■ Cat スプライト

1. 左上から動かす。

「動き」→「○歩動かす」

「動き」→「右回り○度回す」

200 歩、90 度、200 歩、90 度、200 歩、90 度、200 歩、90 度

※全部つなげると速くてわからない

2. 動きごとに 1 秒待つ。

「制御」→「1 秒待つ」

3. ずっと動かす。

作成したブロックを、「制御」→「ずっと」で囲む。

4. スタート設定

「イベント」→「スタートがクリックされたとき」を先頭につける。

5. プログラムスタート時の設定(=初期化)

「動き」→「90 度に向ける」

「動き」→「x 座標を -100 y 座標を 100 にする」

Cat のスクリプト



The image shows a Scratch script for a character named 'Cat'. The script is set against a light gray grid background. In the top right corner, there is a small cartoon cat icon. The script begins with a yellow 'When green flag clicked' block. This is followed by three blue blocks: 'Turn 90 degrees', 'Set x coordinate to -100 and y coordinate to 100', and a large orange 'Forever' loop block. Inside the loop, there are four blocks: 'Wait 1 second', 'Move 200 steps', 'Wait 1 second', and 'Turn 90 degrees'. The loop block has a small arrow at the bottom right, indicating it repeats indefinitely.

```
当緑旗が押されたとき  
90 度に向ける  
x座標を -100 、y座標を 100 にする  
ずっと  
1 秒待つ  
200 歩動かす  
1 秒待つ  
90 度回す
```

■Mouse1 スプライト

1. スプライトの追加

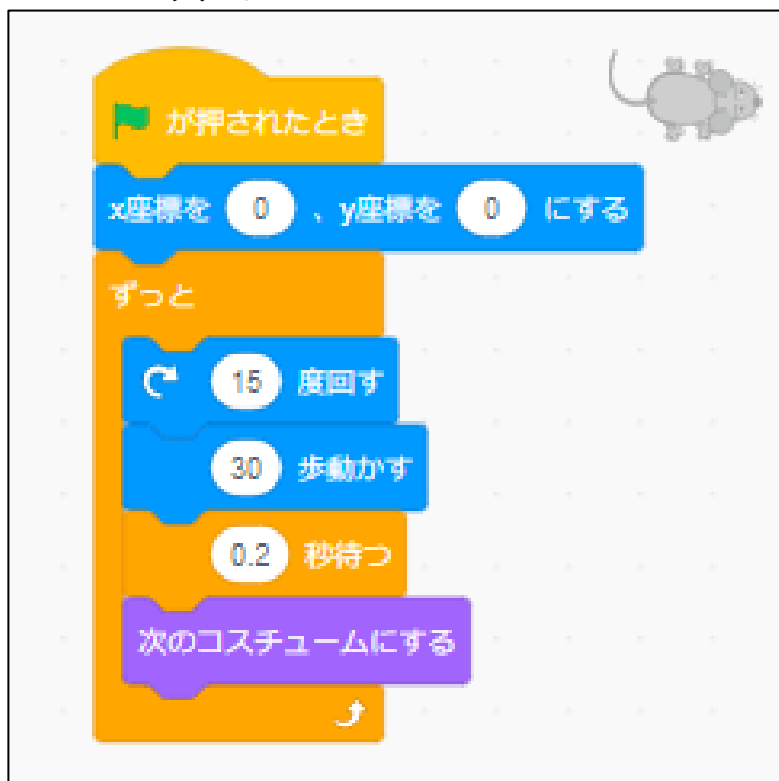


スプライトリストの「スプライトを選ぶ」をクリック

Mouse1 をクリック

2. Mouse1 をぐるぐる回らせる。

Mouse1 のスクリプト



■Dog1 スプライト

1. Dog1 スプライトの追加

2. 端から端まで走らせる。

Dog1 のスクリプト



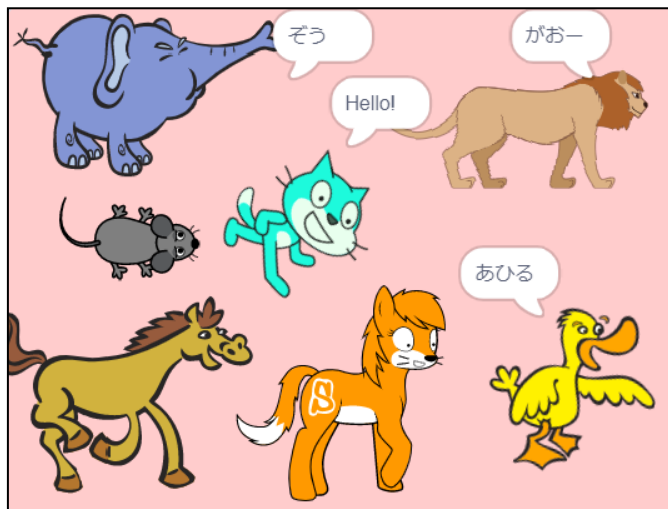
The image shows a Scratch script for a dog sprite. The script starts with a 'when green flag is clicked' event block. This is followed by a 'set x coordinate to -240, y coordinate to 20' block. Then, a 'turn 45 degrees' block. A 'forever' loop block contains the following actions: 'move 20 steps', 'wait 0.1 seconds', 'change to next costume', and 'if reached edge, jump back'. A small dog icon is visible in the top right corner of the script area.

■好きなスプライトを追加して動かしてみよう。

演習2

「見た目」のブロックを使って sprites の見た目を
変えてみましょう。

ファイル名: Exe2



■ Cat スプライト

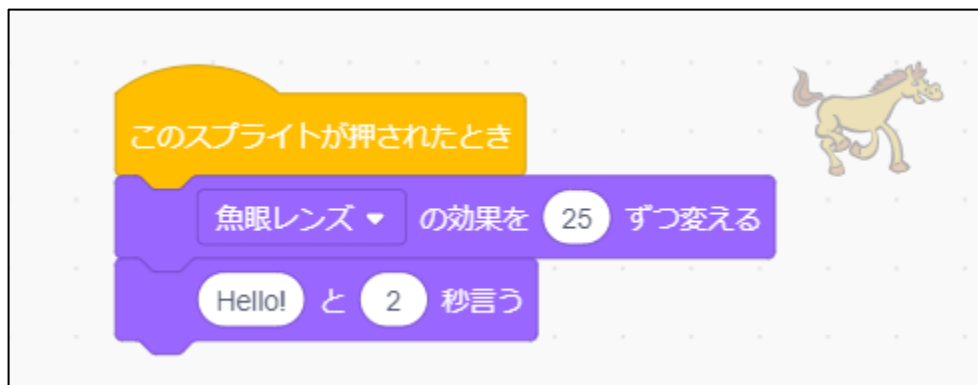
スタートでリセット、クリックで見た目を
変えるようにします。



■ Elephant スプライト



■Horse スプライト



■Lion スプライト



■Duck スプライト



※「明るさ」を「幽霊」にしてみましょう

■Mouse1 スプライト



※各スプライトに音のブロックを使って、サウンド効果を入れてみましょう。

Scratch のペイント機能を使う

ペイントエディター (3.0) - Japanese Scratch-Wiki

[https://ja.scratch-wiki.info/wiki/%E3%83%9A%E3%82%A4%E3%83%B3%E3%83%88%E3%82%A8%E3%83%87%E3%82%A3%E3%82%BF%E3%83%BC_\(3.0\)](https://ja.scratch-wiki.info/wiki/%E3%83%9A%E3%82%A4%E3%83%B3%E3%83%88%E3%82%A8%E3%83%87%E3%82%A3%E3%82%BF%E3%83%BC_(3.0))

コンピュータグラフィックスにおける画像の表現形式

ベクター形式

- ・画像情報を数式などの形で保存
- ・拡大・縮小しても画質が損なわれない

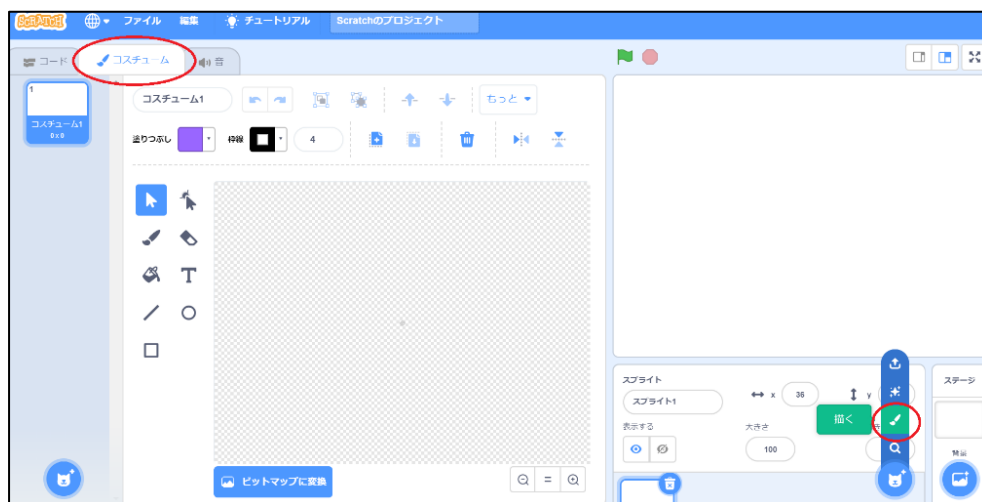
ビットマップ形式 (ラスター形式)

- ・画像情報をピクセルの集合として保存
- ・色の変化を細かく表現することができる

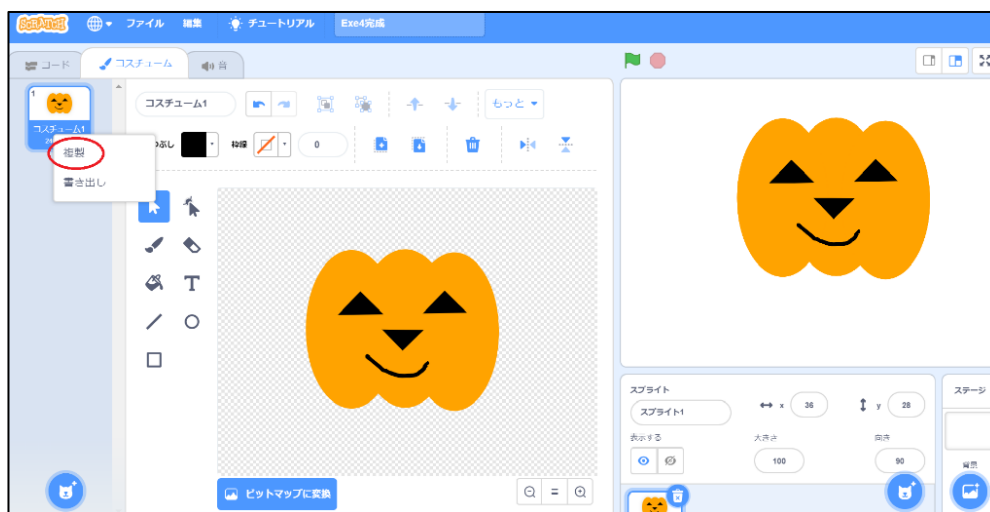
オリジナルスプライトの作成

1. 自分でスプライトを描画しましょう。

- 1)スプライトリストの「描く」クリック
- 2)コスチュームタブをクリックし、スプライトを描画する



↓ベクターモードでかぼちゃを作成した例



- ・「ビットマップに変換」「ベクターに変換」ボタンをクリックすると「ビットマップモード」「ベクターモード」の切り替えができます。
- ・コスチュームを右クリック→「複製」すると、コスチュームをコピーできます。

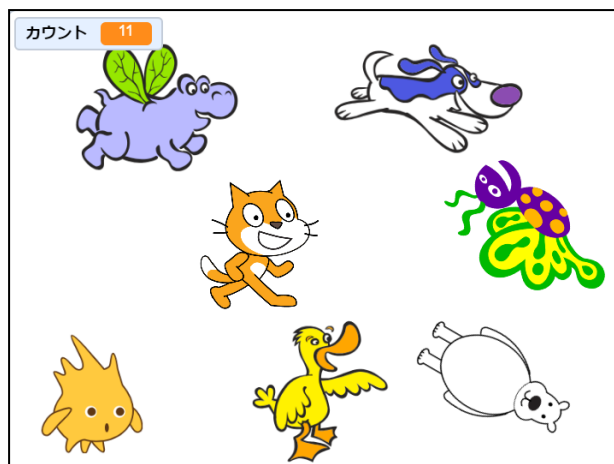
3)複数のコスチュームを作成しましょう。

演習3

クリックしてスプライトを消すゲームです。
変数を利用して、消したスプライトの数をカウントしまし
よう。

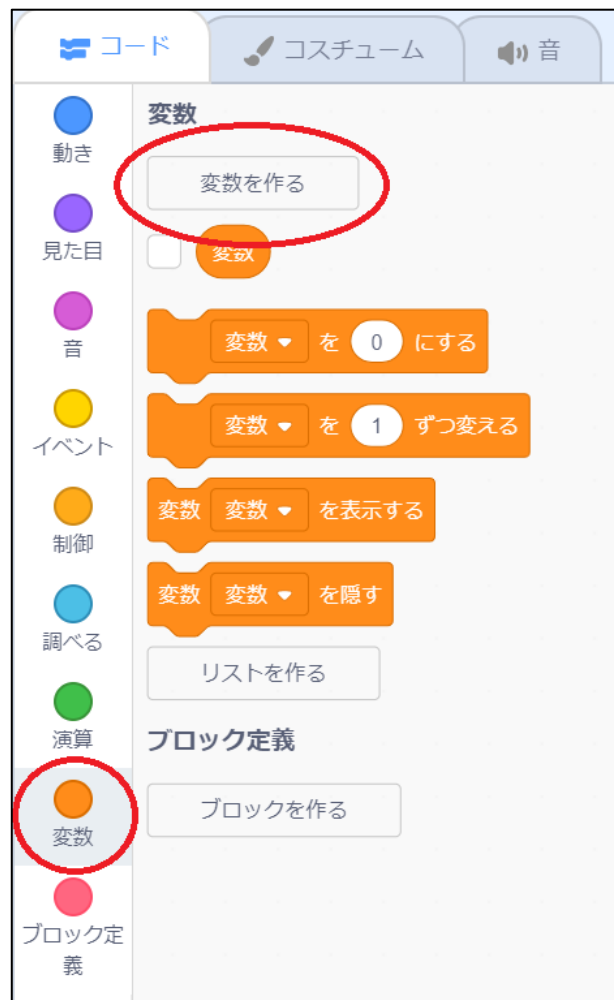
(クリックゲーム1)

ファイル名:Exe3

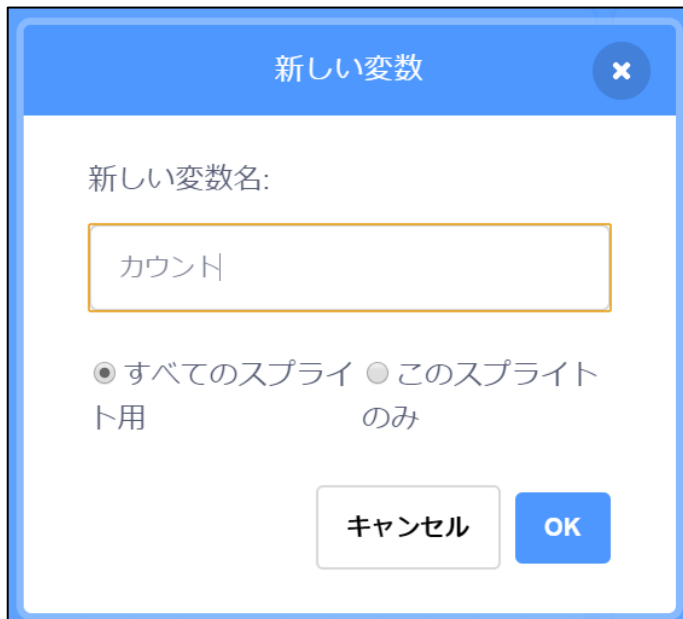


1. 変数の作成

- ① スクリプトカテゴリー「データ」を選択
- ② 「変数を作る」をクリック



③ 変数名:「カウント」、「すべてのスプライト用」を選択し「OK」をクリック。



2. スクリプトの作成

■ Cat スプライト



3. スプライトを増やしましょう。

4. 各スプライトの動作をアレンジしましょう。

ランダム処理の活用

- ランダムな場所へ行く
- ランダムな秒数待つ
- ランダムな大きさにする
- ランダムな色にする
- ランダムな渦巻きにする

5. 旗が押されたときにカウントをリセットするようにしましょう。

演習4

「制御」のブロックを使って数当てゲームを作りましょう。

ファイル名: Exe4

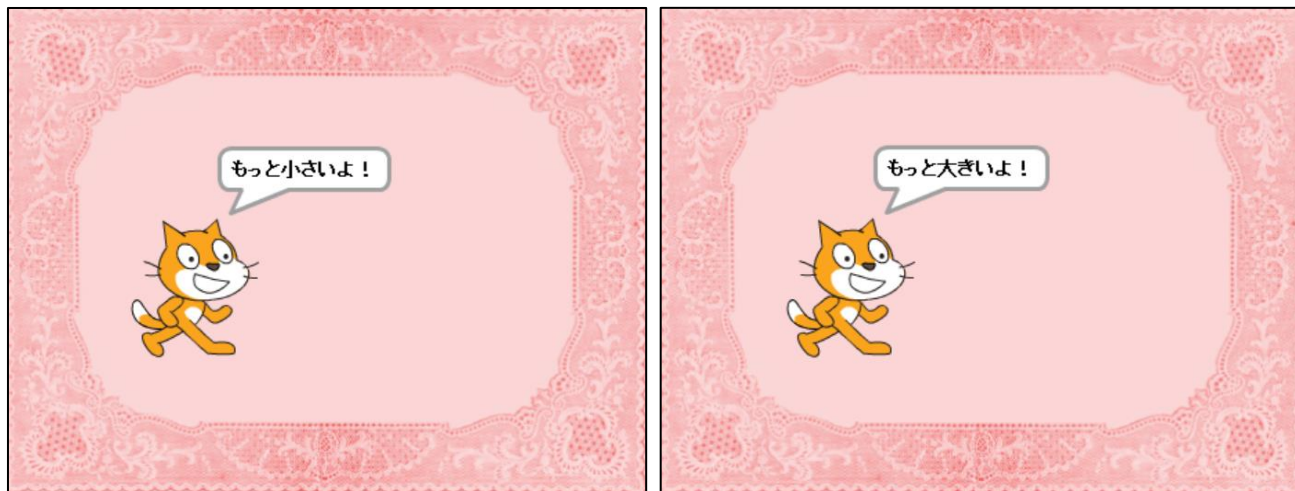


内容

- 変数「数」を作成
- 旗がクリックされたらゲーム開始
- 「数」の初期化1 から 100 までの乱数にする
カテゴリ:変数 「○を○にする」ブロック
- 「いくつかな？(1 から 100 までの数字を半角で入力して)」と聞いて待つ。 カテゴリ:「調べる」
- もし「答え」と「数」が等しければ「当たり」と 2 秒言う、でなければ「はずれ」と 2 秒言う。
カテゴリ:制御 「もし△なら でなければ」ブロック
△部分のブロックは、カテゴリ:演算から形の合うものを探しましょう。

演習5

数当てゲームをバージョンアップしましょう。



ファイル名: Exe5

1. 5回までチャレンジできるようにしましょう。

正解した場合は、

- お祝いの音楽を鳴らして
- 「当たり」と2秒言う
- 「制御」→「すべてを止める」でプログラムを終了しましょう。

2. 「はずれ」と2秒言った後で、

「数」が「答え(入力した値)」より小さければ、「もっと小さいよ！」

「数」が「答え(入力した値)」より大きければ、「もっと大きいよ！」

とアドバイスするようにしましょう。

3. 変数を活用して、何回目かで正解したか表示するようにしましょう。変数名をお任せします。

正解した場合は、

- お祝いの音楽を鳴らして
- 「当たり」と2秒言う
- 「〇回で正解！」 ← **この部分を追加**
- 「制御」→「すべてを止める」でプログラムを終了しましょう。



演習6

クリックゲーム 時間制限版です。

30 秒を超えたらゲームを終了するようにします。

演習 3 をもとに作成します。

ファイル名 : Exe6



1. タイマーを表示しましょう。

「調べる」 → 「タイマー」

にする

2. ステージに背景2つ、おめでたそうな背景とおめでたくなさそうな背景を追加します。

3. ステージに音2つ、おめでたそうな音とおめでたくなさそうな音を追加します。

4. ステージに「30 秒を超えたらゲームを終了する」ようにスクリプトを追加します。

・タイマーが 30 秒超えたら

- おめでたそうな背景に変更する
- おめでたい音を鳴らす(「終わるまで〇〇の音を鳴らす」)
- ゲームを終わる(「すべてを止める」)

5. ゲーム終了時、カウントの値によって背景や音楽を変えるようにバージョンアップしましょう。

ゲーム終了時

カウントが 10 以上の場合は

おめでたい背景にし、おめでたい音楽を鳴らしましょう。

そうでない場合は

おめでたくない背景にし、おめでたくない音楽を鳴らしましょう。

演習7

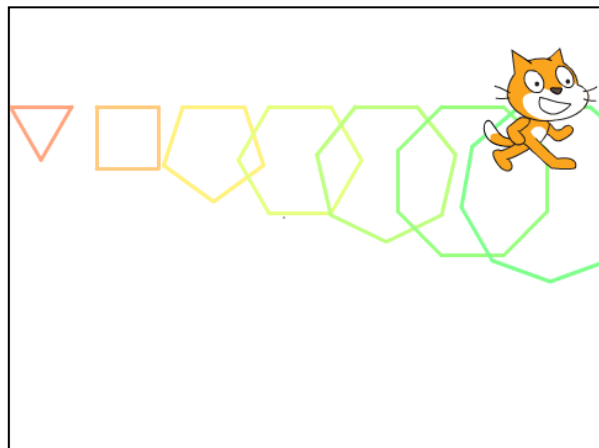
別紙参照

演習8

ペンを使って図形を描きましょう。

ファイル名：Exe8

辺の数	外角	辺の数×外角＝
3	120	$120 \times 3 =$
4	90	$90 \times 4 =$
5		
6		



1. ペンブロックを使って、正方形を描画しましょう。
2. 制御ブロックを使ったスクリプトに書き換えましょう。
3. 繰り返し回数や角度を変更し、三角形や五角形を描画してみましょう。
4. 変数：「辺の数」を作成し、この変数の値を変更するだけで、いろいろな多角形が描画できるようにスクリプトを書き換えましょう。
5. ペンの色や太さを変えてみましょう。



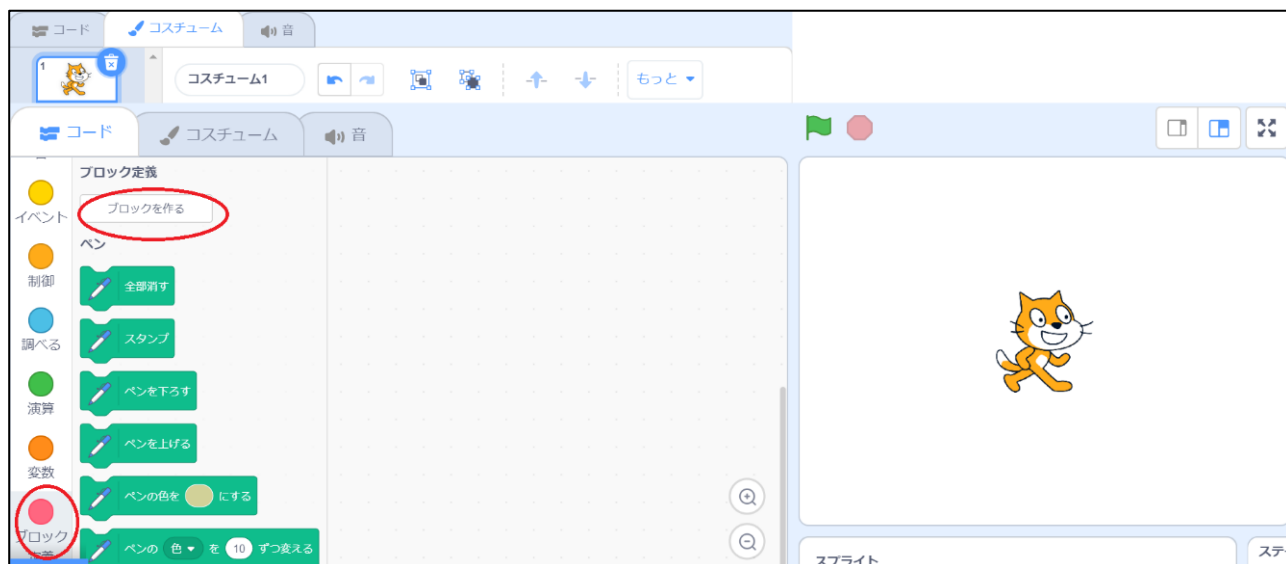
演習 9

オリジナルのブロックを作成して、多角形を描画しましょう。

ファイル名：Exe9

1. 「三角形」というブロックを作成します。

カテゴリ：「ブロック定義」 → 「ブロックを作る」



ブロック名：「三角形」



2. 定義「三角形」のブロックに以下のようなスクリプトを作成し、旗がクリックされたときに、「三角形」を実行しましょう。



3. 同様にブロック「四角形」、「五角形」を作成し、それぞれ実行してみましょう。

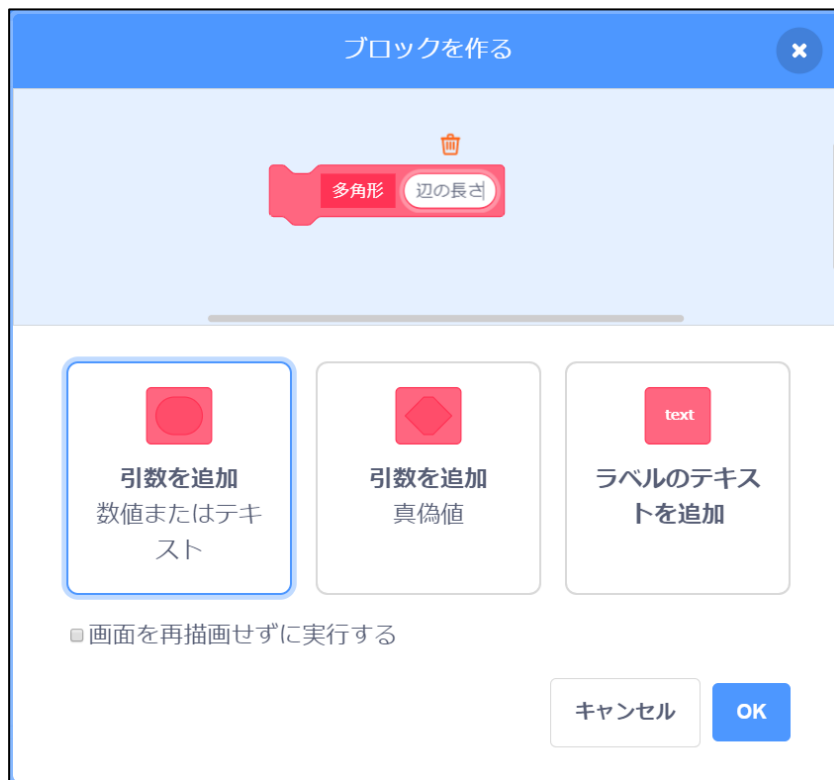
4. さまざまな多角形を描画できるブロック「多角形」を作成しましょう。

ブロック実行時に、辺の長さと辺の数を指定できるようにします。

ブロック名：「多角形」とし、「引数を追加 数値またはテキスト」をクリック



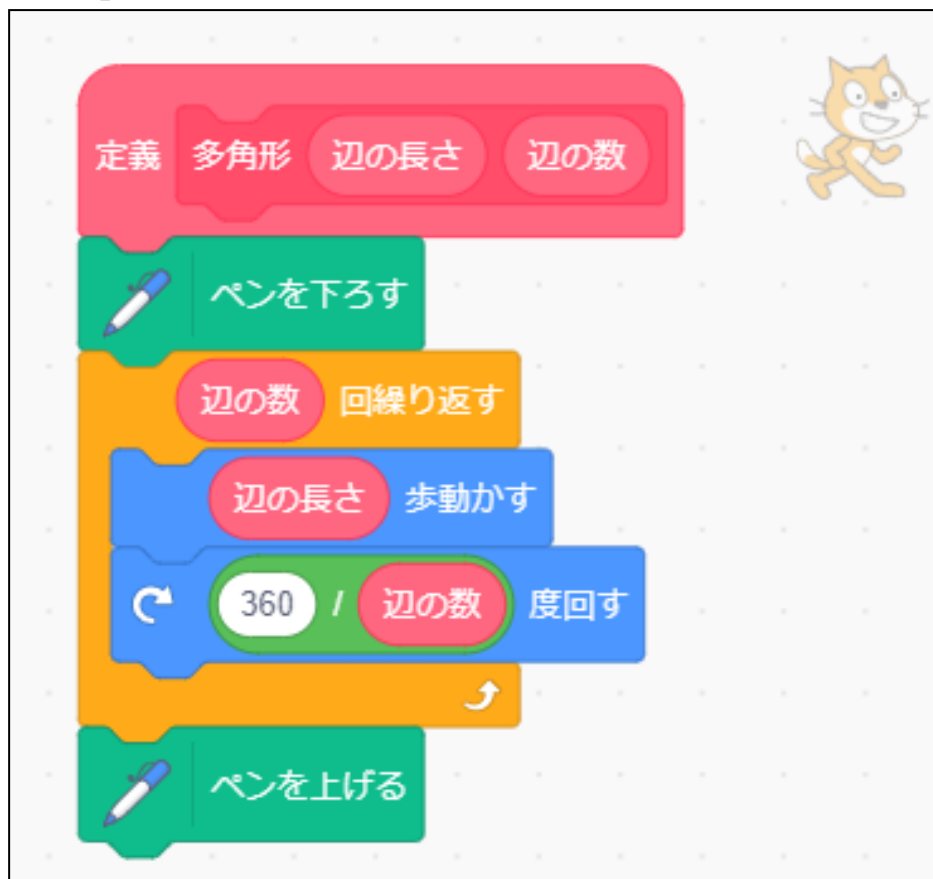
引数名：「辺の長さ」と入力し、



さらに、「引数を追加 数値またはテキスト」をクリックし、
引数名：「辺の数」と入力



5. 定義「多角形」のブロックに以下のようなスクリプトを作成します。



6. 旗がクリックされたときに、「多角形」を実行しましょう。

「多角形」ブロックを使用したスクリプト例



A Scratch script starting with a 'when green flag is clicked' event. The script performs the following steps: 1. 'Erase everything' block. 2. 'Set x coordinate to -240, y coordinate to 0' block. 3. 'Set pen color to 0' block. 4. 'Set pen thickness to 5' block. 5. 'Set number of sides to 3' block. 6. A 'repeat 5 times' loop containing: 'draw polygon with 50 sides and number of sides' block, 'move 70 steps' block, and 'decrease number of sides by 1' block. 7. 'Change pen color by 30' block.



A Scratch script starting with a 'when green flag is clicked' event. The script performs the following steps: 1. 'Erase everything' block. 2. 'Set x coordinate to 0, y coordinate to 0' block. 3. 'Set pen color to 0' block. 4. 'Set pen thickness to 1' block. 5. A 'repeat 50 times' loop containing: 'draw polygon with 50 sides and 5' block, 'rotate 360 / 50 degrees' block, and 'change pen color by 2' block.

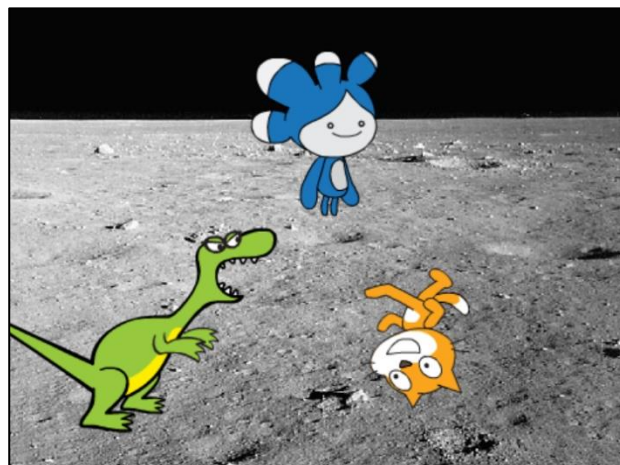
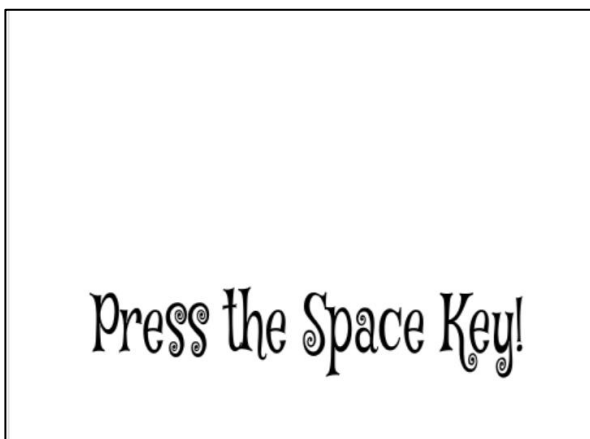
演習 10

ステージやスプライト間でメッセージをやりとりしてみましょう。

ファイル名：Exe10

1. オープニング画面を作成しましょう。

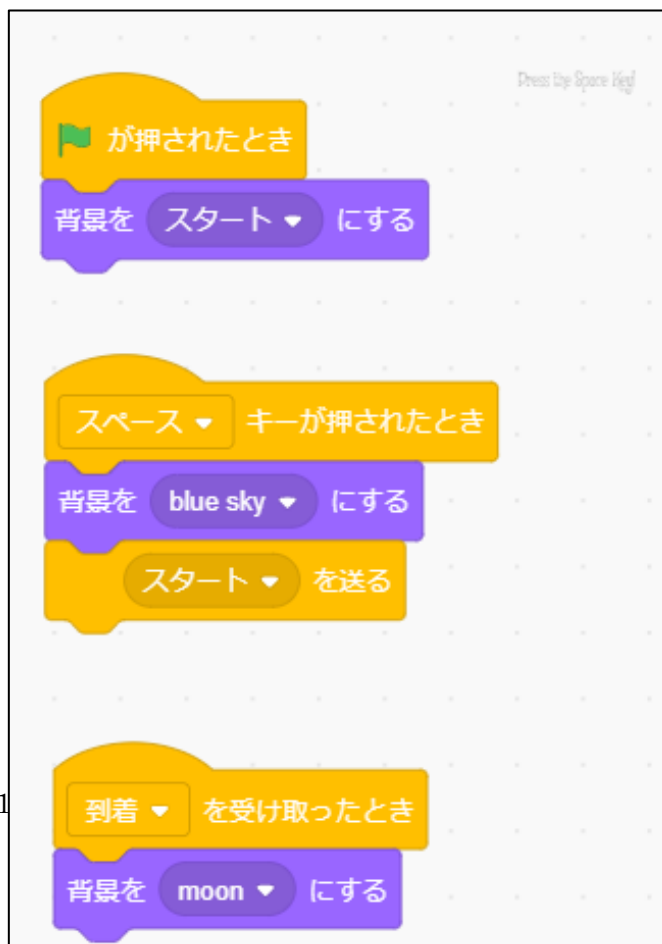
ステージ → 背景



2. 背景を後2つ追加しましょう。

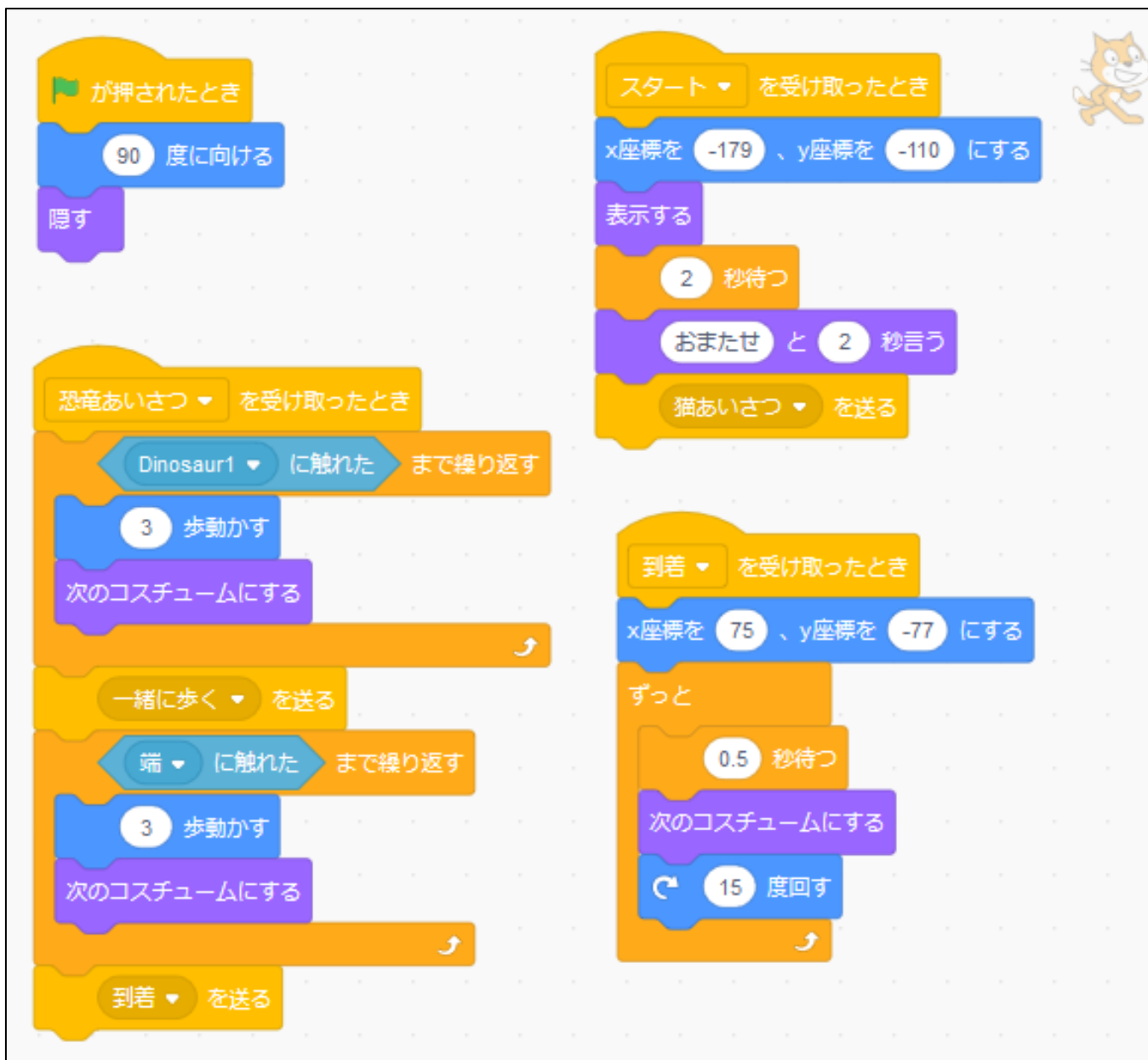
- ・スペースキーを押されたときの背景
- ・到着したときの背景

3. ステージにスクリプトを作成しましょう。



4. スプライト「Dinosaur1」、「Tera」を追加しましょう。

5. ネコのスク립ト



The image shows a Scratch script for a cat sprite, organized into two columns. The script is as follows:

- Column 1 (Left):**
 - When green flag is clicked:
 - Turn 90 degrees
 - Hide
 - When dinosaur says "恐竜あいさつ":
 - Repeat until "Dinosaur1" is touched:
 - Walk 3 steps
 - Change to next costume
 - Send "一緒に歩く"
 - Repeat until "端" is touched:
 - Walk 3 steps
 - Change to next costume
 - Send "到着"
- Column 2 (Right):**
 - When started:
 - Set x coordinate to -179, y coordinate to -110
 - Show
 - Wait 2 seconds
 - Say "おまたせ" for 2 seconds
 - Send "猫あいさつ"
 - When "到着" is received:
 - Set x coordinate to 75, y coordinate to -77
 - Forever loop:
 - Wait 0.5 seconds
 - Change to next costume
 - Repeat 15 times

A small Scratch cat icon is visible in the top right corner of the script area.

6. Dinosaur1 のスクリプト

The image shows a Scratch script for a dinosaur character named 'Dinosaur1'. The script is organized into several event-driven blocks:

- When green flag clicked:**
 - Turn 90 degrees.
 - Set rotation method to 'left only'.
 - Hide.
- When start message received:**
 - Set x coordinate to 59, y coordinate to -81.
 - Set costume to 'dinosaur1-e'.
 - Show.
- When arrival message received:**
 - Set x coordinate to -149, y coordinate to -69.
 - Wait 1 second.
 - Next costume.
- When cat message received:**
 - Turn -90 degrees.
 - Wait 2 seconds.
 - Set costume to 'dinosaur1-c'.
 - Say 'こんにちは。' for 2 seconds.
 - Say 'じゃ、行こうか' for 2 seconds.
 - Send '恐竜あいさつ'.
- When walk message received:**
 - Turn 90 degrees.
 - Set costume to 'dinosaur1-g'.
 - Repeat until 'end' is reached.
 - Walk 3 steps.

A small green dinosaur icon is visible in the top right corner of the script area.

7. Tera のスクリプト



演習11

メッセージを活用したオリジナルプロジェクトを作成しましょう。

ファイル名: Exe11

作成条件: メッセージのブロックを使う

アルゴリズム

問題を解決するための計算方法や処理手順のこと。
これに基づき具体的な手順を記述したものがプログラムである。

代表的なサーチ(探索)、ソート(整列)のアルゴリズムを学習しましょう。

リスト 配列(array)

複数個のデータを並べたもの。
リスト中の各データを要素(element)という。

インデックス(index)、添字(そえじ)

リストの各要素につけられた番号。1,2,3, …と、順に番号がついている。
(一般的なプログラミング言語では、0,1,2, …)

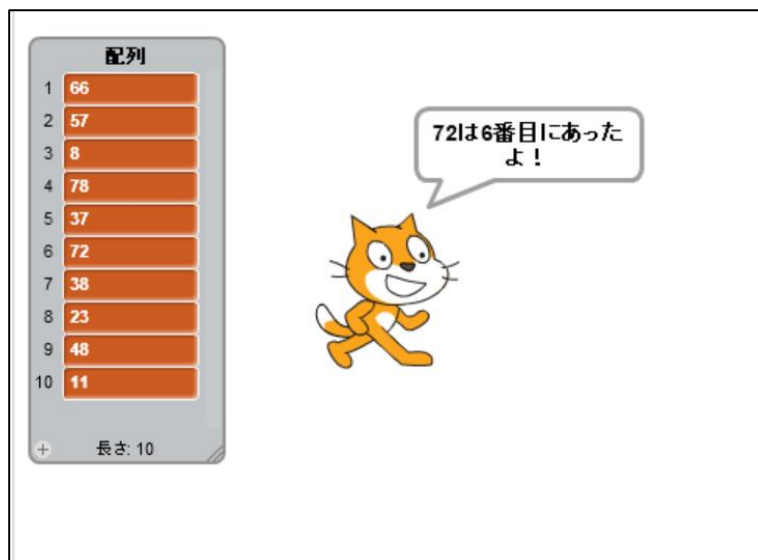
演習12

リニアサーチ(線形探索法)

ファイル名:Exe12

1. 以下の変数、リストを作成しましょう。

変数 「要素数」、「インデックス」
 リスト 「配列」



2. ネコに、旗を押す度に、上の図のようにリスト「配列」に要素数(10個)分に1から100までのランダムな数をセットするスクリプトを作成しましょう。
3. ネコにスクリプトを追加します。スペースキーを押すと、入力した数字を探すスクリプトを追加しましょう。
4. 数字が見つからなかったときには、「答えはなかったよ!」と2秒言うようにしましょう。

演習13

最小値、最大値のサーチ

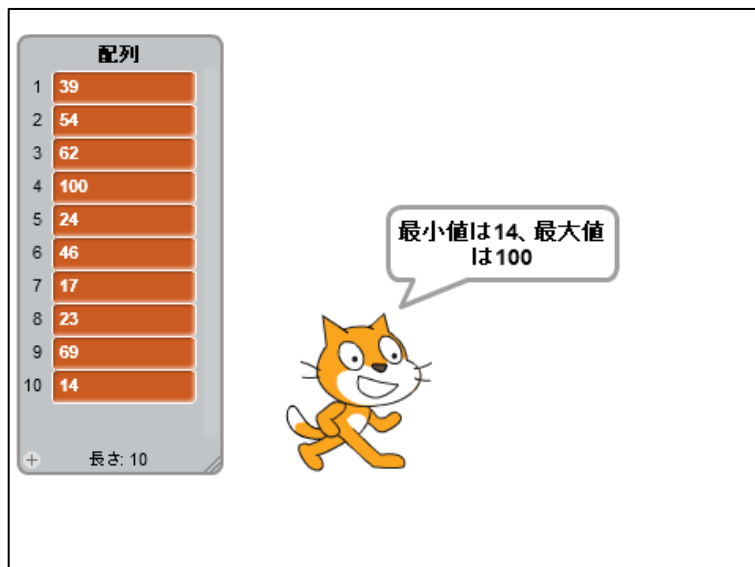
ファイル名: Exe13

1. 以下の変数、リストを作成しましょう。

ネコ → データ → リスト

変数 「要素数」、「インデックス」、
「最小値」

リスト 「配列」



2. ネコに、旗を押す度に、上の図のようにリスト「配列」に要素数(10個)分に1から100までのランダムな数をセットするスクリプトを作成しましょう。
3. ネコにスクリプトを追加します。スペースキーを押すと、最小値を探すスクリプトを追加しましょう。
4. 最大値も探して言いましょう。

演習14

バブルソート

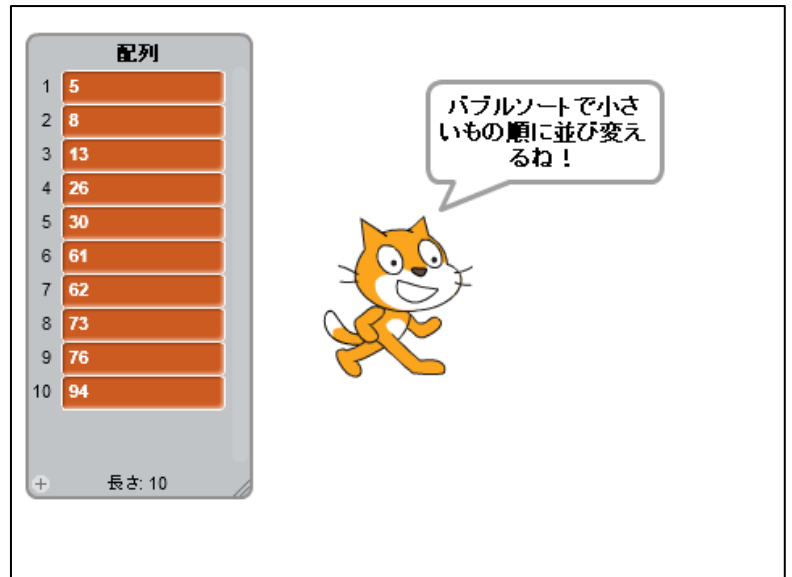
ファイル名: Exe14

1. 以下の変数、リストを作成しましょう。

ネコ → データ → リスト

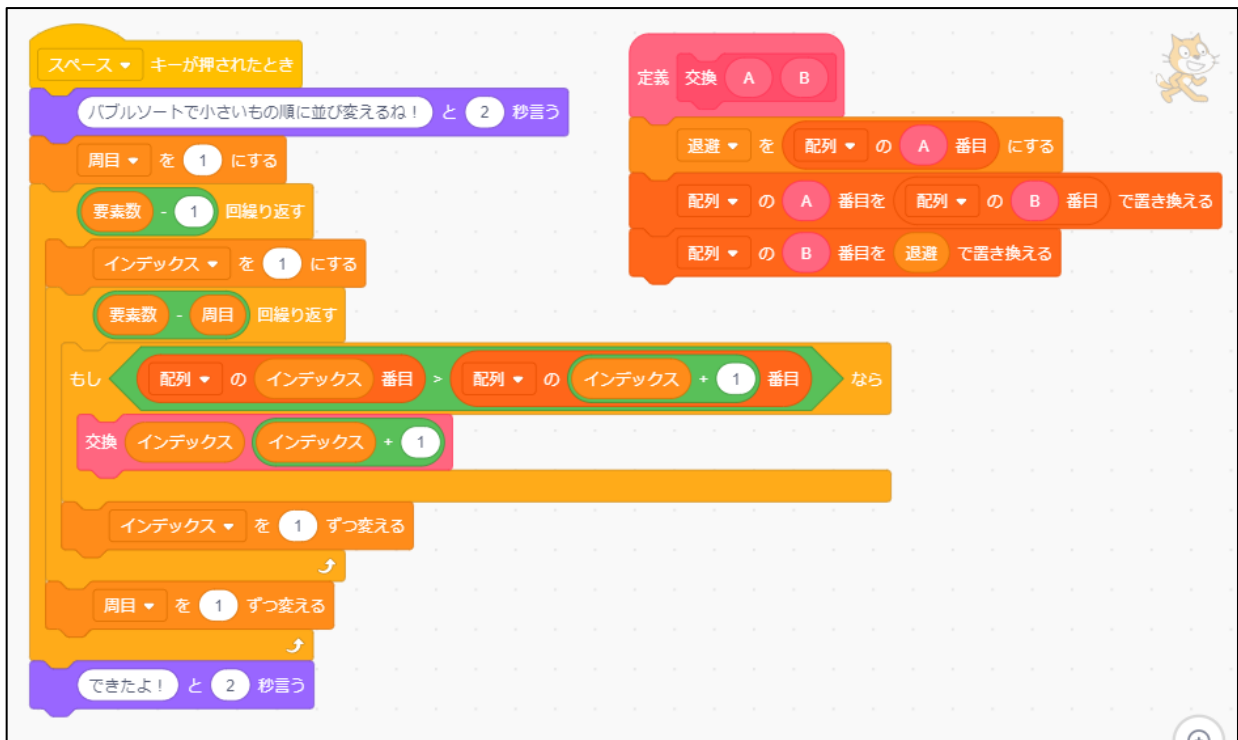
変数 「要素数」、「インデックス」、
「周目」、「退避」

リスト 「配列」



2. ネコに、旗を押す度に、上の図のようにリスト「配列」に要素数(10個)分に1から100までのランダムな数をセットするスクリプトを作成しましょう。

3. ネコにスクリプトを追加します。スペースキーを押すと、バブルソートで小さいもの順(昇順)に並び替えます。



4. 大きいもの順(降順)に並べ替えるにはどうしたらいいのか考えましょう。

演習15

選択ソート

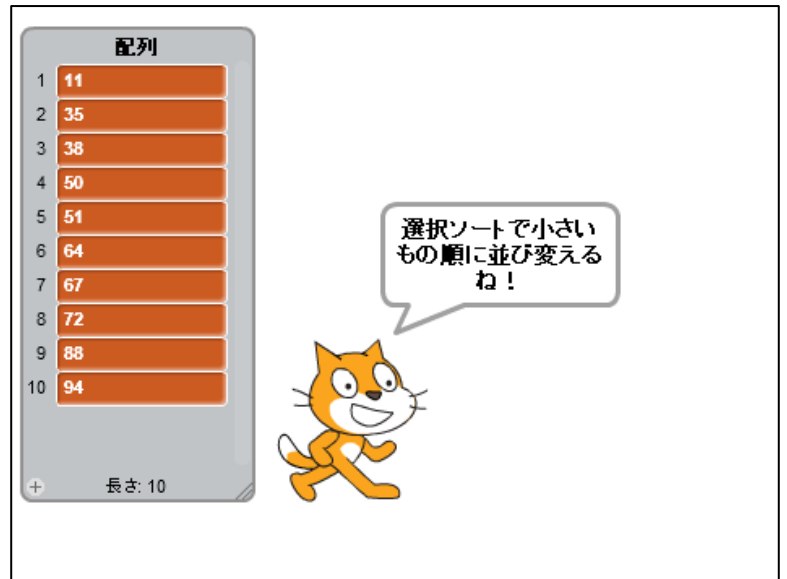
ファイル名: Exe15

1. 以下の変数、リストを作成しましょう。

ネコ → データ → リスト

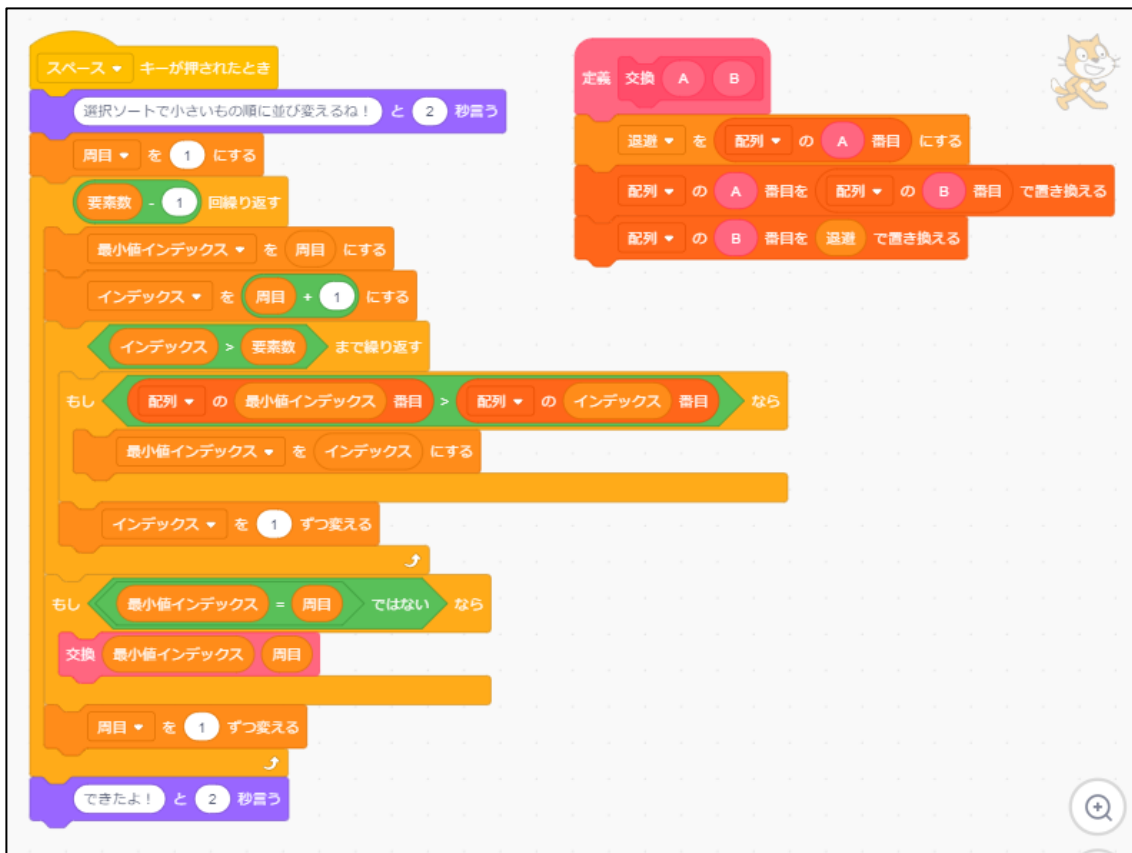
変数 「要素数」、「インデックス」、
「周目」、「退避」、
「最小値インデックス」

リスト 「配列」



2. ネコに、旗を押す度に、上の図のようにリスト「配列」に要素数(10個)分に1から100までのランダムな数をセットするスクリプトを作成しましょう。

3. ネコにスクリプトを追加します。スペースキーを押すと、選択ソートで小さいもの順(昇順)に並び替えます。



4. 大きいもの順(降順)に並べ替えるにはどうしたらいいのか考えましょう。

演習16

挿入ソート

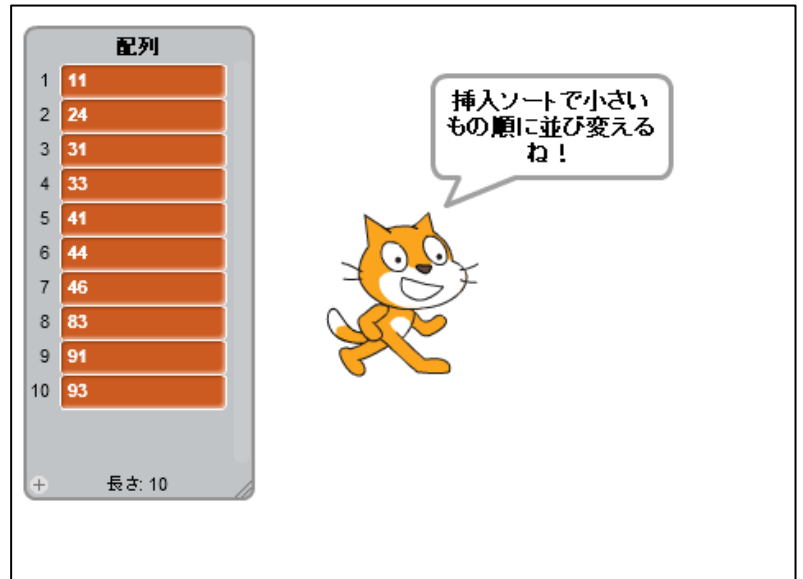
ファイル名:Exe16

1. 以下の変数、リストを作成しましょう。

ネコ → データ → リスト

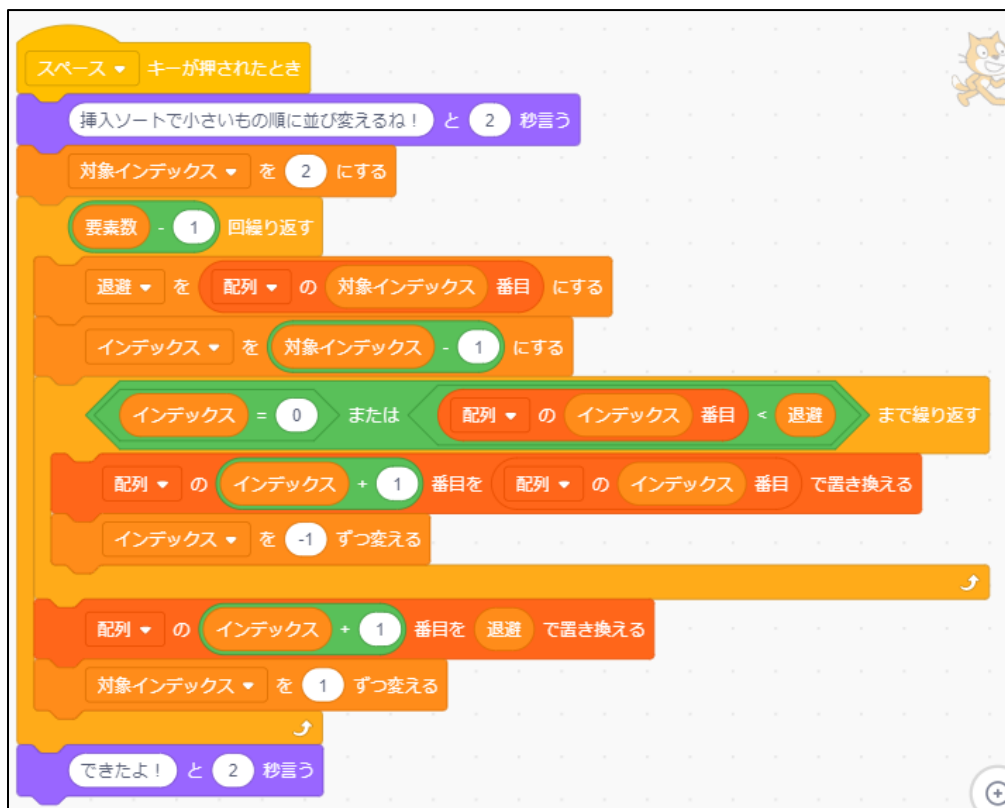
変数 「要素数」、「インデックス」、
「対象インデックス」、「退避」

リスト 「配列」



2. ネコに、旗を押す度に、上の図のようにリスト「配列」に要素数(10個)分に1から100までのランダムな数をセットするスクリプトを作成しましょう。

3. ネコにスクリプトを追加します。スペースキーを押すと、挿入ソートで小さいもの順(昇順)に並び替えます。



4. 大きいもの順(降順)に並べ替えるにはどうしたらいいのか考えましょう。

演習17

演習14、演習15、演習16の時間計測版を作成します。

ソート実行にかかった時間を表示しましょう。

ファイル名: Exe14t 、Exe15t 、Exe16t



1. 「旗がクリックされたとき」部分の変更

- ・リスト「配列」に要素数(10個)分に1から100,000までのランダムな数をセットするように変更
- ・リスト「配列」を隠し、セット後表示するようにしましょう。

2. 「スペースキーを押されたとき」、リスト「配列」を隠し、ソート後表示するようにしましょう。

処理時間を短縮するための処理

演習18

各アルゴリズムについて考察しましょう。

配布: 「ソートアルゴリズム.xlsx」「ソート時間計測2.sb3」

2. シート「ソートアルゴリズム」

各ソートアルゴリズムの説明、計算量を調べて、印刷して提出しましょう。

3. シート「計測データ」

ターボモード(編集→ターボモード)にして、

各ソート、要素が100、1000、10000の場合の処理時間を3回ずつ計測しましょう。

※シェルソート、クイックソートは配布した「ソート時間計測2.sb3」を使用して時間計測しましょう。

演習19

Scratch 課題

誰かに遊んでもらいたくなるようなオリジナルプロジェクトを作成しましょう。

ファイル名:Exe19

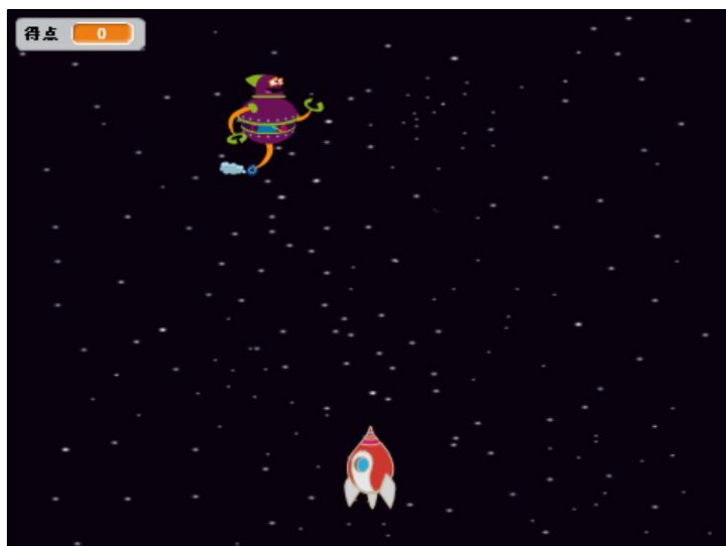
作成条件:変数を使う

演習20

おまけの Shooting Game

ファイル名: Exe20

1. スプライト「Spaceship」「Robot1」「Ball」「Ball2」を追加しましょう。
Cat スプライトは削除しましょう。



2. Spaceship のスクリプト

```

    1. 旗が押されたとき
       大きさを 30 %にする
       x座標を 0、y座標を -120 にする
       スタート を送る

    2. 右向き矢印 キーが押されたとき
       x座標を 10 ずつ変える

    3. 左向き矢印 キーが押されたとき
       x座標を -10 ずつ変える

    4. スタート を受け取ったとき
       ずっと
         もし Ball2 に触れた なら
           ゲームオーバー と 2 秒言う
           すべてを止める
       ずっと
         もし スペース キーが押された なら
           Ball のクローンを作る
           0.1 秒待つ
    
```

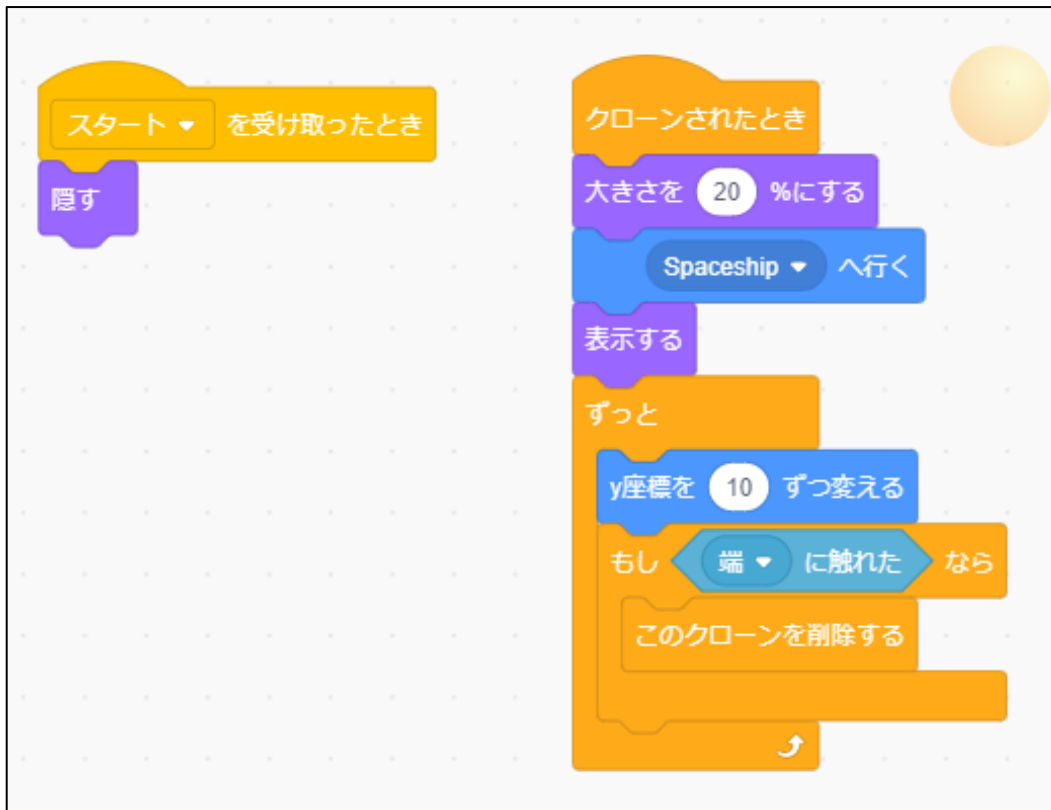
3. Robot1 のスクリプト



The image shows a Scratch script for a character named Robot1. The script is as follows:

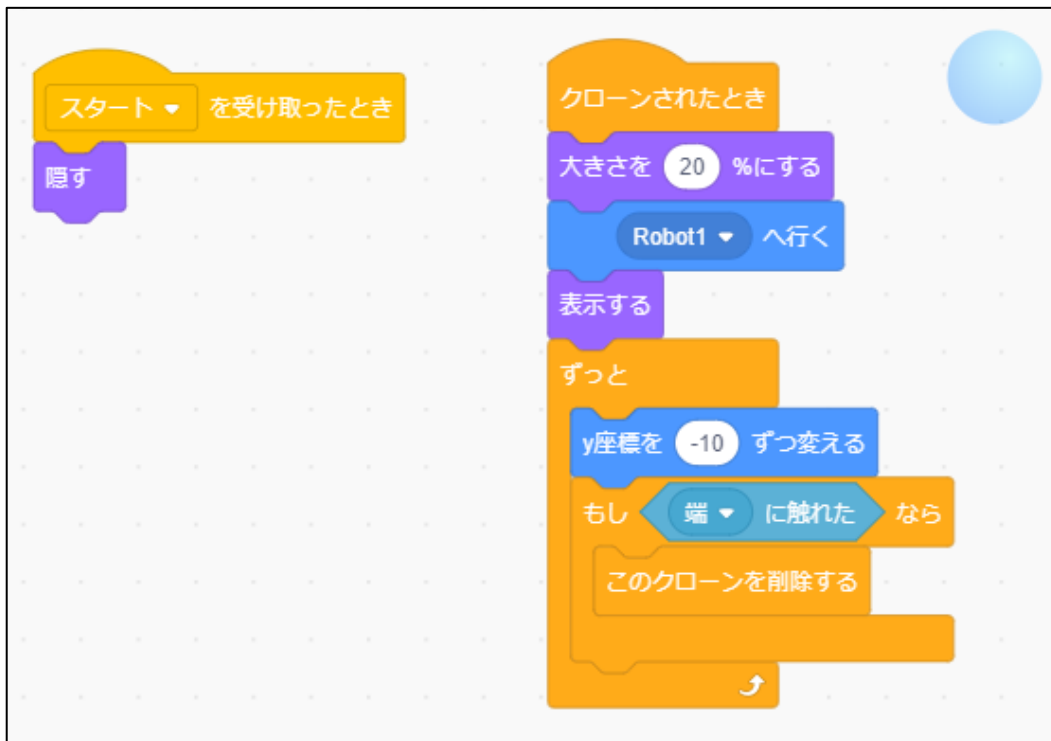
- スタート** を受け取ったとき
 - 大きさを **50** %にする
 - x座標を **150** 、y座標を **100** にする
 - 得点 を **0** にする
- ずっと**
 - 10** 歩動かす
 - もし端に着いたら、跳ね返る
 - もし **Ball** に触れた なら
 - color** の効果を **25** ずつ変える
 - 得点 を **1** ずつ変える
 - もし **得点 = 20** なら
 - やられた と **2** 秒言う
 - すべてを止める
 - Ball2** のクローンを作る

4. Ball のスクリプト



The image shows a Scratch script for a Ball object. On the left, the 'Start when green flag clicked' event block is followed by a 'Hide' block. On the right, the 'When cloned' event block is followed by a 'Set size to 20%' block, a 'Move to Spaceship' block, and a 'Show' block. A 'Forever' loop contains a 'Change y coordinate by 10' block, an 'If edge reached' block (set to 'End'), and a 'Delete this clone' block.

5. Ball2 のスクリプト



The image shows a Scratch script for a Ball2 object. On the left, the 'Start when green flag clicked' event block is followed by a 'Hide' block. On the right, the 'When cloned' event block is followed by a 'Set size to 20%' block, a 'Move to Robot1' block, and a 'Show' block. A 'Forever' loop contains a 'Change y coordinate by -10' block, an 'If edge reached' block (set to 'End'), and a 'Delete this clone' block.