

この教材の使い方

前提

☆ Micro:bit の基本操作経験

学習のポイント

☆ 配線

必要なもの

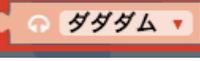
スピーカー

わにぐちクリップ2本

音をならそう！

1.プログラムをつくろう！

 **音楽** をクリック。

   をクリックして「最初だけ」

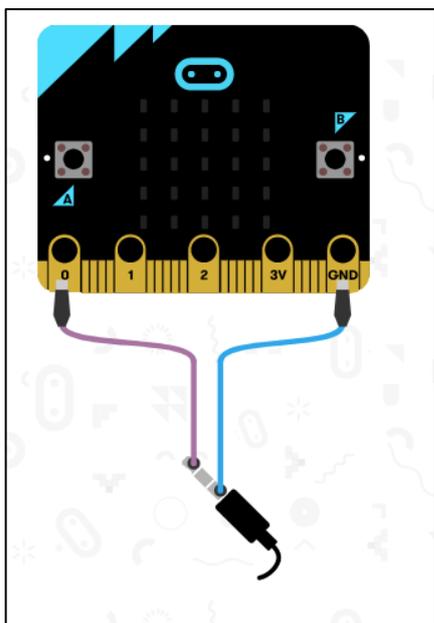
ブロックの中に入れる。

下の画面のようになれば OK。



※メロディは好きなものを選ぼう。

2.配線しよう！



上の画面を参考にスピーカーとワニグチクリップを配線しよう。

配線がおわったら、micro:bit にプログラムをダウンロードしよう。

※ダウンロードの仕方は、前回のテキストにあります。

※micro:bit 背面のリセットボタンをおすと、「最初だけ」ブロックを何回も実行することができます。

この教材の使い方

前提

- ☆ Micro:bit の基本操作経験
- ☆ Scratch で迷路ゲームを作ったことがあること
- ☆ Scratch でフルーツキャッチを作ったことがあること

学習のポイント

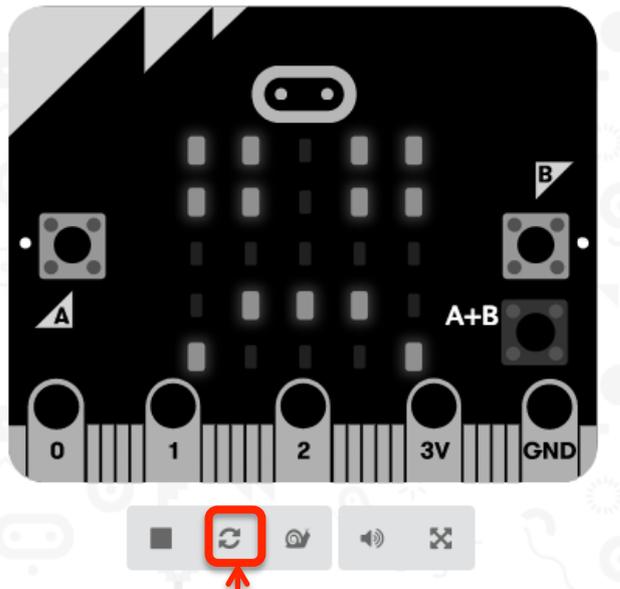
- ☆ Micro:bit の座標系
- ☆ Micro:bit のゲームライブラリとスプライト操作。
- ☆ 変数の復習

注意

下記のようなエラーが出たとき。



- (1) 変数名が間違っていないかチェックする。
- (2) それでもダメなときは・・・



ここをクリックして更新する。

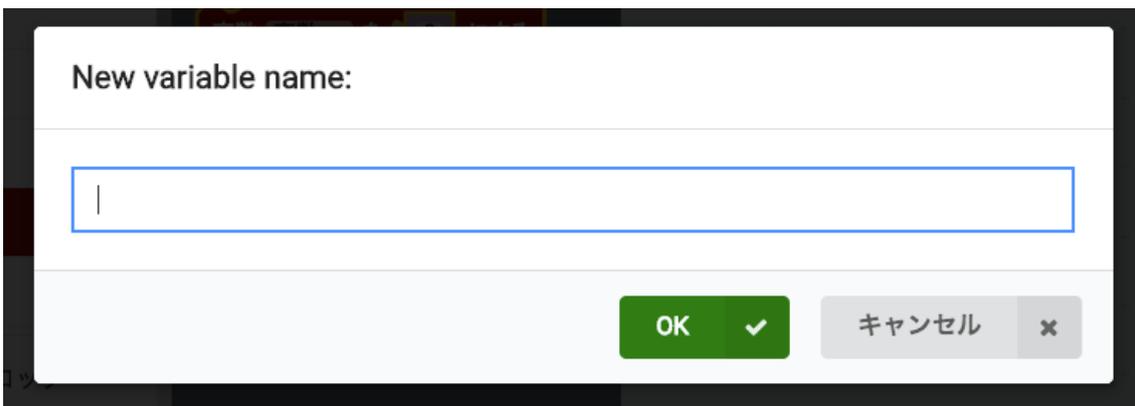
迷路ゲームをつくろう！

1. プレイヤーの変数をつくろう！

 変数 をクリック。

 をクリック。

下記のウィンドウがあらわれたら、変数名(New variable name)に「p」と入力し、OK をクリック。



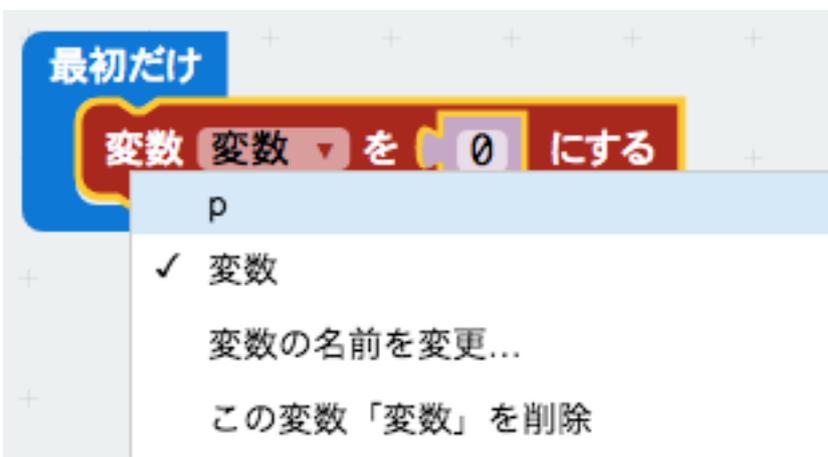
New variable name:

OK ✓ キャンセル ✕

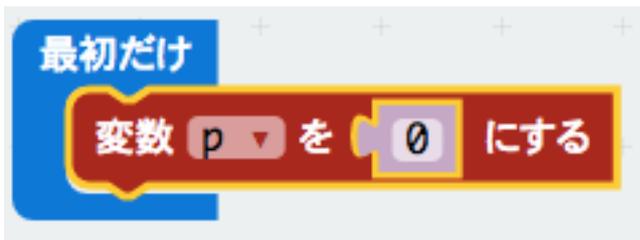
[変数「変数」を0にする]ブロックを「最初だけ」ブロックの中に格納する。



変数名を「p」に変更する。



以下のような画面になれば OK。



「変数」とは・・・

変化する数値などを格納しておくための入れ物のこと。

ここでは、プレイヤーのSpriteの現在位置を記録する。

復習：Scratchの「フルーツキャッチ」では、

変数を使ってフルーツをキャッチした数を得点として記録していた。

2. プレイヤーのSpriteをつくらう！

▼ 高度なブロック

をクリック。

∞ ゲーム

をクリック。

∞ Spriteを作成

X:

2

Y:

2

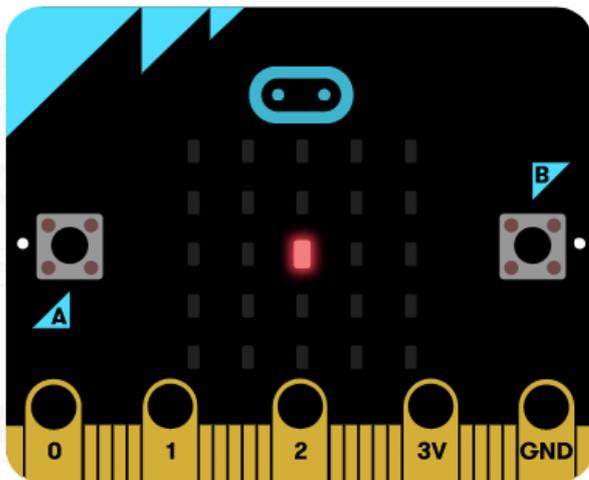
をクリック。

「最初だけ」ブロックの中に下記の画面になるように格納する。



シュミレーターの画面上の LED が下記のようになることを確認する。

※「X 座標=2、Y 座標=2」の位置が点灯



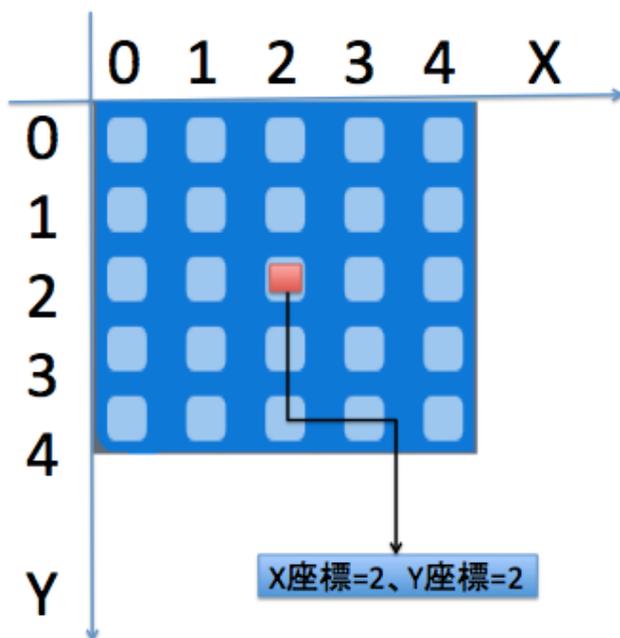
「座標」とは・・・

LED やスプライトの表示位置などを決めるための位置情報。

X 座標と Y 座標の組み合わせであらわす。

復習：Scratch の「迷路ゲーム」や「もぐらたたきゲーム」でも座標を使用してスプライトの表示位置を決めていた。

micro:bit の座標系は下記のようにになっている。





プレイヤーのスプライトをスタート地点（例 X 座標=0、Y 座標=0）に移動するにはどうしたらよいだろうか？

次のプリントに行くまえに、考えてみよう！

プレイヤーのSpriteをスタート地点 (X 座標=0、Y 座標=0) に移動するには・・・

下記の画面の赤枠内のようにスクリプトを変更する。



※スタート地点は、X 座標=0、Y 座標=0でなくても、好みの位置でかまいません。

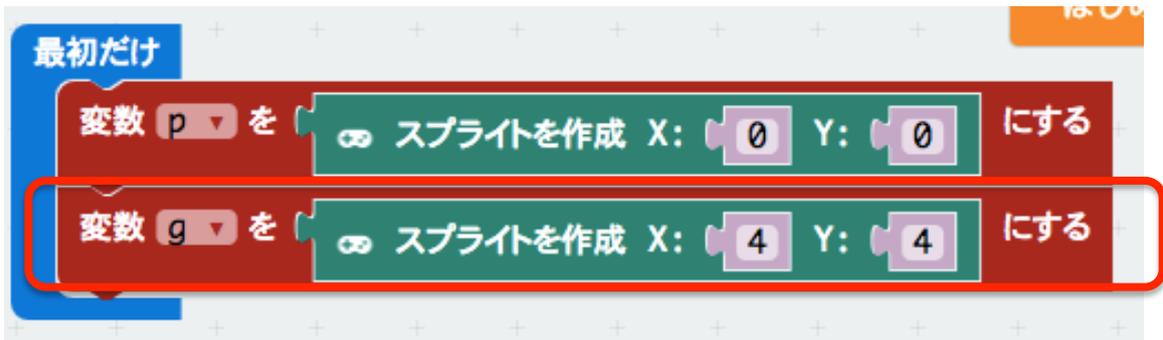


ゴールのSpriteをゴール地点(例:X 座標=4、Y 座標=4)に表示するにはどうしたらよいだろうか？

次のプリントにいくまえに、考えてみよう！

ヒント：プレイヤーのSpriteの作成手順を思い出してみよう。
(まず、ゴールの変数「g」を作る必要があります。)

ゴールのSpriteをゴール地点(例:X座標=4、Y座標=4)に表示するには・・・
下記の画面の赤枠内のようなブロックを追加する。



※ゴール地点は、X座標=0、Y座標=0でなくても、好みの位置でかまいません。

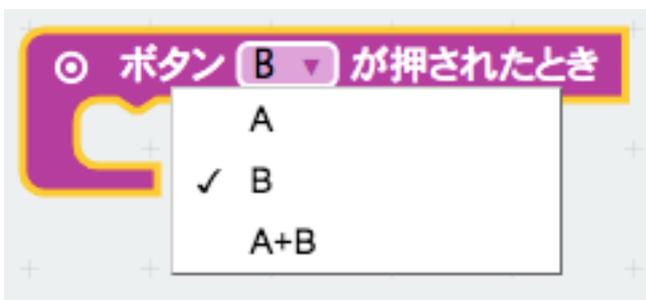
3. プレイヤーのSpriteを動かそう！

(1) B ボタンを押したときに、プレイヤーのSpriteが右方向に移動するようにします。

入力 をクリック。

ボタン A が押されたとき をクリックしてスクリプトエリアに配置。

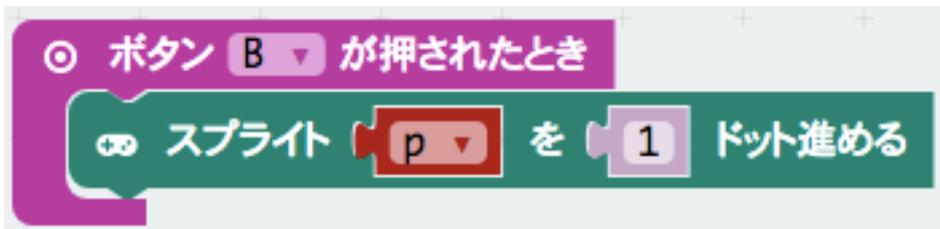
下の画面のように「ボタン B が押されたとき」に変更する。



ゲーム をクリック。

Sprite をクリックして、「ボタン B が押されたとき」ブロックの中に格納。

変数名を「p」に変更し、以下の画面になれば OK。



(2)

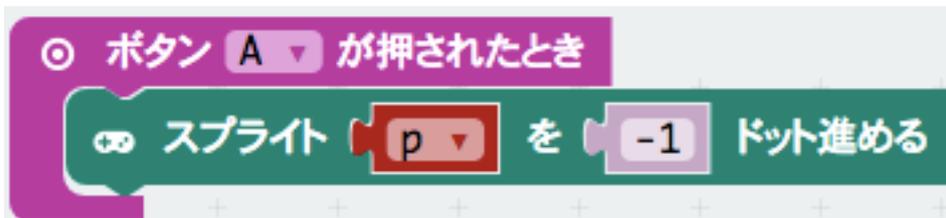


A ボタンを押したときに、プレイヤーが左に移動するようにするにはどうしたらよいだろうか？

次のプリントにいくまえに、考えてみよう！

ヒント：数字の 1 のところにはマイナスの数も入れられます。
Scratch の時に学習したことを思いだしてみよう！

A ボタンを押したときに、プレイヤーが左に移動するようにするには・・・
下の画面のようなスクリプトブロックを作成します。



(3)A+B ボタンを 1 回押したときに、プレイヤーが下方向に方向転換（右に 90 度回転）するようにします。
下の画面のようなスクリプトブロックを作成します。



プレイヤーのSpriteを上下左右に動かせるようになったことを確認しよう！

4. ゴールしたらアイコンが表示されるようにしよう!

✖ 論理 をクリック。



をクリックし、「ずっと」ブロックの中に格納。

下の画面のようになれば OK。



∞ ゲーム をクリック。



をクリックして、「ずっ

と」ブロックの中に、下の画面になるように格納。



変数名を「p」に変更。下の画面のようになれば OK。



≡ 変数

をクリック。



をクリックして「ずっと」ブロックの中に、下の画面のようになるように格納。



をクリック。



をクリックして、「ずっと」ブロックの中に格納。

下のような画面になれば OK。



プレイヤーを操作してゴールさせてみよう！

完成したら、micro:bit にダウンロードしよう。

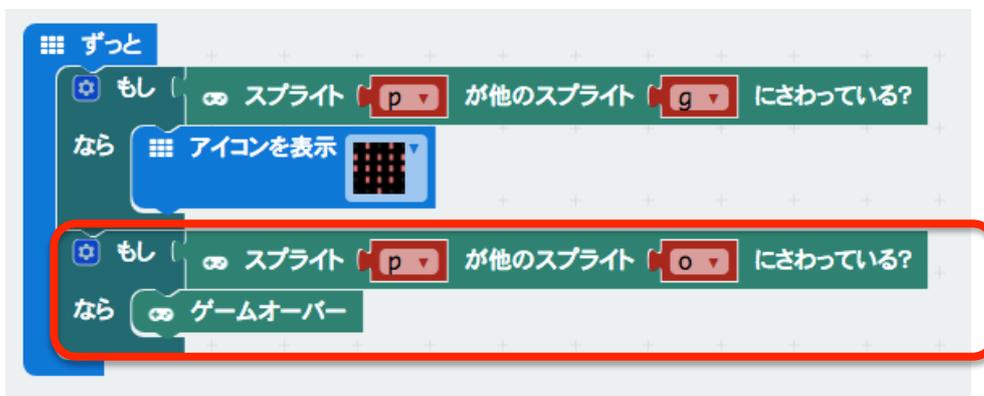
※ダウンロードの仕方は、前回のテキストにあります。

5. 追加課題：できた人はやってみよう！

(1) 障害物を配置しよう。



(2) 障害物にぶつかったらゲームオーバーになるようにしよう！



(3) 障害物を動かそう！

