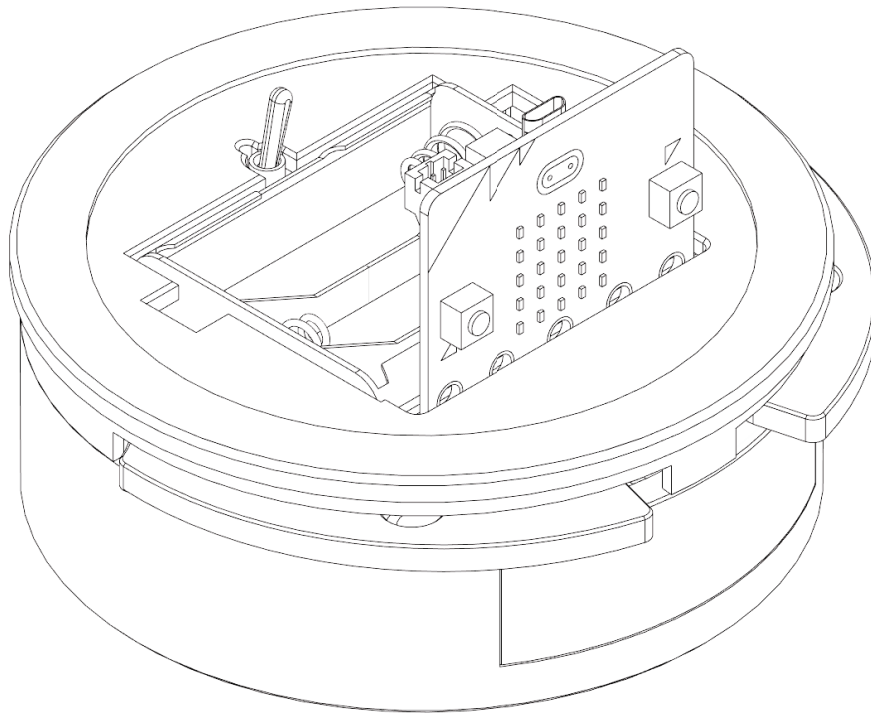


# micro:bit プログラミングカーリング 手引書



動画資料 URL <https://sedu.link/curling-mv>

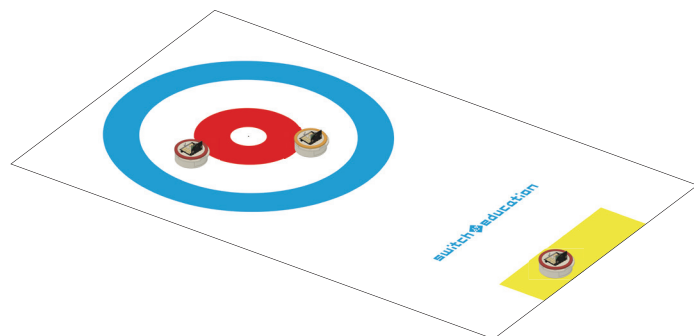


2023/12/08

# プログラミングカーリング

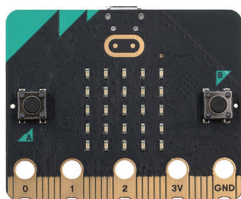
本書は小学校の先生等、プログラミングカーリングの運営を担当する方に向けた資料です。プログラミングカーリングに挑戦する上でおすすめの情報を記していますが、状況に応じて内容を変更してお楽しみください。

プログラミングカーリングとは、簡単なプログラムを作り micro:bit でカーリングロボットを動かして行うゲームです。1人1台ロボットを動かして複数人で対戦します。プログラムは走る時間と向きを変えるだけのとても簡単なものです。ロボットを狙い通りに動かすためには、プログラミングをして実際にロボットを動かし、うまくいかなかった所を調整する作業を何度も繰り返すことになります。その試行錯誤を通してプログラミングに必要な操作手順の基礎を学習することができます。ゲームを通して戦略を考え、その通りにロボットを動かすことでプログラミングの楽しさを体験できます。

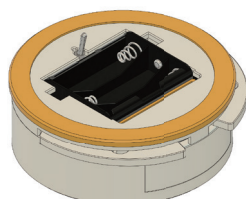


## 準備するもの

### 人数分必要な物



micro:bit



カーリングロボット  
(ロボットベース)



単3電池 x 3本

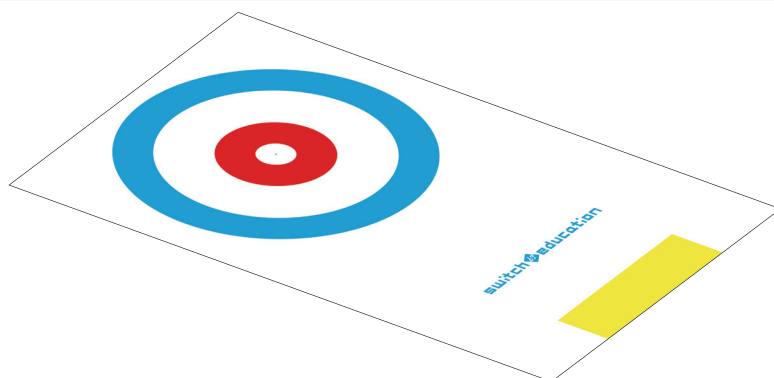


パソコン・タブレット



USB ケーブル  
(micro:bit 側 Micro-B)

### ゲームに1つ必要な物



カーリングコート

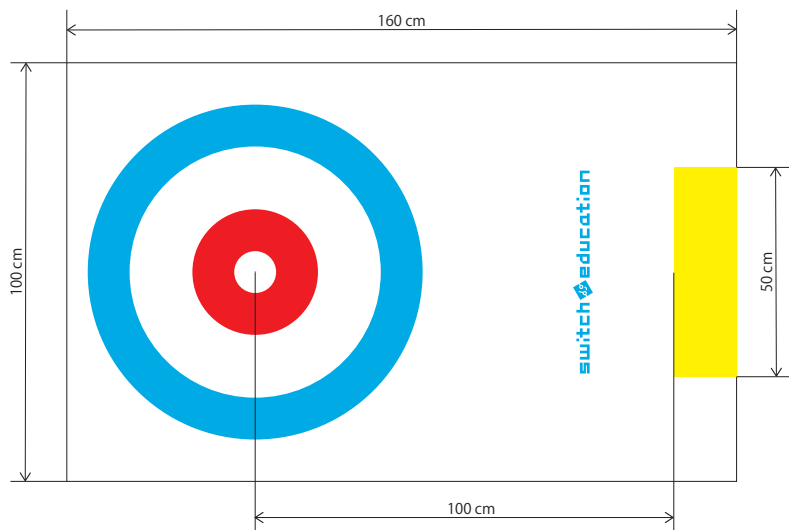
# 推奨ルール

## ① 参加人数 …… 8人（4対4）

1人1台カーリングロボットを使います。コートスペースの都合上、8台が推奨です。多すぎるとコートの中にロボットがたまりすぎてしまいます。

## ② コート

黄色いスタートエリア内からストーン（カーリングロボット）を走らせます。100cm先にあるハウス（円）の中心を目指します。



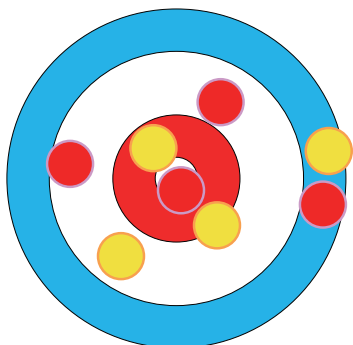
## ③ ゲームの進め方

赤チーム、黄チーム交互にストーンを走らせます。全てのストーンを走らせた時点でハウスの中心に近い色のストーンのチームに得点が入ります。一つのゲームをエンドと呼び、エンドを何度か繰り返して合計点を競います。2, 3エンド行うのがおすすめです。そのエンドで得点を得たチームは、次のエンドでは先攻となります。

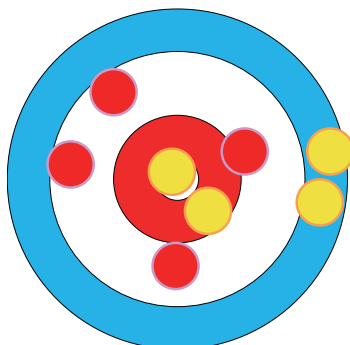
## ④ 得点の数え方

円の中心に最も近いストーンのチームにだけ得点が入ります。得点は相手チームよりも円の中心に近いストーンの数だけ入ります。

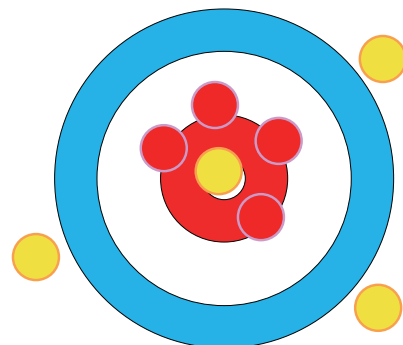
例



赤チーム 1点



黄チーム 2点



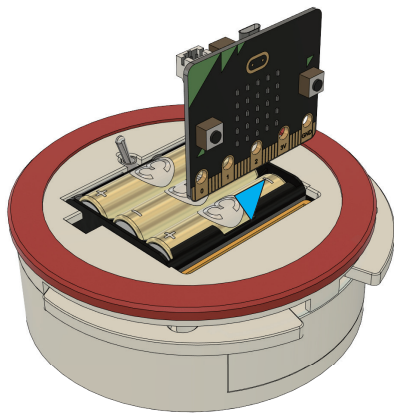
黄チーム 1点

## ⑤ ストーンの動作時間

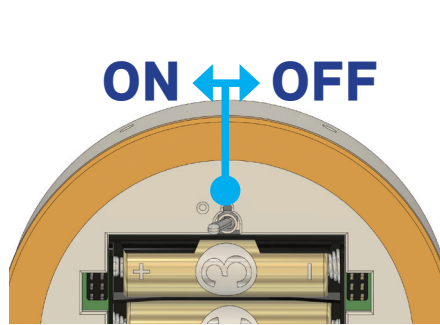
プログラムでストーンを動かしますので、ハウスの周りをぐるぐる回り続けるようにずっと動き続けることもできてしまいます。そうなるとゲームが進みませんので15秒程を上限として、その時間内にストーンの動作を停止するようにしてください。

	1エンド	2エンド	3エンド	合計
赤	1	0	0	1
黄	0	2	1	3

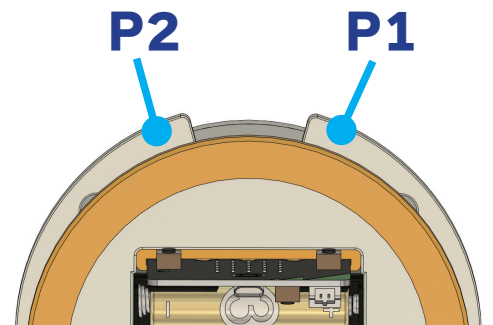
# カーリングロボット



micro:bit の向き



電源スイッチ

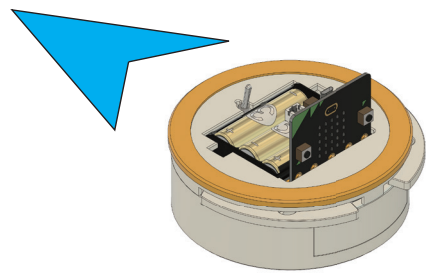


バンパースイッチ

正しい向きで単3電池を3本セットし、micro:bit を取り付けます（上図左）。プログラムを micro:bit にダウンロードしたらロボットの電源スイッチを ON にします。電源スイッチは小さな丸い凹みがあるほうが ON、反対側が OFF です。（上図中央）

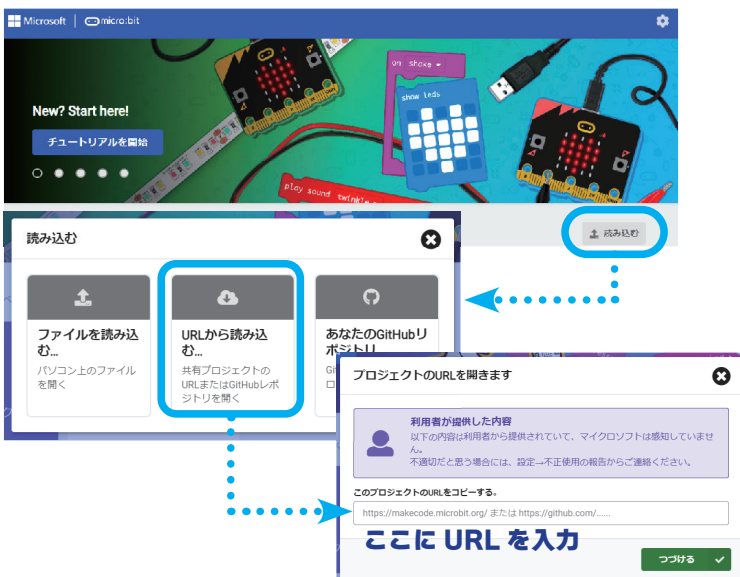
カーリングロボットにはバンパースイッチが付いています。バンパースイッチに他のロボットが接触した時にロボットを走らせることで、カーリングストーンが弾き飛ばされる様子を再現します。（上図右）バンパースイッチはそれぞれ micro:bit の P1 と P2 の端子に接続されています。

カーリングロボットの進行方向は右図の向きです。

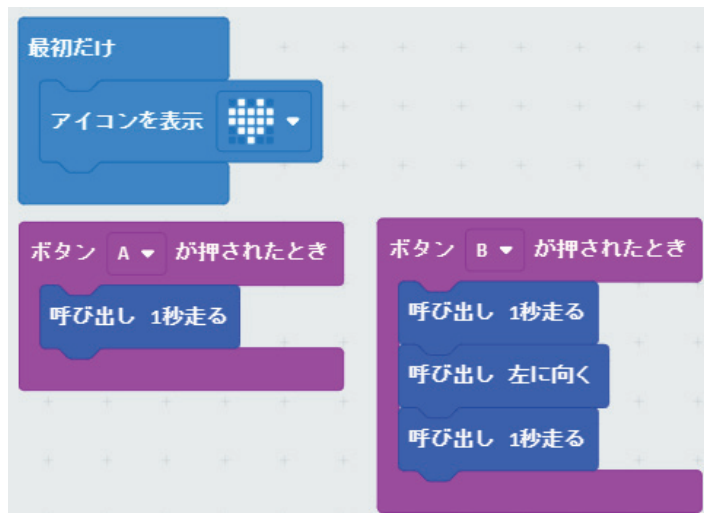


## プログラム

サンプルプログラムを用意しました。4・5 ページの URL にアクセスすると、カーリングゲームに必要な関数（ブロック）が作成済みのプログラムが入手できます。バンパースイッチが押されたときにロボットが1秒間直進するプログラムがあらかじめ作られていますが、ゲームの公平性を保つためにこの部分は変更しないようにしてください。間違えて変更しないよう、関数とバンパーのプログラムはすぐには見えないところに配置してあります。サンプルプログラムのプログラミングエリアを左右に移動すると下図のようなプログラムがあります。「ずっと」の中身がバンパースイッチが押されたときの動作です。バンパースイッチ P1 と P2 どちらが押されても同じ動作をします。



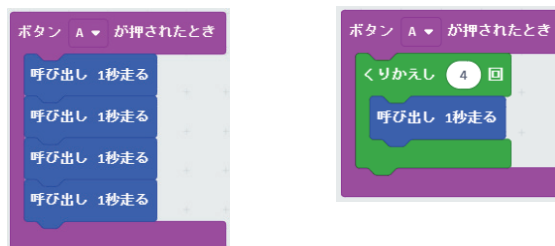
変更してはいけない部分



小学校低学年を対象にしたプログラムです。使える関数は移動用の「1秒走る」、「左に向く」、「右に向く」の3種類に加えて、調整用の「まっすぐ走るようにする」が4種類あります。

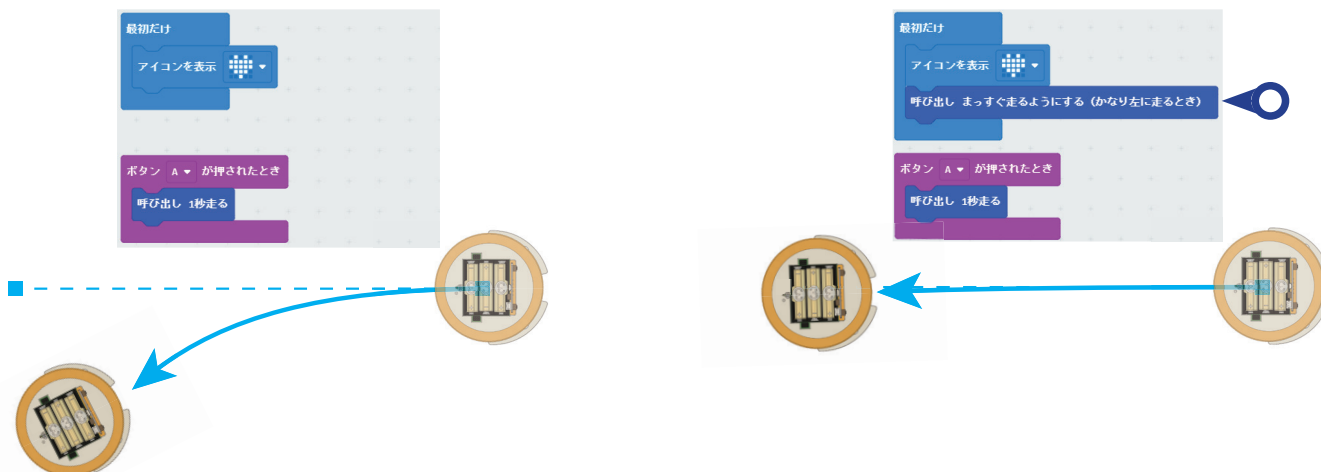
サンプルプログラムにはボタンA、Bを押したときの動作の例が作られています。それぞれのブロックの並びと、実際のロボットの動作を見比べてどのようにすれば思い通りにロボットを動かせるかの参考にしてください。

例えば4秒間ロボットを走らせる場合には「1秒走る」ブロックを4つつなげるかループで4回実行してください。1秒間でどれだけ進むかは電池の残量などによっても変化し、正確に円の中心を狙うのは難しいので、ロボットのスタート位置や向きを工夫して挑戦してください。

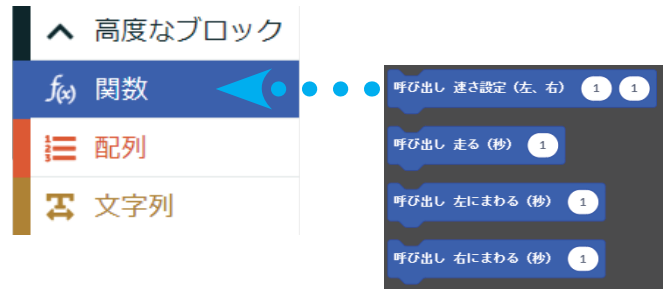
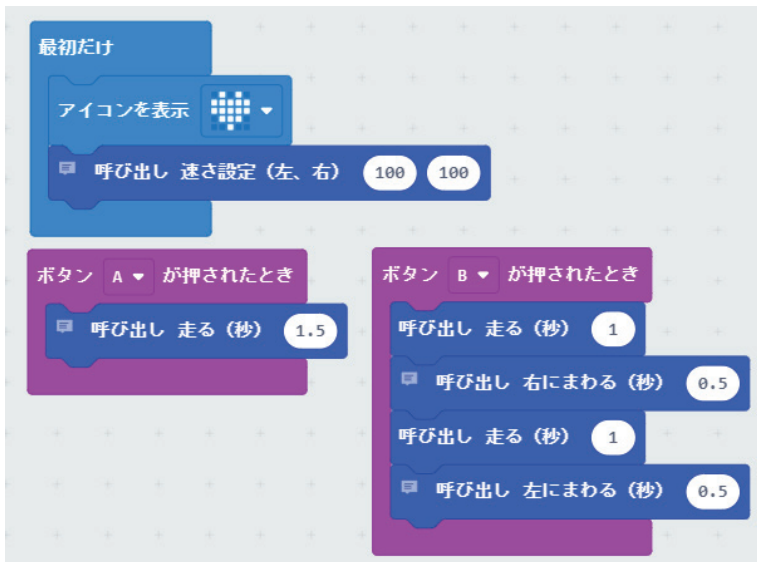


「1秒走る」を実行した際、ロボットが真っすぐには走らず徐々に左右どちらかにそれていってしまうことがあります。これは左右のモーターの個体差によるものです。真っすぐに走らせたい場合は下図のように調整してください。調整は曲がりの程度に合わせて2段階で可能ですが、微調整はできませんので期待通りに調整が効かない可能性があります。中級のプログラムでは具体的に数値を入力して微調整が可能です。

## かなり左に曲がってしまう → まっすぐに走るようにする



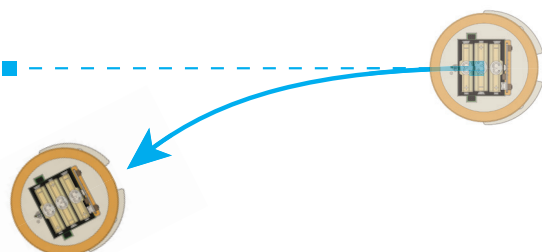




小学校高学年以上を対象にしたプログラムです。使える関数は「走る (秒)」「右にまわる (秒)」「左にまわる (秒)」の3種類に加えて、調整用のブロックが1種類です。走る時間、回転する時間 (秒) を数値で指定可能です。細かい時間を調整することで、狙ったルートでロボットを走行させることができます。サンプルプログラムにはボタン A、B を押したときの動作の例が作られています。それぞれのブロックの並びと、実際のロボットの動作を見比べてどのようにすれば思い通りにロボットを動かせるかの参考にしてください。

ロボットの左右のモーターには個体差が存在するため、直進させるつもりが左右にそれてしまうことがあります。その場合「速さ設定 (左、右)」ブロックを使って左右のモーターの速さのバランスを調整して、直進に近づけることができます。設定できる値の範囲は-100~100です。例えばカーリングロボットが左にそれていってしまう場合、右モーターが左よりも速いという事ですので、右モーターを遅くします。(100が最大ですので、左をより速くすることはできません。) 速さの数値をマイナスにするとモーターの回転方向が変わります。

**かなり左に曲がってしまう (右モーターが速い) -----> まっすぐに走るようにする (右モーターを遅くする)**



# ゲームの進め方

ここでは micro:bit の基礎的な使い方は参加者が理解している前提で解説をすすめます。 所要時間は目安です。

## ステップ① 動作確認 所要時間 5分

初級中級どちらもまずサンプルプログラムを micro:bit にダウンロードしてロボットを動かしてみましょう。まずはロボットの電源スイッチを ON にします。micro:bit のボタン A、B を押してロボットが動くことを確認し、その動きがプログラムの通りになっている事確かめます。

バンパースイッチに触れた場合もロボットは動いてしまいますので、「ボタンを押したら動いた」ということを必ず確認してください。

## ステップ② 中心を目指そう 所要時間 15分

カーリングコートのスタートエリアからハウスの中心までは 100cm です。まずは基本戦略として中心を狙う必要がありますので、この練習を行います。参加者が多い場合、カーリングコート以外にも 100cm の基準となる練習コースがあると便利です。

まずは直進だけさせて円の中心で止めるプログラムを作ります。初級ではピッタリ止めることはできないと思いますが、中級では走る時間を微調整してハウスの中心を目指します。また、直進せず大きくカーブしてしまう場合はここで調整します。

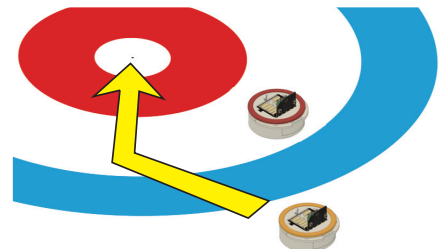
## ステップ③ 練習試合 所要時間 10分

時間に余裕がない場合、このステップは省略します。ステップ②で作った直進プログラムだけでゲームをやってみます。実際にゲームを行う事で、どのような動きをすれば有利にゲームを進められるかを考えます。

## ステップ④ 色々なルートを考える 所要時間 15分

ゲームを行うと様々な状況が発生します。

- ・直接ハウスの中央を狙いたい
- ・押されてハウスの中心に近づくようにあえて手前で止めたい
- ・すでにコート上にあるロボットをよけてハウスの中心を狙いたい
- ・相手のロボットをはじき出したい
- ・相手のロボットが中心を狙えないように邪魔したい



ロボットを動かす番になった時に、その状況に応じたプログラムを作るのが理想的ではありますが、時間がかかりすぎてしまいますのであらかじめ状況を想定して動作を考え、ボタン A・B で実行するプログラムを作成します。チーム内でそれぞれのロボットの動作を共有し、状況に応じて適切なロボットを動かすのも面白いかもしれません。

## ステップ⑤ ゲーム 所要時間 10分

交互にロボットを走らせて、得点を競います。走らせる順番は事前に決めておき、状況に応じて順番を変えてもよいというルールにすると時間もかからずおすすめです。

## ステップ⑥ プログラム修正 所要時間 10分

ゲームをやってみてうまくいかなかったところの修正、新しく思いついたアイデアの挑戦をしてみましょう。

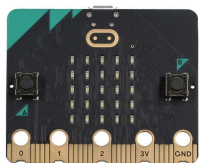
## ステップ⑦ ゲームの繰り返し

ステップ⑤、⑥を繰り返してゲームを進めます。

# 製品情報

## micro:bit v2.2

---



<https://sedu.link/microbit>

プログラミング教育向けのマイコンボードです。  
ロボットベースには micro:bit が含まれておりません。

## micro:bit 用ロボットベース（接触スイッチ）

---

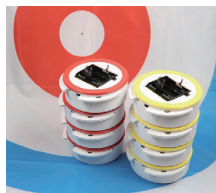


<https://sedu.link/mb-rb-switch>

micro:bit を接続してコントロールできる接触スイッチ（バンパースイッチ）付きのロボットです。色分けリングはついていません。

## プログラミングカーリングで遊ぼうセット

---



<https://sedu.link/mb-curling>

色分けリング付きのロボットベース 8 台（色分けリング付き）とカーリングコートのセットです。

## カーリングコート

---



<https://sedu.link/curling-court>

カーリングコート単体の商品です。練習用のコートが欲しいときなどにご利用ください。