



1. 傾けるだけでMaqueenを操縦する！

「ロゴ」はココ



h:bitの傾きセンサーだけを使って、動きをプログラムします。

```

when green flag clicked
  forever
    if [ロゴが下になった] then
      [無線で送信 D = 1 v] [n:bitを向こう側へ倒すと、前へすすむ]
    end
    if [左に傾けた] then
      [無線で送信 D = 2 v] [左に傾けると、ゆっくり左にまがる]
    end
    if [右に傾けた] then
      [無線で送信 D = 3 v] [右に傾けると、ゆっくり右にまがる]
    end
    if [ロゴが上になった] then
      [無線で送信 D = 4 v] [手前に起こすと、後ろへすすむ]
    end
    if [画面が上になった] then
      [無線で送信 D = 0 v] [なにもしなければ、とまれ]
    end
  end

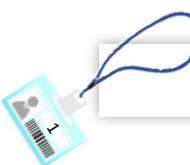
```

NEXT DAY
http://www.nextday.jp



ボタンだけのプログラムよりも、多くの動きができます。

※「最初だけ」ブロックも必要です
テキストのプログラムと同じものを使います



2. ボタンと傾きでMaqueenを操縦する！



```

when green flag clicked
  forever
    if [button A pressed v] then
      if [button B pressed v] then
        send "D" (9) via radio
        if [tilt (down) v] then
          move (9) steps
        else
          move (1) steps
      else
        send "D" (4) via radio
        if [tilt (up) v] then
          move (4) steps
        else
          move (1) steps
    else
      if [button B pressed v] then
        send "D" (2) via radio
      else
        send "D" (3) via radio
    end
  end
end
  
```

ボタンと傾きセンサーを組み合わせることで、さらに多くの動きをプログラムできます。

ボタンAとBが“同時に押されている時に、さらにn:bitを傾けることで”、3種類の動きをするようにします。

1. ゆっくり前に進む (n:bitを水平にする)
2. 最高速で前に進む (n:bitを前に倒す)
3. バックする (n:bitを手前に起こす)

コンピュータ（機械）に人間の意思を伝える設計のことを、「マンマシンインターフェース（HMI）」といいます。

人が“なにかを操作したときには、機械が人の感覚に合った動きになる”ようにします。

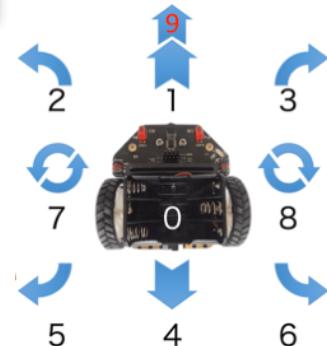
こうして、操作ミスや事故が起きないようなつくりにすることが、プログラミングの技なのです。

※「最初だけ」ブロックも必要です
テキストのプログラムと同じものを使います



3. ボタンと傾きでMaqueenを操縦！

チャレンジ2と同じ動きをするプログラムです。
ブロックの作り方が異なっています。
さらに、いまどの方向に動かしているのか、画面の
LEDに表示する機能を追加しました。



最初だけ

```
無線のグループを設定 5
変数 command を 1 にする
```

ずっと

```
アイコンを表示
数を表示 command
```

画面のLEDに方向を表示します。「ずっと」ブロックを
ふたつ使うことで、「並列
処理」ができます。

ずっと

```
もし ボタン A+B が押されている なら
無線で送信 "D" = command
```

```
でなければもし ボタン A が押されている なら (-
無線で送信 "D" = 7
```

```
でなければもし ボタン B が押されている なら (-
無線で送信 "D" = 8
```

```
でなければ
無線で送信 "D" = 0
```

画面が上になった とき

```
変数 command を 1 にする
```

ロゴが下になった とき

```
変数 command を 9 にする
```

ロゴが上になった とき

```
変数 command を 4 にする
```

チャレンジ



4. 全方向フル操縦！！

チャレンジ3のプログラムを改良して、0から9の動きをすべて行います！

最初だけ

無線のグループを設定 5
変数 AB_forward を 1 にする
変数 A_left を 2 にする
変数 B_right を 3 にする

ずっと

アイコンを表示
数を表示 command

左に傾けた とき

変数 A_left を 7 にする

ずっと

もし ボタン A+B が押されている なら
変数 command を AB_forward にする
でなければもし ボタン A が押されている なら (−)
変数 command を A_left にする
でなければもし ボタン B が押されている なら (−)
変数 command を B_right にする
でなければ
変数 command を 0 にする
無線で送信 "D" = command

[無線で送信] ブロックは、
1個だけ！

右に傾けた とき

変数 B_right を 8 にする

画面が上になった とき

変数 AB_forward を 1 にする
変数 A_left を 2 にする
変数 B_right を 3 にする

ロゴが下になった とき

変数 AB_forward を 9 にする
変数 A_left を 7 にする
変数 B_right を 8 にする

ロゴが上になった とき

変数 AB_forward を 4 にする
変数 A_left を 5 にする
変数 B_right を 6 にする