

PIECE プログラミングモジュール

やってみようプログラミング

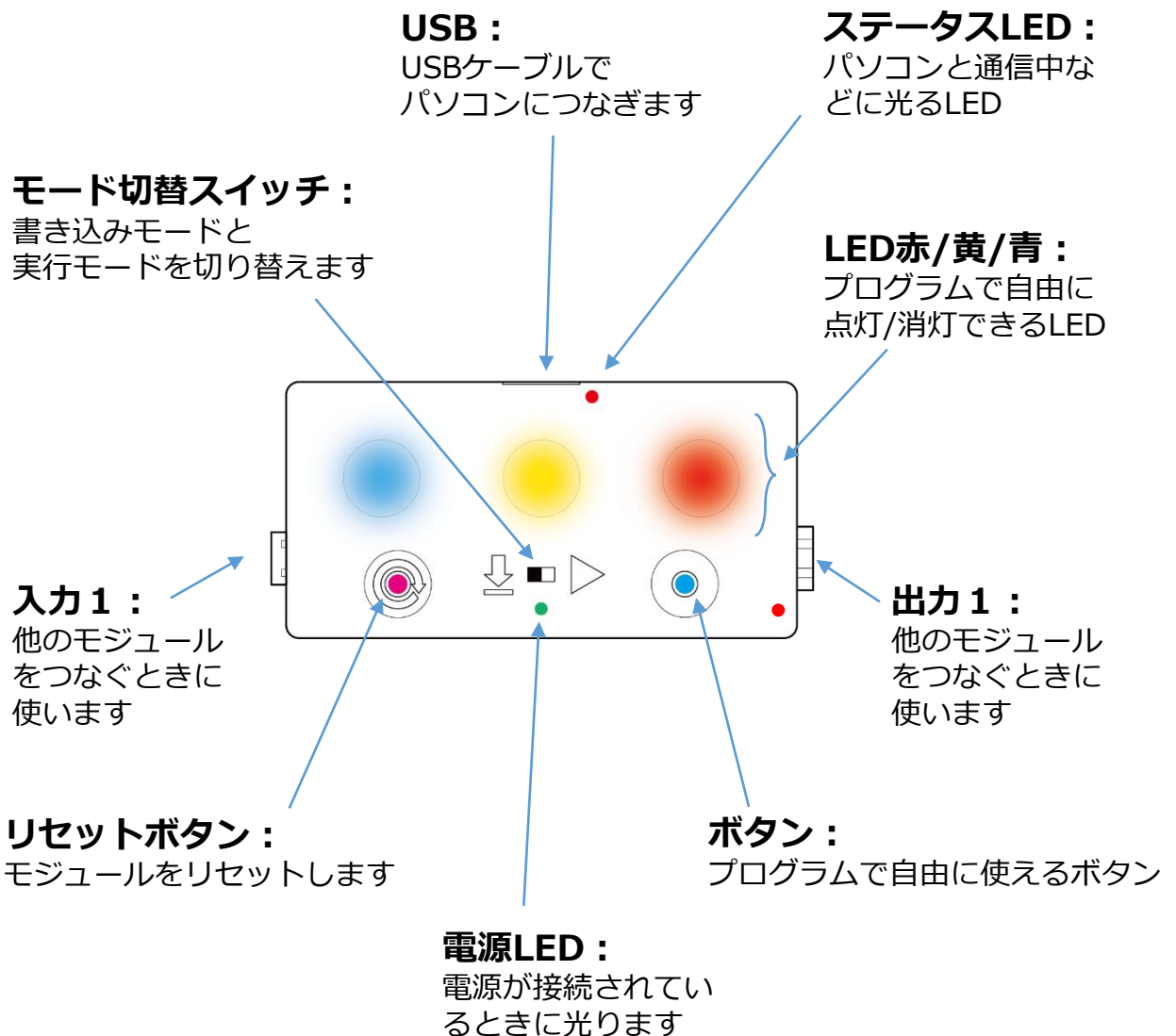


1. プログラミングモジュールを知ろう
2. プログラミングをはじめよう
3. プログラムをつくろう
4. センサーライトをつくろう
5. いろんな動きに挑戦

| | | | |
|---|---|---|----|
| 年 | 組 | 番 | 名前 |
|---|---|---|----|

1. プログラミングモジュールを知ろう

各部のなまえ



プログラミングとはプログラムをつくること

今日の予定、旅行の日程や夏休みの計画など、あることを行うために必要な段取りを順番に書き出したものをプログラムといいます。

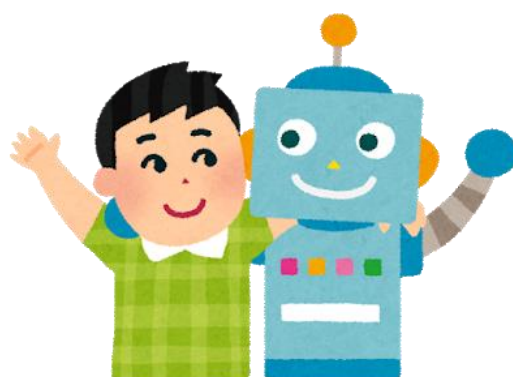
コンピューターにさせたいプログラムを作成することをプログラミングといいます。

プログラミングってなぜ必要？

コンピューターにプログラミングするときにはコンピューターにわかる言葉で書く必要があります。これは外国の人とお話するときと似ています。外国の人とお話するときはその人が分かる言葉、つまり外国語で話さないと通じませんよね。コンピューターなどの機械と話すときはコンピューターが理解できる言葉でやってほしい命令を書かなくてはなりません。今回は、PIECE用のプログラミングソフトを使って動きの命令を機械に伝えていきます。



外国の人と話すための「外国語」



キカイと話すための「プログラミング」

2. プログラミングをはじめよう

プログラミングの準備



ブラウザーには Google Chrome または Microsoft Edge を使います。

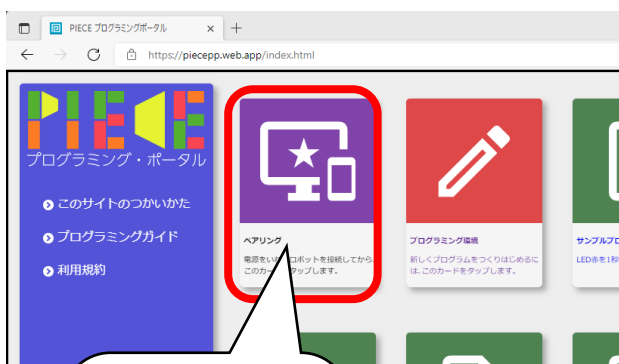


PIECEプログラミングポータルにアクセスする

- 1 ブラウザーのアドレスバーに下記URLを入力してサイトに移動します。

piecepp.web.app

- 2 PIECEプログラミングポータルのページが開きます。



ペアリング
電源をいれたロボットを接続してから、
このカードをタップします。



ペアリングのカードがあるときは、

- 3 へ進みます。



ペアリングのカードがないときは、

- 4 へ進みます

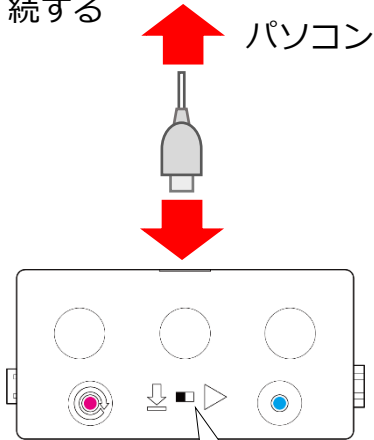
3 ペアリングする



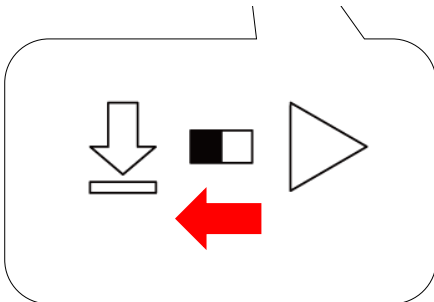
PIECEプログラミングモジュールを始めてパソコンにつないだときは「ペアリング」をします。ペアリングは最初に一回だけで、次に使うときは必要ありません。

※使用するブラウザが変わったときや、ブラウザを使用するログインユーザーが変わったときにはペアリングが必要なことがあります。

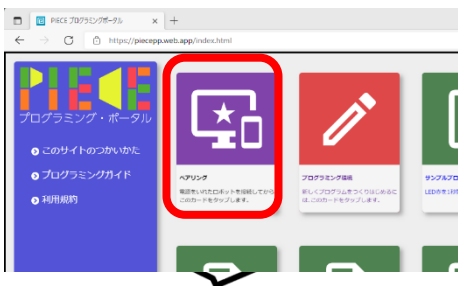
- ① プログラミングモジュールとパソコンをUSBケーブルで接続する



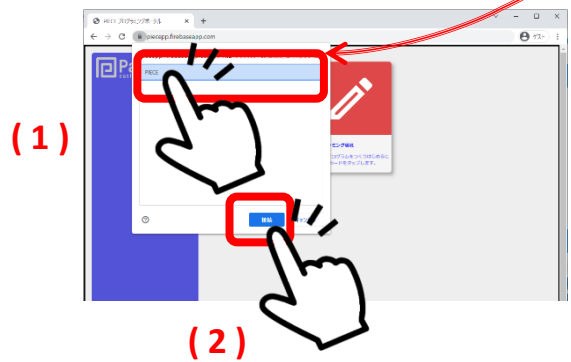
- ② モード切替スイッチを左にする



- ③ 「ペアリング」をタップ



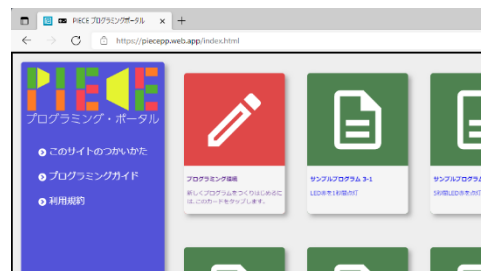
- ④ (1) PIECE をタップして選ぶ
(2) **接続** を押す



- ⑤ **OK** をタップ

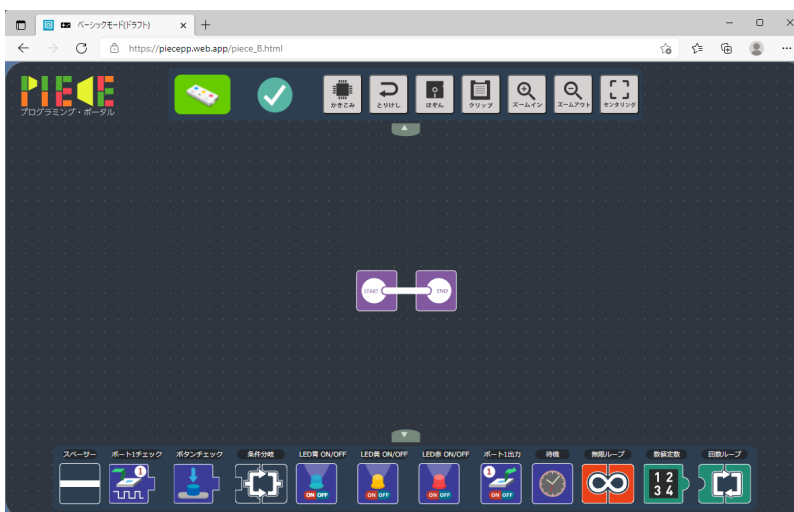
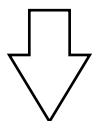


- ⑥ ペアリング終了



4 プログラム編集画面を開く

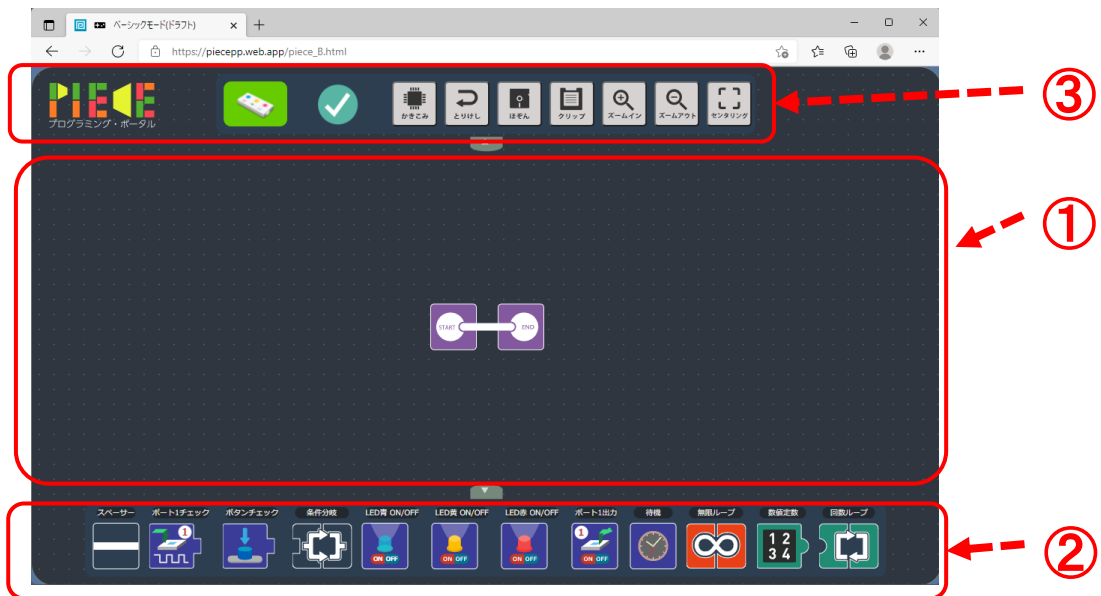
プログラムを新しく作るときは、プログラミングポータル画面の、プログラミング環境のカードをタップします。



プログラム編集画面が開きます。
さあ、プログラミングをはじめよう！

編集画面の説明

プログラム編集画面



①ワークスペース

この場所にプログラムを作成します。

スタートとエンドのアイコンはあらかじめ配置されています。

②ドック

プログラムで使用する命令アイコンが並んでいます。

LEDをオンオフするアイコンや、プログラミングモジュールの入力や出力を制御するためのアイコン、分岐やループのアイコンなどが用意されています。

③スイッチボード

プログラムの情報を表示したり、プログラムの保存や、プログラムの書き込みなどの操作をするためのアイコンが並んでいます。

練習：うごかしてみよう

命令アイコンを画面下のドックからドラッグして、スタートとエンドの間の置きたい場所で離すとプログラム上にアイコンが配置されます。



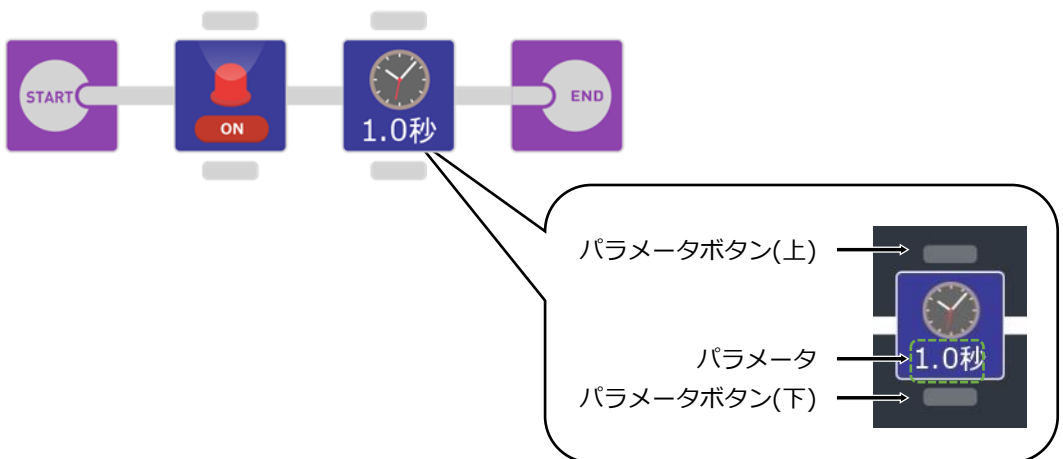
練習：

「赤LEDを1秒間点灯する」

下図のようなプログラムをつくります。

待機アイコンの秒数を変えるときはアイコンの上下についているパラメータボタン をクリックします。

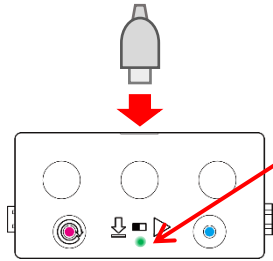
または、パラメータを短くタップしてキーボードで値を直接入力することもできます。



プログラムを書き込もう

プログラムができれば、プログラムを書き込みましょう。

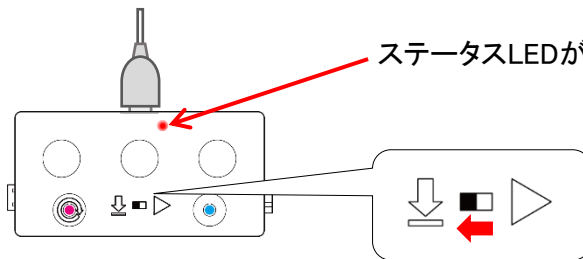
①プログラミングモジュールとパソコンをUSBケーブルで接続します。



電源表示LEDが点灯します。

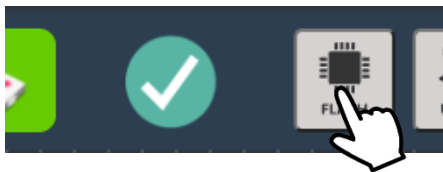
充電専用USBケーブルでは書き込みできません。
付属のUSBケーブルを使います。

②プログラミングモジュールのモード切替スイッチを左側にスライドして書き込みモードにします。




ステータスLEDが点灯することを確認します。

③FLASH(フラッシュ)ボタンをタップしてプログラムを書き込みます。



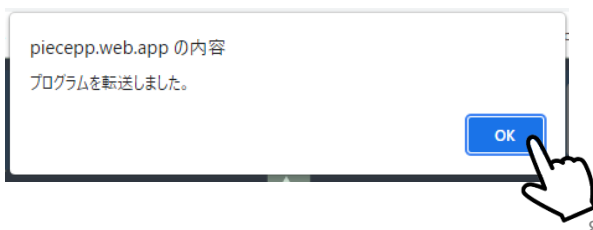
チェックマーク  が表示されているときにプログラムを書き込むことができます。



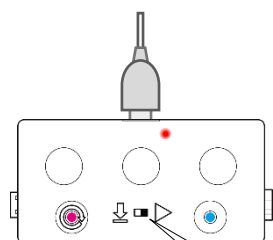
未接続マーク  が表示されているときはプログラムを書き込むことができません。
ケーブルの接続とモード切替スイッチの位置を確認し、**プログラミングモジュールのリセットスイッチを一回押してみます。**

④画面にプログラムを転送しましたと表示されれば書き込み完了です。

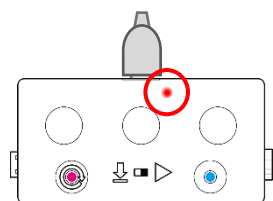
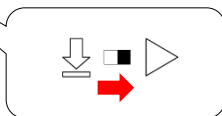
OKをタップするとメッセージが消えます。



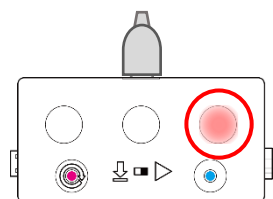
プログラムを実行しよう



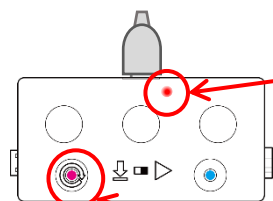
モード切替スイッチを右側にスライドして、実行モードにします。



実行モードにするとステータスLED赤が数回すばやく点滅したあとに、プログラムを実行します。



1 秒間LED赤が点灯！



プログラムが終了するとステータスLED赤がゆっくり点滅します。

もう一度プログラムを実行するときは、リセットスイッチを一回押します。



1 秒間LED赤が点灯しましたか？

3. プログラムをつくろう

ここでは、いくつか問題をときながらプログラムの方法や考え方を知りましょう。

問題：

「青色→黄色→赤色の順番でLEDが1秒ずつ点灯して消える」

3つのLEDが順番に1つずつ点灯するプログラムをつくってみましょう。1つ点灯させたらそのLEDを消して、次のLEDを点灯させるようにします。順番に点灯していくように見せるために、LEDが点灯して次のLEDが点灯するまでに1秒間待つことにします。

答え：



モジュールに書き込んで動きを確認してみましょう。



順次処理

プログラムがスタートすると、命令された順番どおりに命令を実行していきます。途中の命令をとばしたり、逆の順番で実行することはありません。このことを順次処理といいます。

問題：

「青色LEDと赤色LEDが交互に1秒ずつ点灯する」動作をずっとくりかえす。

青と赤のLEDが交互に点灯するプログラムをつくってみましょう。青LEDを点灯させて1秒後に青LEDを消灯します。同時に赤LEDを点灯させて1秒後に赤LEDを消灯するという動作をずっとくりかえすようにします。

ずっとくりかえすために、無限ループアイコンを使います。

無限ループ

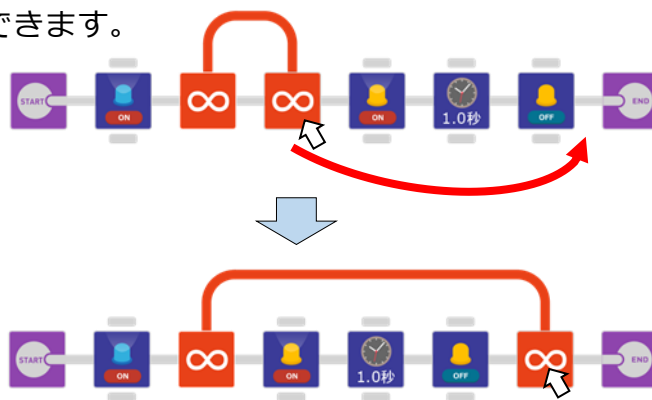


無限ループアイコンで囲まれた範囲をずっとくり返します。

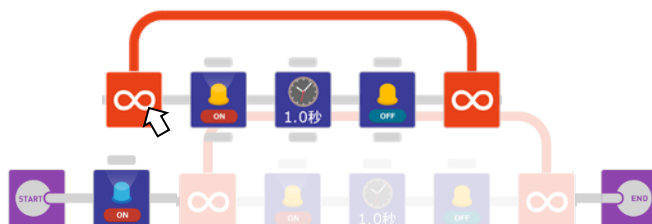


ループの操作方法

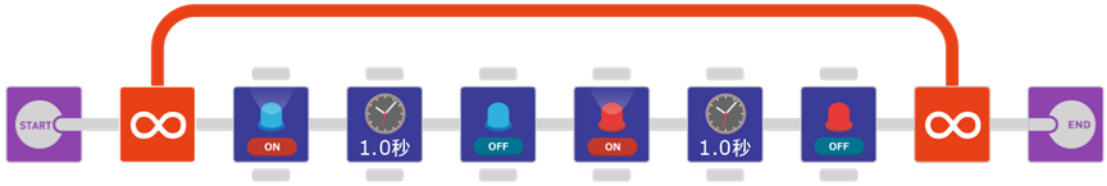
無限ループアイコンの後ろ側のアイコンをドラッグして囲む範囲を変えることができます。



アイコンの先頭側をドラッグすると無限アイコンで囲まれている範囲を一緒に移動することができます。



答え：



ポイント

動作をずっとくりかえすには、無限アイコンをつかいます。

青LEDと赤LEDを交互に点滅させるプログラムを無限アイコンで囲みます。



モジュールに書き込んで動きを確認してみましょう。

問題：

「青色LEDと赤色LEDが交互に1秒ずつ点灯する」動作を5回くりかえす。

無限にくり返すのではなく、決まった回数をくりかえすプログラムをつくってみましょう。

回数を決めてくりかえすには、回数ループと数値定数アイコンをつかいます。

回数ループ



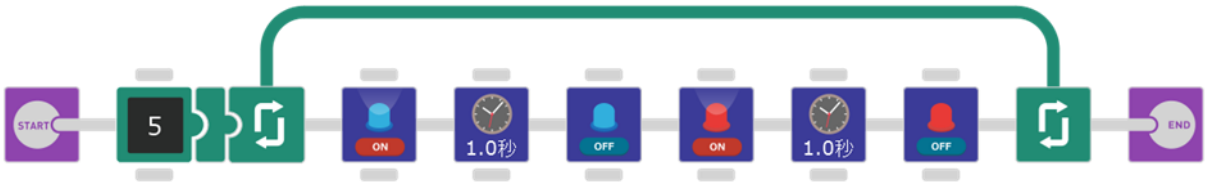
回数ループアイコンで囲まれている範囲を左側につないだアイコンで決められた数字の回数くりかえします。

数値定数



上下のボタンで数値を設定します。
またはパラメータの短くタップして値を直接入力します。

答え：



ポイント

決められた回数をずっとくりかえすには、回数ループと数値定数アイコンをつかいます。

LEDを交互に点滅させるプログラムを回数ループアイコンで囲みます。

5回くりかえしたらループから抜けて先に進み、ENDアイコンに到着するので、プログラムが終わりLEDが消えます。



モジュールに書き込んで動きを確認してみましょう。



くりかえし

同じ動作をくりかえし行うときに、何度も何度も同じプログラムを書いていくのは大変ですし、間違いが起きやすくなります。同じ動作を何度も処理したいときは、回数ループアイコンや無限ループアイコンを使ってくりかえします。同じ動作を何度も行う処理のことを、「くりかえし」といいます。

問題：

「ボタンを押している間は黄色LEDが点灯して、ボタンを押していないときは黄色LEDは消灯している。」

PIECEプログラミングモジュールにはボタンがついています。このボタンを使って、押しているときと押していないときで動きが変わるプログラムをつくってみましょう。

ボタンが押されたかどうかをチェックして動きかたを変えるにはボタンアイコンと条件分岐アイコンをつかいます。

ボタン



ボタンをチェックするときにつかいます。押されていて右側のアイコンに「○」を押されていないときは「×」を伝えます。

条件分岐



左側につながったアイコンから「○」を受け取ったら上側の線へ、「×」を受け取ったら下側の線へ進みます。

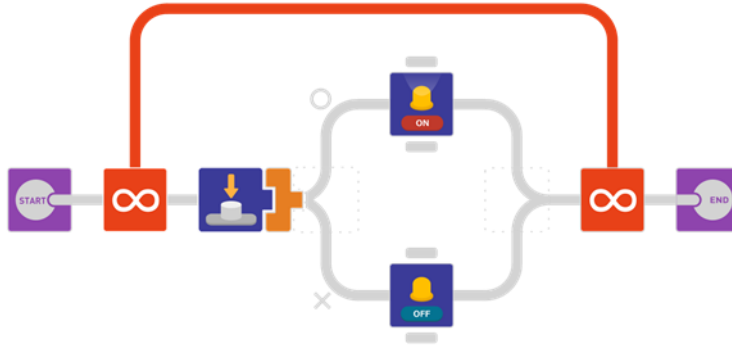


ビックリマーク



条件分岐アイコンの左側には、必ず「○」「×」を伝えるアイコンをつながないといけません。もしも違うアイコンをおいてしまうと、プログラムの文法が間違っていること示すビックリマークが表示されます。プログラムが完成したときにビックリマークがないようにしましょう。

答え：



ポイント

ボタンが押されているときは分岐アイコンの上側に進むので、上側には黄LEDオンを、下側にはLEDオフを置きます。

ボタンはいつ押されるかわかりませんので、ずっとチェックする必要があります。ずっとチェックするために無限ループをつかいます。



モジュールに書き込んで動きを確認してみましょう。



条件分岐

人は道を歩いているときに、目の前に急にボールがとびだしてきたら、すぐに止まることができますよね。これは「ボールがとびだしてきたら」止まる、「ボールがないときは」歩く。というように条件によって動きをかえているということです。

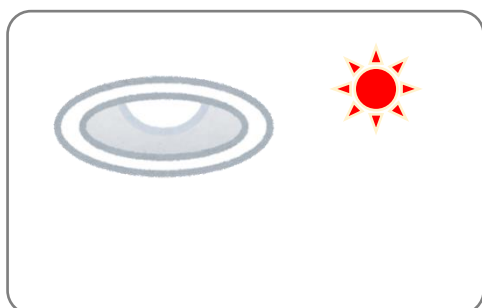
このように条件によって動きをかえることを、条件分岐といいます。

4. センサーライトをつくろう

光センサー（LIGHTモジュール）を使って、

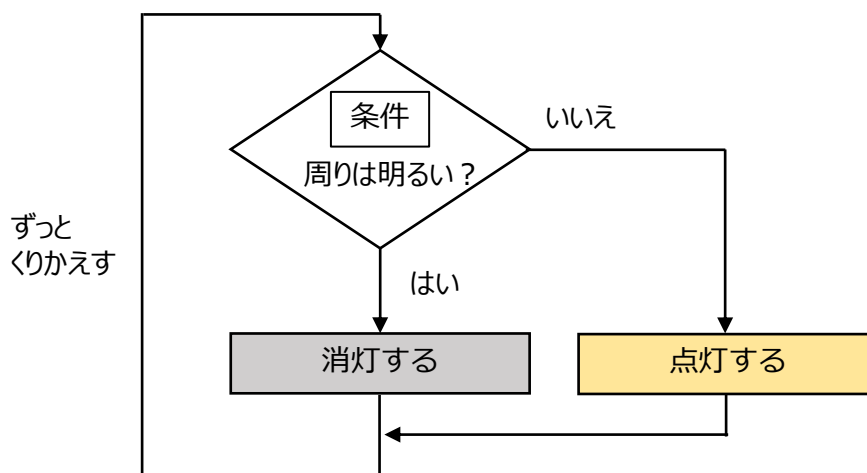
周りが暗いときは自動で点灯して明るくなると自動で消灯する センサーライト

をつくってみましょう。自動で消灯するので、消し忘れがなく無駄な消費電力を押さえることができます。



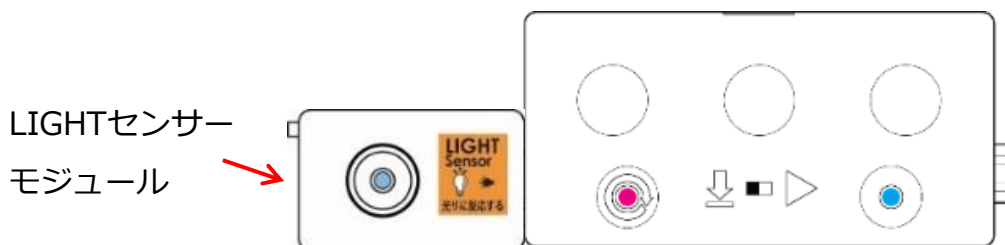
※ここではライトに黄色LEDを点灯させることにします。もしも別売のTORCHモジュールをもっていれば、それを使うとより明るいセンサーライトをつくることができますよ。

このセンサーライトの動作を図で表すと以下のようになります。



☑ 準備しよう

プログラミングモジュールの入力1に下図のように**LIGHTセンサーモジュール**をつなぎます。



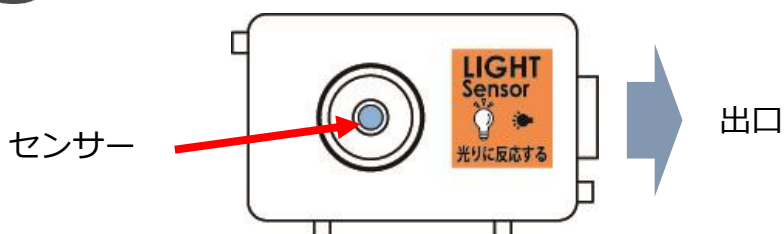
ポート1チェック



プログラミングモジュールの入力1に接続したモジュールの状態を調べる際には、ポート1チェックをつかいます。入力1につないだモジュールがオン信号を出しているときは[○]に、信号をだしていないときは[×]になります。



LIGHT センサーモジュール

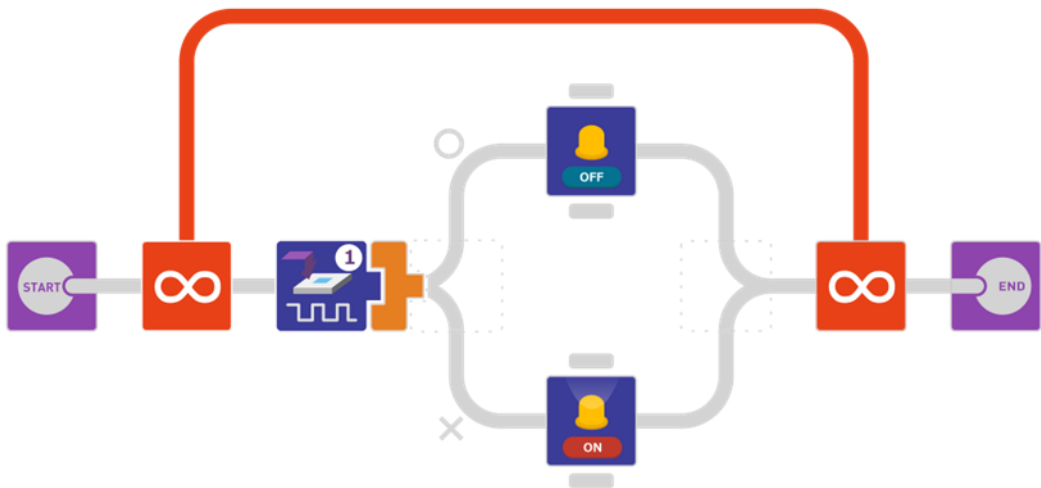


センサーが光を感じると出口から信号が出る

ポイント

- ☑ “明るさ”をしらべるには、**LIGHTセンサーモジュール**をつかいます。
- ☑ LIGHTセンサーの状態をチェックするときは**ポート1チェック**をつかいます。
- ☑ 「明るいとき」と「暗いとき」の条件によって動きかたが変わるので**条件分岐アイコン**をつかいます。
- ☑ “明るさ”はずっとチェックしたいので、**無限ループアイコン**をつかいます。
- ☑ ライトには、青/黄/赤LEDのどれか、または全部をつかうといいでしょう。

答え：



モジュールに書き込んで動きを確認してみましょう。

まとめてみよう

コンピューターとプログラミング

コンピューターにさせたいことを順番に書き出したものをプログラムといい、プログラムをつくることを（ ）といえます。

（ ）： プログラムは最初からアイコンを並べた通りに順番に実行する。

（ ）： 同じ動作を指定した回数またはずっと実行する。

（ ）： 条件によって動きかたを変える。

身の回りのプログラミング

夜だけ光るライトには（ ）センサーがついていて、このセンサーを使って、暗いときにライトを（ ）、明るいときにライトを（ ）ように動きます。

センサーを使うと（ ）でオンやオフできるようになり、電気を（ ）使うことができるようになります。

条件分岐 プログラム 自動 プログラミング くりかえし
明るさ センサー 効率的に つける 消す 順次処理

5. いろいろな動きに挑戦

センサーを取り付ける場所をイメージして、さらにボタン操作を組み合わせれば、もっと楽しく使いやすい動きをつくることができるかもしれません。使われている状態を想像しながらプログラムしてみましょう。

こんなプログラムをつくってみよう

暗くなるとしばらくの間点灯する

暗いときにボタンを押すと、しばらくの間点灯する

暗くなると自動で点灯して、明るくなったときにボタンを押して消灯する

明るいときは赤LEDが点滅していて、暗くなると黄LEDが点灯する