

中学生

小学生

考える力を伸ばす

プログラミング教室

ベーシックコース

別冊



爆弾イライラ棒

1回目： 年 月 日

2回目： 年 月 日

名前：

どんなゲームを作るのかな？

「爆弾イライラ棒_完成.sb3」を確認しよう！

最初に、完成品のゲームを試してみます。どんなゲームを今から作るかを確認してみましょう。

操作手順1 ゲームをスタートします。

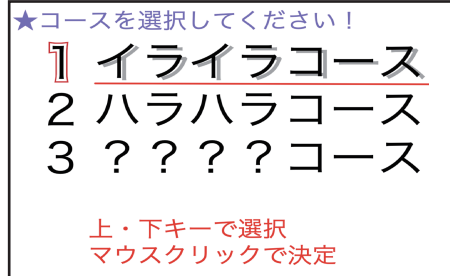
ボタンを押して、
ゲームをスタートします。



画面をクリック

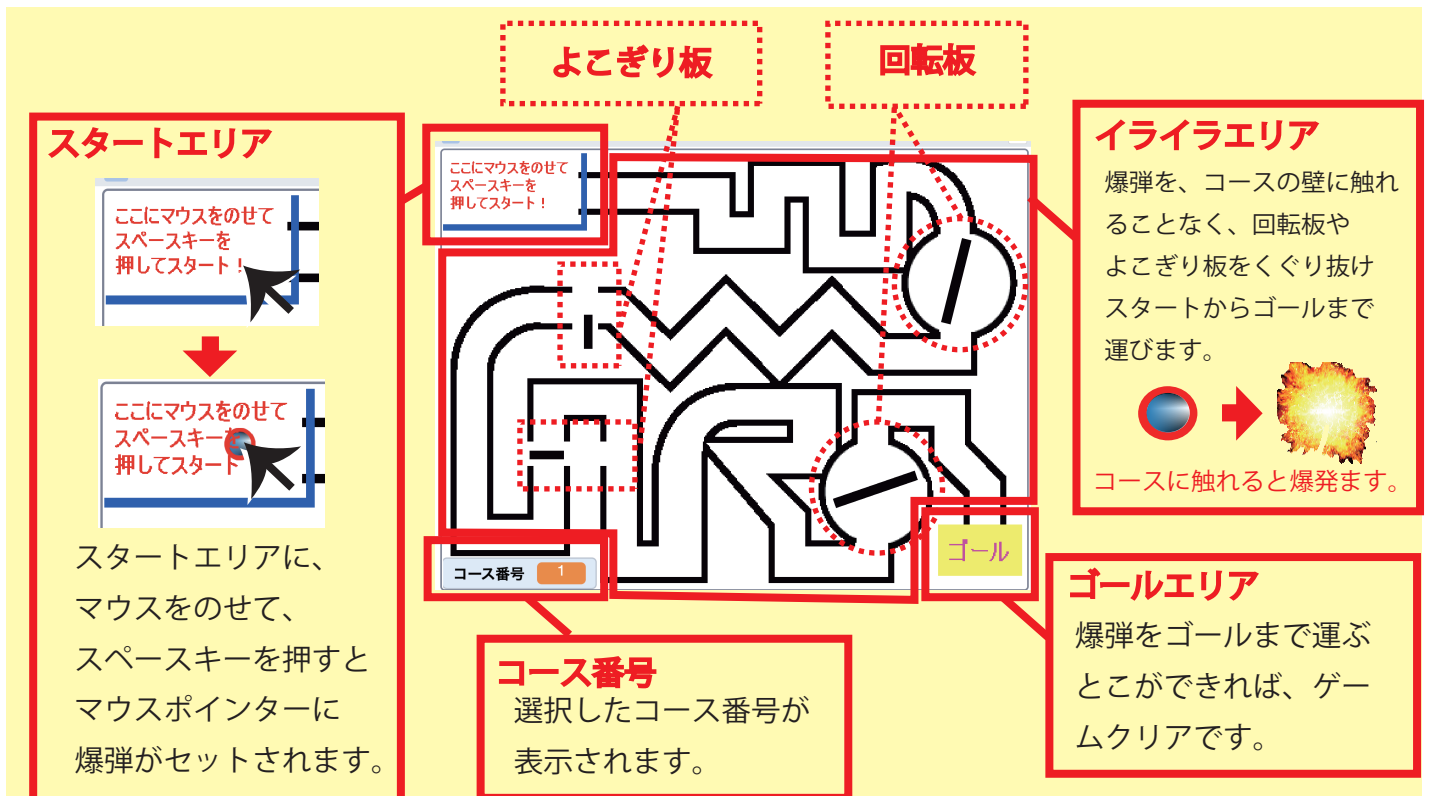
操作手順2 コースを選択します。

右・左キーでコースの選択を、
1～3の中から選びます。



上下キーでコースを選択し、画面をクリック

操作手順3 マウスポインターにセットされた爆弾を、イライラエリアの壁を触らずに慎重に操作して、スタートエリアからゴールエリアまで運びます。



ゴールまでたどり着くと
ゲームクリア

ゲームクリア

爆発すると
ゲームオーバー

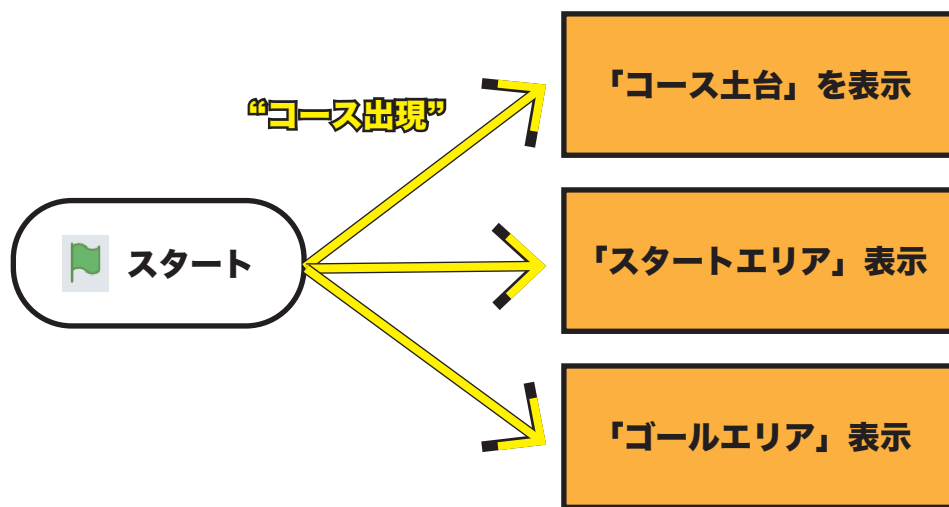
ゲームオーバー

プログラムを作っていこう！

1 コースを表示させよう

ステージで全体の流れを作ります。

プログラムを実行すると“コース出現”を送り、それぞれのコースのパーツを表示させます。



1 ステージのプログラムを作ります

ステージでは、ゲームに必要なパラメーターを設定し、“コース出現”をプログラム全体に送ります。



ゲームがスタートしたとき

- ・“コース番号”変数を1にします。
- ・“コース出現”メッセージを、プログラム全体に送ります。

2

「コース土台」のプログラムを作ります



「コース土台」では、選ばれたコースを表示させます。

```

    旗が押されたとき
    隠す

    コース出現を受け取ったとき
    最背面へ移動する
    表示する
    変数 コース番号 を表示する
    コスチュームを コース番号 にする
    
```

ゲームがスタートしたとき

- ・かくします。

“コース出現”メッセージが送られてきたとき

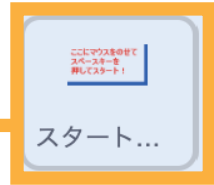
- ・他のスプライトの表示の一番下にします。
- ・表示させます。
- ・“コース番号”の内容を、画面上に表示します。
- ・コスチューム番号を“コース番号”の値のものにします。

確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・プログラムを実行して、ゲームをスタートします。
- “コース番号”変数は、画面上に表示しましたか？
- “コース番号”変数は、1になっていますか？
- 画面にコース1が表示されていますか？

3 「スタートエリア」のプログラムを作ります



コースを表示させるときに、「スタートエリア」も表示させます。

```

    旗が押されたとき
    隠す

    コース出現を受け取ったとき
    x座標を -143、y座標を 120 にする
    表示する
    
```

ゲームがスタートしたとき

- ・かくします。

“コース出現”メッセージが送られてきたとき

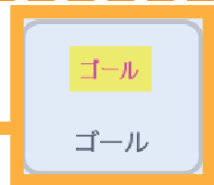
- ・位置を $(x, y) = (-143, 120)$ にします。
- ・表示します。

確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・スタートボタンを押して、ゲームを実行します。
- 「スタートエリア」は表示しましたか？

4 「ゴールエリア」のプログラムを作ります



コースを表示させるときに、「ゴールエリア」も表示させます。

```

    旗が押されたとき
    隠す

    コース出現を受け取ったとき
    x座標を 201、y座標を -148 にする
    表示する
    
```

ゲームがスタートしたとき

- ・かくします。

“コース出現”メッセージが送られてきたとき

- ・位置を $(x, y) = (201, -148)$ にします。
- ・表示します。

確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・スタートボタンを押して、ゲームを実行します。
- 「スタートゴール」は表示しましたか？

2 「爆弾」の基本を作ろう

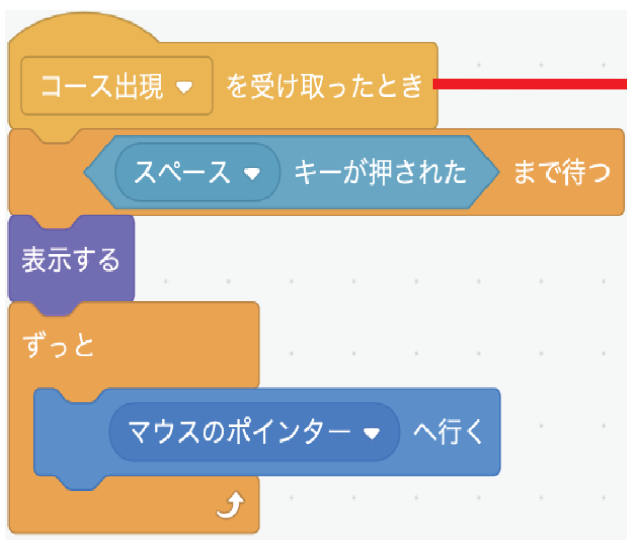
1 「爆弾」のプログラムを作ります

「爆弾」は、スペースキーが押されると表示され、マウスの動きについて来るようにします。



ゲームがスタートしたとき

- ・かくします。
- ・コスチュームを通常にします。
- ・大きさを40%にします。



“コース出現”メッセージが送られてきたとき

- ・スペースキーを押すまで待ちます。
- ・表示します。

- ・ずっと くりかえし ます。
- ・爆弾が、マウスポインターの位置にいきます。

(爆弾はずっとマウスポインターについていきます。)

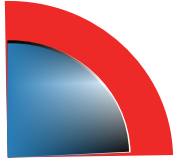
確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・スタートボタンを押して、ゲームを実行します。
- 画面上でスペースキーを押すと、爆弾が表示されましたか？
- 表示された爆弾は、マウスの動きについてきますか？


3 「爆弾」の色での制御を作ろう

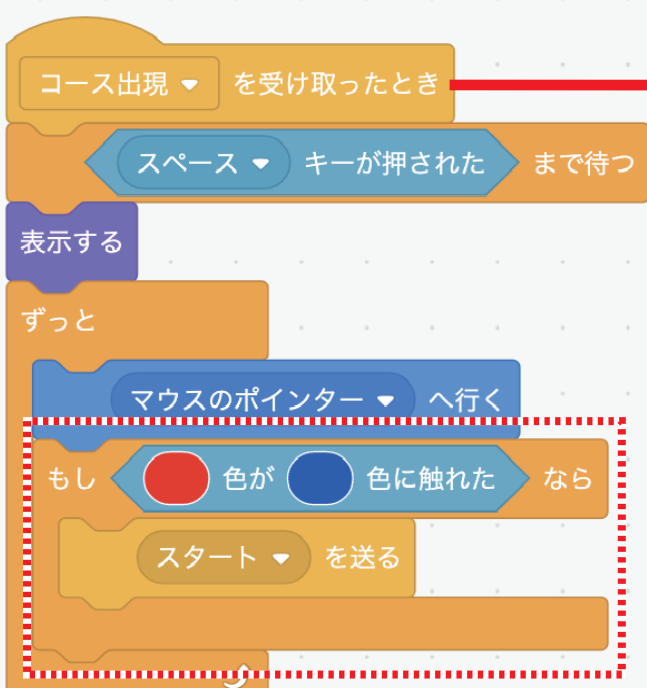
「爆弾」は、下記の表のように、触れる色により処理を行います。

判定方法 何が	何にふれた	そのときどうする？
 赤 爆弾のふちが	スタートエリアのラインにふれた 青 <small>ここにマウスをのせて スペースキーを 押してスタート！</small>	ゲーム開始
	ゴールエリアにふれた 黄 ゴール	ゲームクリア
	コースにふれた 黒	爆弾して ゲームオーバー

1 「爆弾」のプログラムを作ります



の部分を追加して、「爆弾」のスタートラインを割ったかを判断します。



“コース出現”メッセージが送られてきたとき

- ・スペースキーを押すまで待ちます。
- ・表示します。

・ずっと くりかえし ます。

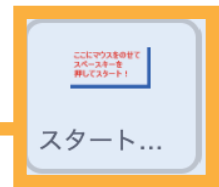
- ・爆弾が、マウスポインターの位置にいきます。

<もし、爆弾のふちがスタートエリアの
ラインにふれたら>

- ・“スタート”メッセージをプログラム
全体に送ります。

2 「スタートエリア」のプログラムを作ります

コースを表示させるときに、「スタートエリア」を表示させます。

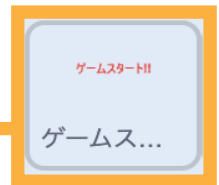


“スタート”メッセージが送られてきたとき

- ・かくします。

3 「ゲームスタート」のプログラムを作ります

「ゲームスタート」は、“スタート”メッセージを受け取り、文字を表示します。



ゲームがスタートしたとき

- ・かくします。

“スタート”メッセージが送られてきたとき

- ・位置を座標 $(x, y) = (0, 0)$ にします。
- ・1秒、表示します。
- ・かくします。

✓確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・スタート後、スタートエリアでスペースキーを押して「爆弾」を表示させます。

「爆弾」がスタートエリアの青の線に触ったとき、「スタートエリア」は隠れましたか？

「爆弾」がスタートエリアの青の線に触ったとき、「ゲームスタート」の文言が1秒、表示されましたか？

4 「爆弾」のプログラムを作ります



の部分を追加して、「爆弾」が「ゴールエリア」についたかを判断します。

```

    コース出現 ▼ を受け取ったとき
        スペース ▼ キーが押された まで待つ
        表示する
        ずっと
            マウスのポインター ▼ へ行く
            もし (赤) 色が (青) 色に触れた なら
                スタート ▼ を送る
            もし (赤) 色が (黄緑) 色に触れた なら
                ゲームクリア ▼ を送る
    
```

“コース出現”メッセージが送られてきたとき

- ・スペースキーを押すまで待ちます。
- ・表示します。

・ずっと くりかえし します。

・爆弾が、マウスポインターの位置にいきます。

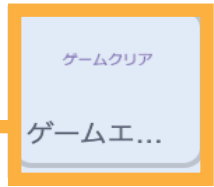
<もし、爆弾のふちがスタートエリアのラインにふれたら>

・“スタート”メッセージをプログラム全体に送ります。

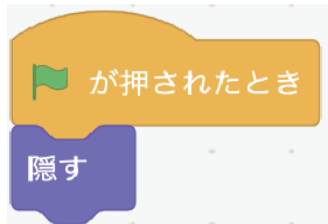
<もし、爆弾のふちがゴールエリアにふれたら>

・“ゲームクリア”メッセージをプログラム全体に送ります。

5 「ゲームエンド」のプログラムを作ります

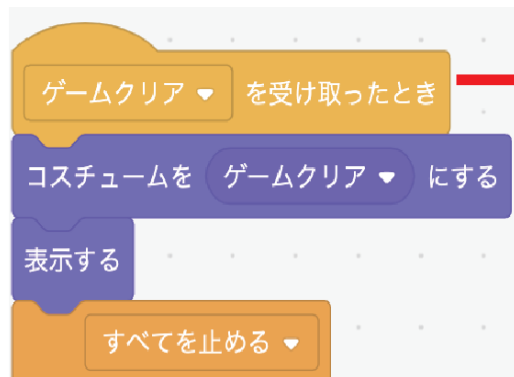


「爆弾」が「ゴールエリア」にたどり着いたとき、ゲームクリアを表示します。



ゲームがスタートしたとき

- ・かくします。



“ゲームクリア”メッセージが送られてきたとき

- ・コスチュームを“ゲームクリア”にします。
- ・表示します。
- ・プログラム全体を止めます。

確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・スタート後、スタートエリアでスペースキーを押して「爆弾」を表示させます。
- 「爆弾」がゴールエリアについたとき、“ゲームクリア”の文言は表示されましたか？

6 「爆弾」のプログラムを作ります



の部分を追加して、「爆弾」が「ゴールエリア」についたかを判断します。

```

    コース出現 ▼ を受け取ったとき
        スペース ▼ キーが押された まで待つ
        表示する
        ずっと
            マウスのポインター ▼ へ行く
            もし 色が 色に触れた なら
                スタート ▼ を送る
            もし 色が 色に触れた なら
                ゲームクリア ▼ を送る
            もし 色が 色に触れた なら
                ゲームオーバー ▼ を送る
    
```

“コース出現”メッセージが送られてきたとき

- ・スペースキーを押すまで待ちます。
- ・表示します。

ずっと くりかえし ます。

・爆弾が、マウスポインターの位置にいきます。

<もし、爆弾のふちがスタートエリアのラインにふれたら>

・“スタート”メッセージをプログラム全体に送ります。

<もし、爆弾のふちがゴールエリアにふれたら>

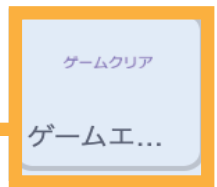
・“ゲームクリア”メッセージをプログラム全体に送ります。

<もし、爆弾のふちがコースにふれたら>

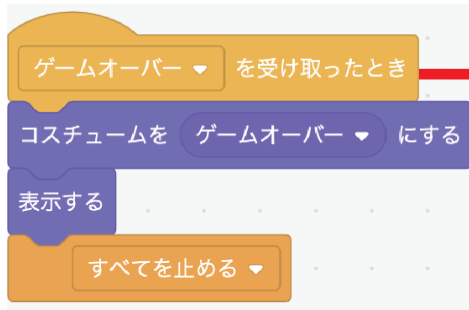
・“ゲームオーバー”メッセージをプログラム全体に送ります。

7

「ゲームエンド」のプログラムを作ります



次のプログラムを追加して、「爆弾」がコースにぶつかったときにゲームオーバーを表示します。



“ゲームクリア”メッセージが送られてきたとき

- ・コスチュームを“ゲームオーバー”にします。
- ・表示します。
- ・プログラム全体を止めます。

確認してみよう！

本項で作成したプログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・スタート後、スタートエリアでスペースキーを押して「爆弾」を表示させます。
- 「爆弾」がコースにふれたとき、“ゲームオーバー”の文言は表示されましたか？

8

「爆弾」のプログラムを作ります



下記のブロック定義を作成して、「爆弾」が爆発したときの動きを作成します。



“爆発”ブロックを実行したときの動き

- ・コスチュームを“爆発”にします。
- ・大きさを100%にします。
- ・0.2秒待ちます。
- ・かくします。

確認してみよう！

プログラムを実行して、動きを確認しよう。確認できたら□に✓をつけよう。

- ・ブロック定義「爆発」を実行します。
- 「爆弾」が爆発しましたか？