マニュアルに戻る。

ピンポンゲーム

このActivityでは、ボールをラケットで打ち返すピンポンゲームを作ってみる。

作成するゲーム

画面には四角い枠が描かれており、その中にボールの役割をするタートルと、長方形のラケットがある。ボールはラケットや壁にぶつかると跳ね返る。ラケットは左側の2個のボタンで上下に操作する。

ボールを制限時間の間、ラケットで打ち返し続けられたらクリアとなる。一回でもラケットの後ろの壁にボールがぶつかると、ゲームオーバーとなる。

CENTER:&show(http://dolittle.eplang.jp/data/wiki_image/image320/pingpong3.png,50%,画像);

壁を作る（ステップ1）

最初に、ゲームに登場するオブジェクトを画面に置くことにする。このゲームでは、5個の長方形が登場する。上下と右にある長方形は、ボールを跳ね返す壁である。左には2つの長方形が存在する。小さいほうはパドルである。ゲームをする人は、このパドルを上下に動かしてボールを打ち返す。大きいほうは、パドルでボールを打ち返せなかったことを知るための壁である。ボールがこの壁にぶつかると、ゲームオーバーになる。

CENTER:&show(http://dolittle.eplang.jp/data/wiki_image/image320/pingpong1.png,50%,画像);

次のプログラムでは、5個の長方形を作成している。プログラムを簡単にするために、**線の太さ**で太さ20の線を描くことで長方形とした。続いて、コの字の形に線を描き、「壁」という名前の図形を作っている。続いて上を向いた後、左壁とパドルを作っている。

このプログラムのように、座標を指定して図形やボタンを画面に置く場合には、終了ボタンの隣にある**方眼紙**をチェックして、位置を確認しながらプログラムを作ると便利である。

```
// 壁を作る（ステップ1）
かめた＝タートル！ 作る。
かめた！（緑）線の色 20 線の太さ。
かめた！500 歩く 90 右回り 420 歩く 90 右回り 500 歩く。
壁＝かめた！図形を作る -200 200 位置。
かめた！90 右回り。
左壁＝かめた！（黄）線の色 440 歩く 図形を作る -210 -230 位置。
パドル＝かめた！（青）線の色 120 歩く 図形を作る -190 -210 位置。
```

パドルを動かす（ステップ2）

画面にボタンを表示して、パドルを上下に動かしてみよう。次のプログラムでは、画面の左側に「上ボタン」と「下ボタン」という名前の2つのボタンを表示して、それぞれが押されたときにパドルを上下に50ずつ移動させている。今回はボタンの2個目のパラメータに**UP**と**DOWN**を指定することで、上下の矢印キーでも操作できるようにした。

CENTER:&show(http://dolittle.eplang.jp/data/wiki_image/image320/pingpong2.png,50%,画像);

```
// パドルを動かす（ステップ2）
上ボタン＝ボタン！"上" "UP" 作る -380 50 位置。
下ボタン＝ボタン！"下" "DOWN" 作る -380 0 位置。
上ボタン：動作＝「パドル！ 0 50 移動する」。
下ボタン：動作＝「パドル！ 0 -50 移動する」。
```

ボールを動かす（ステップ3）

「かめた」 には、壁やパドルを描いた後で、ボールの役割をしてもらうことにする。次のプログラムでは、かめたが移動したときに線を引かないように、**ペンなし**を実行している。ボールは、最初は右の壁に向かって動くようにした。タートルの初期位置は、**乱数**を使い、少しずつ異なる位置から動き始めるようにした。横の位置（x座標）は「乱数（200）」で1から200の値に、縦の位置（y座標）は「乱数（300）- 150」で-149から150の値になる。タートルの初期角度は45度で固定とした。

タートルが壁やパドルに衝突したときは、跳ね返る動作をすることが望ましい。そこで、かめたの**衝突**には、タートルオブジェクトの**跳ね返る**を設定した。タートルのぶつかった向きに合わせて、自然な角度で跳ね返ることができる。

最後にタイマーを生成し、「かめた！20 歩く」を0.1秒間隔で60秒間繰り返し実行するようにした。実行すると、かめたが壁で囲まれた空間を跳ね返りながら動き回る。

CENTER:&show(http://dolittle.eplang.jp/data/wiki_image/image320/pingpong3.png,50%,画像);

```
// ボールを動かす（ステップ3）
かめた！ ペンなし。
かめた！（乱数（200））（乱数（300）-150）位置。
かめた！45 向き。
かめた：衝突＝タートル：跳ね返る。
時計＝タイマー！ 作る 60秒 時間「かめた！20 歩く」実行。
```

ゲームの勝敗を判定する（ステップ4）

このゲームでは、パドルでボールを打ち返せずに1回でも左壁にボールが衝突したら負け（ゲームオーバー）とする。そして、タイマーでボールが動いている時間内にボールが左壁に衝突しなければ勝ち（ゲームクリア）である。

次のプログラムでは、勝敗を判定するために、「ゲームクリア」という変数を導入した。この変数には**はい**または**いいえ**という真偽値を代入して使う。初期値は「はい」であるが、ボールが左壁に衝突したときは「いいえ」 を代入する。このとき、「ゲームクリア」をルートのプロパティとして扱うために、前に「：」を付けている。そして、タイマーの実行を中断している。

タイマーが終了した後の処理は「最後に実行」で行う。タイマーがどのように終了したのかは、「ゲームクリア」の値で判定している。値が真（はい）の場合は、無事に60秒間経って終了したので、青い字で「ゲームクリア」を表示している。値が偽（いいえ）の場合は、左壁に衝突して中断したので、赤い字で「ゲームオーバー」を表示するようにした。

```
// ゲームの勝敗を判定する（ステップ4）
ゲームクリア＝はい。
左壁：衝突＝「：ゲームクリア＝いいえ。時計！ 中断」。
時計！「
    「ゲームクリア＝＝はい」！ なら「
        ラベル！"ゲームクリア！"作る （青）文字色。
    」そうでなければ「
```

```
        ラベル！"ゲームオーバー！"作る （赤）文字色。
    」実行。
」最後に実行。
```

CENTER:&show(http://dolittle.eplang.jp/data/wiki_image/image320/pingpongOver.png,50%,画像);
CENTER:&show(http://dolittle.eplang.jp/data/wiki_image/image320/pingpongClear.png,50%,画像);

以上でピンポンゲームは完成である。実行すると、左壁に衝突したときは「ゲームオーバー！」が表示される。時間内にすべてのボールをパドルで打ち返せたときは、「ゲームクリア！」が表示される。

最後にステップ1からステップ4までの全体のプログラムを掲載しておく。

```
// 壁を作る（ステップ1）
かめた＝タートル！ 作る。
かめた！（緑）線の色 20 線の太さ。
かめた！500 歩く 90 右回り 420 歩く 90 右回り 500 歩く。
壁＝かめた！図形を作る -200 200 位置。
かめた！90 右回り。
左壁＝かめた！（黄）線の色 440 歩く 図形を作る -210 -230 位置。
パドル＝かめた！（青）線の色 120 歩く 図形を作る -190 -210 位置。

// パドルを動かす（ステップ2）
上ボタン＝ボタン！"上" "UP" 作る -380 50 位置。
下ボタン＝ボタン！"下" "DOWN" 作る -380 0 位置。
上ボタン：動作＝「パドル！ 0 50 移動する」。
下ボタン：動作＝「パドル！ 0 -50 移動する」。

// ボールを動かす（ステップ3）
かめた！ ペンなし。
かめた！（乱数（200））（乱数（300）-150）位置。
かめた！45 向き。
かめた：衝突＝タートル：跳ね返る。
時計＝タイマー！ 作る 60秒 時間「かめた！20 歩く」実行。

// ゲームの勝敗を判定する（ステップ4）
ゲームクリア＝はい。
左壁：衝突＝「：ゲームクリア＝いいえ。時計！ 中断」。
時計！「
    「ゲームクリア＝＝はい」！ なら「
        ラベル！"ゲームクリア！"作る （青）文字色。
    」そうでなければ「
        ラベル！"ゲームオーバー！"作る （赤）文字色。
    」実行。
」最後に実行。
```

今後の改良としては、難易度を選べるようにして、実行時間、ボールの速さ、パドルの大きさなどを変えられるようにしてもよいかもしれない。また、パドルで打ち返した回数をカウントし、得点として表示することも考えられる。

From:
https://dolittle.eplang.jp/ - **プログラミング言語「ドリトル」**

Permanent link:
https://dolittle.eplang.jp/ch_pingpong?rev=1514993106

Last update: **2018/01/04 00:25**