

オブジェクト指向プログラミング 第6回

グラフィックスとタイマー 簡単な物理シミュレーション

担当：高橋、佐藤聖也

```
BallMain.java

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class BallMain extends JPanel implements ActionListener{
    //描画パネルのサイズ
    private int xPanelSize;
    private int yPanelSize;

    //BallPropagator クラスのインスタンス
    BallPropagator b1;

    //ボールの色
    Color c1;

    //コンストラクタ
    BallMain(int xPanelSize, int yPanelSize){
        //描画パネルのサイズ
        this.xPanelSize = xPanelSize;
        this.yPanelSize = yPanelSize;

        //BallPropagator クラスのインスタンスの生成（ボールの生成）
        //ボール b1 の初期状態（ボールの半径，位置，速度，加速度）
        b1 = new BallPropagator(10, 100, 100, 100, -50, 0, 0);

        //////////////////////////////////////
        //ここでボールの色を決める（RGB カラーの生成）
        //////////////////////////////////////
    }

    //イベント（タイマーによる呼び出し）が発生したときの処理
    public void actionPerformed(ActionEvent e){

        //ボール b1 の位置の更新
        b1.xUpdate();
        b1.yUpdate();
        //ボール b1 の速度の更新
        b1.vxUpdate();
        b1.vyUpdate();

        // 再描画
        repaint();
    }
}
```

```
// 描画するときの処理
public void paintComponent(Graphics g){
    super.paintComponent(g);

    g.fillOval((int)(b1.x-b1.r), (int)(b1.y-b1.r),
               (int)(2*b1.r), (int)(2*b1.r));

}

// メイン・メソッド
public static void main(String[] args) {

    // 微小時間間隔
    BallPropagator.dt=0.1;

    // フレームを生成する
    JFrame frame = new JFrame();

    // パネルを生成する
    BallMain panel = new BallMain(500,400);
    panel.setBackground(Color.white);
    panel.setPreferredSize(
        new Dimension(panel.xPanelSize, panel.yPanelSize));

    // ボールの動く範囲を決定する (境界条件)
    BallPropagator.xmin = 0;
    BallPropagator.xmax = panel.xPanelSize;
    BallPropagator.ymin = 0;
    BallPropagator.ymax = panel.yPanelSize;

    // フレームを設定する
    frame.add(panel);
    frame.pack();
    frame.setTitle("反射するボール");
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setResizable(false);
    frame.setVisible(true);

    // イベントを定期的発生させるためのタイマー
    Timer timer = new Timer(10, panel);
    timer.start();

}

}
```

BallPropagator.java

```
public class BallPropagator{

    public static double dt;//プロパゲータの微小時間間隔
    public static int xmin; //境界座標
    public static int xmax;
    public static int ymin;
    public static int ymax;

    double r;      //ボールの半径
    double x, y;   //ボールの位置 (中心座標)
    double vx, vy; //ボールの速度
    double ax, ay; //ボールの加速度

    ////////////////////////////////////////
    //
    // ここに BallPropagator クラスのコンストラクタを書きなさい
    // ( r, x, y, vx, vy, ax, ay を初期化する )
    //
    ////////////////////////////////////////

    //x 軸方向の位置の更新
    public void xUpdate() {
        this.x = this.x + this.vx * dt;

        //x 方向の反射 ( cf. 反発係数 )
        if(this.x < (double)xmin+this.r ){
            this.vx *= -1;
            this.x = (double)xmin+this.r;
        }else if((double)xmax-this.r < x ){
            this.vx *= -1;
            this.x = (double)xmax-this.r;
        }
    }

    //y 軸方向の位置の更新
    public void yUpdate() {
        this.y = this.y + this.vy * dt;

        //y 方向の反射 ( cf. 反発係数 )
        if(this.y < (double)ymin+this.r ){
            this.vy *= -1;
            this.y = (double)ymin+this.r;
        }else if((double)ymax-this.r < this.y ){
            this.vy *= -1;
            this.y = (double)ymax-this.r;
        }
    }
}
```

```
//////////////////////////////////////////  
//  
// ここに x 軸方向の速度を更新するためのメソッドを書きなさい  
//  
//////////////////////////////////////////  
  
//////////////////////////////////////////  
//  
// ここに y 軸方向の速度を更新するためのメソッドを書きなさい  
//  
//////////////////////////////////////////  
}
```

課題

1. BallPropagator.java に、コンストラクタを書き加えなさい。
2. BallMain.java にコードを加筆して、ボールに色をつけなさい。
3. BallMain.java にコードを加筆して、反射するボールを 2 つに増やしなさい。
4. BallPropagator.java に、ボールの速度を更新するためのメソッドを書き加えなさい。ただし、加速度はどちらのボールも $ax = 0$, $ay = 9.8$ とせよ。
5. BallPropagator.java を書き換え、ボールが壁に非弾性衝突するようにせよ。ただし、反発係数 $\gamma = 0.8$ (gamma) を用いること。
6. BallPropagator クラスのインスタンスフィールド r , x , y , vx , vy , ax , ay にはアクセス修飾子が付いていない。ここに `private` 修飾子を加えた上で、プログラムが動作するよう BallPropagator.java および BallMain.java を書き換えなさい。(アクセサメソッドを追加する。)

課題提出

BallPropagator.java のファイル冒頭に課題が何番までできたか明記した上で、BallPropagator.java および BallMain.java を提出せよ。ただし、ファイル名はそれぞれ BallPropagator_00rd000.java, BallMain_00rd000.java のように学籍番号を付け、実行可能にしてから提出すること。

提出の際の注意点：

- ・ コンパイルすると実行できる形式で提出すること。
- ・ 学籍番号のフォルダに提出ファイルを全て入れ、そのフォルダを zip ファイルにして提出すること。

提出先：<https://tdu.app.box.com/f/533ec58b594e4e7e8007acf063641a4e>

提出期限：6/21 23:59