

ソフトウェア演習Ⅱ (Java) [課題 6 : GUI+インタフェース例] 担当 : 青野雅樹

Java によるオブジェクト指向言語の概念の中でクラスの継承とともに、重要な概念がインタフェース(interface)である。GUI を使ったプログラムでインタフェースを理解することが、今回の課題の主旨である。以下の内容を満たす Java プログラムと、GUI を使ったアニメーションの静止画像 (キーボードの適当なキーでストップした状態の画像) を Moodle にアップせよ。〆切は 8 月 3 日とする。プログラムの動作テストができるように必要なプログラムと条件(4)で得られる画像をまとめて ZIP でアップすること。

第 6 回の資料で説明している GUI+インタフェースプログラム ([Runnable インタフェースと KeyListener インタフェース利用するプログラム](#)) を以下のように改良せよ。

[1] Ball.java というボールクラスを作成せよ。

[2] main 関数を含むクラス (Kadai6.java) では、Ball クラスのオブジェクトを [10 個~30 個の範囲で生成し](#)、以下の条件を満たすアニメーションプログラムを作成せよ。

【条件】

(1) Ball.java には、以下のメンバー変数を持たせる。

(ア) ボールの半径 (1~80 のランダムな半径)

(イ) 色(0.0~1.0 の RGB それぞれランダムな色)

(ウ) 位置 (x,y とともに[0~400]のランダムな初期位置からスタート)

(エ) 方向ベクトル(dx, dy とともに、[-40.0~+40.0]の間のランダムなベクトル)

(オ) 直前の位置(lastx, lasty)(初期値は x,y にそれぞれ等しい)

これらをメンバー変数でもたせること。[色は float または double のいずれかとし、その他のデータ型は任意](#)とする。このとき、*i* 番目のボールは、(dx[i], dy[i]) の方向ベクトルで移動することとなる (ベクトルの大きさが速度に相当)。

(2) アプリケーション全体を制御する Swing の JFrame クラスではキーボードのイベントを検出させること。イベントとしては、以下の 3 種類。

(ア) ESC キーで終了

(イ) S キーでアニメーションを静止

(ウ) R キーでアニメーションを再開

なお、JFrame のラベルに各自の名前と日付とボール数を加えること。

(3) Ball.java クラス内で定義するボールの移動制御部分は『コメントとヒント』を参照。

(4) 実行結果の静止画は、S キーを押したところでスクリーンコピーをとり、jpg か png 画像とすること。

(5) main プログラムで作成する JFrame のウィンドウのサイズは、600×600 とし、[起動時の引数でボール個数を入力](#)させること。(例 : javau Kadai6 30)

(6) Thread の sleep 時間は適当な数値 (ミリ秒) で固定してよいが、数値を埋め込まず、プログラム内の変数 (final 可) で与えること。

コメントとヒント：

条件(3)のボールの移動制御は `animate` 関数または、`Ball.java` 内の適当な関数（例：`public void move(java.awt.Rectangle bounds)`のような関数）で、以下のように行うことを薦めます。第6回の授業の資料にあるプログラム 6-17 では、`animate` 関数でボールの移動制御を行っていましたが、`Ball` クラスを作るので、その中で以下のような中身をもつ `move` 関数のようなものを生成して行うのが簡単かと思います。

```
if ((x - radius + dx < 0 || x + radius + dx > bounds.width))
    dx = -dx; //画面の左右の壁にぶつかったら x 方向を変更
if ((y - radius + dy < 0 || y + radius + dy > bounds.height))
    dy = -dy; //画面の上下の壁にぶつかったら y 方向を変更
x = lastx + dx; //現在の x 位置
y = lasty + dy; //現在の y 位置
lastx = x;
lasty = y;
```

以下の図が実行例です。S キーでアニメーションを止めたものです。この課題では、**Ball クラスのオブジェクトを複数生成することになります**が、シンプルなのは配列で以下のように（`Kadai6.java` の中で）宣言します。

`Ball [] balls = null;`（後述の `saveBalls` も同様）で宣言しておいて、コンストラクタか `public void init()` メソッドのような 1 回だけ初期化用に呼び出す関数を（`Kadai6.java` の中で）準備して、そこで、たとえば

```
balls = new Ball[numBalls];
for (int i = 0 ; i < numBalls ; i++){
    balls[i] = new Ball();
    saveBalls[i] = new
Ball(); /* ストップ、再開時の
回避用 */
}
```

のように、オブジェクトを作成しておくといいかと思います。個々のオブジェクト作成時に、条件(1)の個々のメンバー変数の初期化を行えば結構です。

