

プログラミング演習I (第6回) 課題

• 基本課題① basic_Dice

- 400x300のウィンドウを作成せよ。
- ウィンドウをクリックすると、Aさんが2,4,6の目が均等に出る3面ダイス、Bさんが1,4,7の目が均等に出る3面ダイスを振るようにせよ。
- 振った結果を示し、どちらが勝ったか(または引き分けたのか)を下図のようにprintやprintlnを用いて「標準出力するプログラム」を作成せよ。値の大きい方が勝ち、同じなら引き分けとする。

クリック!

```
Aさん: 2  
Bさん: 3  
Bさんの勝ち
```

クリック!

```
Aさん: 2  
Bさん: 3  
Bさんの勝ち  
Aさん: 4  
Bさん: 3  
Aさんの勝ち
```

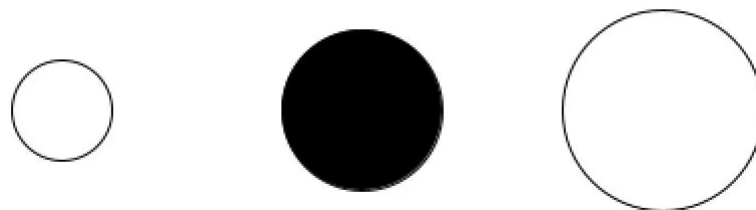
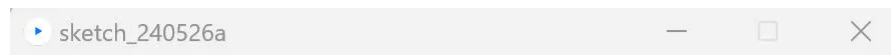
クリック!

```
Aさん: 2  
Bさん: 3  
Bさんの勝ち  
Aさん: 4  
Bさん: 3  
Aさんの勝ち  
Aさん: 4  
Bさん: 4  
引き分け
```

プログラミング演習I (第6回) 課題

• 基本課題② basic_ToggleButton

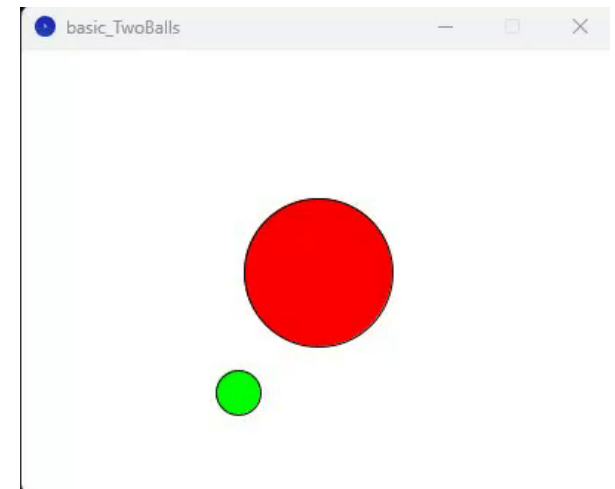
- 450x150の背景白色のウィンドウを作成し、その上下中央に3つの丸いボタン（左から順に、中心(75,75)で直径50ピクセル、中心(225,75)で直径80ピクセル、中心(375,75)で直径100ピクセルとする）を離しつつ並べて提示せよ。
- また、そのボタンがそれぞれマウスのボタンで押されるたびに塗りつぶし色が白→黒→白→黒と変化するようにせよ。
- それぞれのボタンは独立して動作するようにせよ。
- 丸の外部で反応しないようにし、また丸の内部はすべて反応するようにせよ。
- 起動時は、左から白・黒・白とせよ。



プログラミング演習I (第6回) 課題

• 基本課題③ スケッチ名：`basic_TwoBalls`

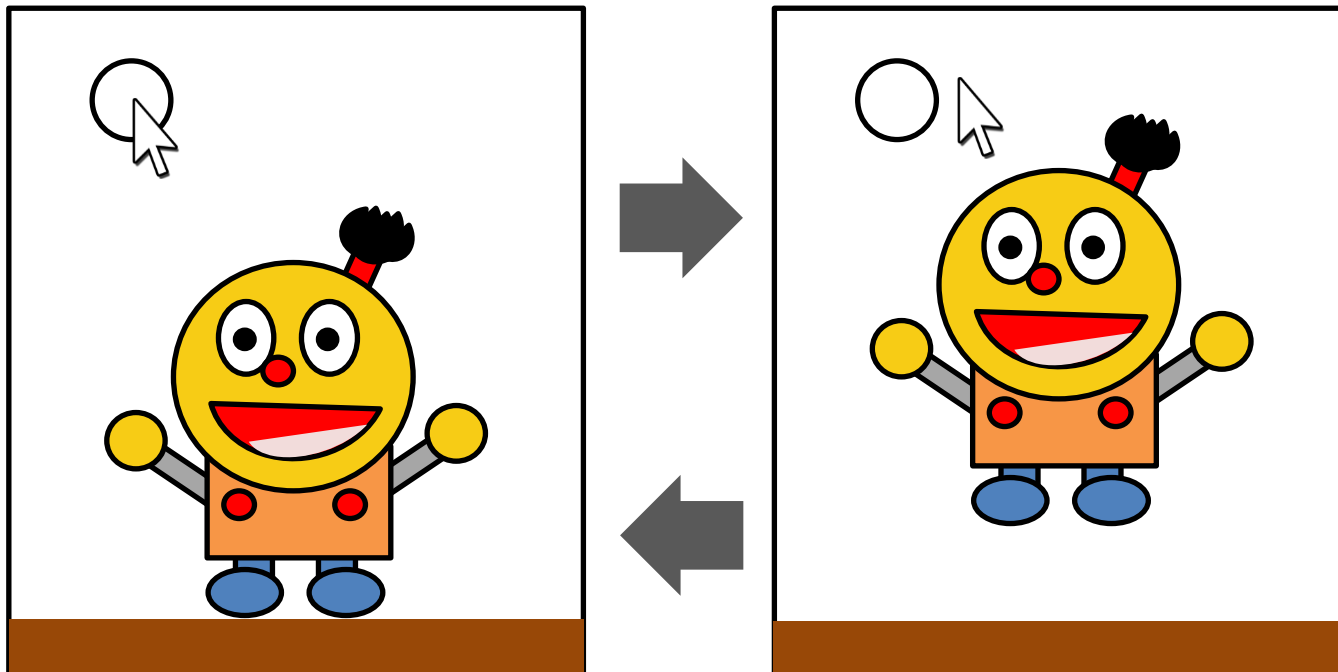
- 400x300のウィンドウ内を、毎フレームX方向に3ピクセル、Y方向に2ピクセル動く緑色の円（直径を30ピクセルとする）を作成せよ。円の初期位置は画面内でランダムな位置にせよ。
- 動く円は上下左右の端まで来ると跳ね返るようにせよ（円の中心部が端に来た時に跳ね返るようにするだけでよい）
- また、ウィンドウ中央に直径100ピクセルの赤色の円を描画せよ。
- 動く緑色の円が、赤色の円と少しでも重なったときに、動く円の色を黄色にせよ。また、赤色の円から完全に離れたら緑色になるようにせよ。
- 重なるというのはどういう距離の計算になるかな？



プログラミング演習I (第6回) 課題

• 発展課題① advanced_Charajump

- 以前作成したdrawCharacter(int cx, int cy)を利用し、キャラクターがジャンプするプログラムを作りたい (ellipseなどのように使えます)
- キャラクターと地面となる面を表示し、画面左上に用意した直径50ピクセルの丸ボタンをクリックすることでキャラクターがジャンプするようなプログラムを作成せよ。
- ジャンプについては初速をわかる程度に設定しつつ鉛直上方投射するものとし、地面にくると止まるようにせよ。



プログラミング演習I (第6回) 課題

発展課題② advanced_MonteCarlo

- モンテカルロ法とは確率に無関係のものを確率で計算するものである。円周率の近似値を求めたい。
- まず、800x800のウィンドウ内に直径800ピクセルの円を描け。また、draw()のたびに、画面内でランダムにXY座標を決定し、その点が円の外側なら直径3ピクセルの青丸を、内側なら直径3ピクセルの赤丸を表示せよ（線は太さ1で、その塗りつぶし色と同一のものにせよ）
- 800x800の正方形の面積は800x800、直径800ピクセルの円の面積は $400 \times 400 \times \pi$ であるため、
$$\frac{\text{円の面積}}{\text{正方形の面積}} = \frac{400 \times 400 \times \pi}{800 \times 800} = \frac{\pi}{4}$$
 となる。
- $$\frac{\text{円内の点の数}}{\text{正方形内の点の数}} \approx \frac{\text{円の面積}}{\text{正方形の面積}} = \frac{\pi}{4}$$
 のため、
$$\pi \approx 4 \times \frac{\text{円内の点の数}}{\text{全ての点の数}}$$
 となる。これを利用し、円周率の近似値をdraw()1000回毎に標準出力せよ。

