

今日のテクニック

ランダムな値を発生したいときは random を利用しよう！

0以上で10より小さい値をランダムに発生する方法は

```
float num = random( 10 );
```

0以上で5より小さい整数値 (0, 1, 2, 3, 4) を同一の確率でランダムに発生する方法は...

```
int num = (int)random( 5 );
```

(0.000... ~ 4.9999... の値が生成されそれを整数化)

10以上20以下の整数値をランダムに発生する方法は

```
int num = (int)random( 10, 21 );
```

プログラミング演習I (第4回) 課題

• 基本課題① basic_Dice

- 400x300のウィンドウを作成し, そのウィンドウ内をクリックするたびに, Aさんが1~6の目が均等に出るサイコロ, Bさんが1~6の目が均等に出るサイコロを2つ振った結果を示すとともに, どちらが勝ったか (または引き分けたのか) を下図のように標準出力するプログラムを作成せよ
- 値をランダムに生成するには, random を使えば良い

クリック!

```
Aさん: 3  
Bさん: 12  
Bさんの勝ち
```

クリック!

```
Aさん: 3  
Bさん: 12  
Bさんの勝ち  
Aさん: 6  
Bさん: 5  
Aさんの勝ち
```

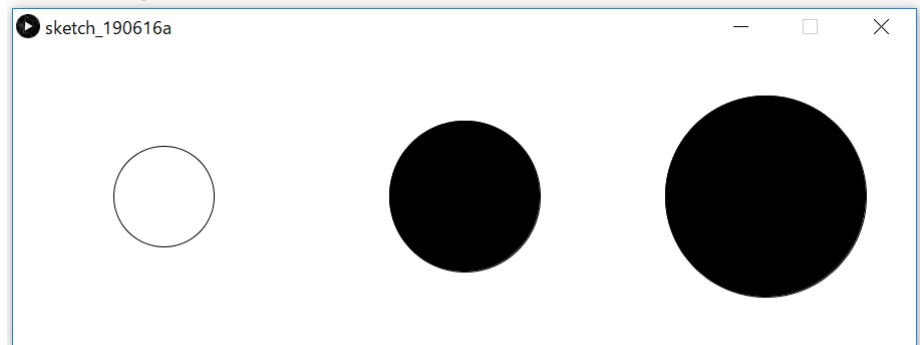
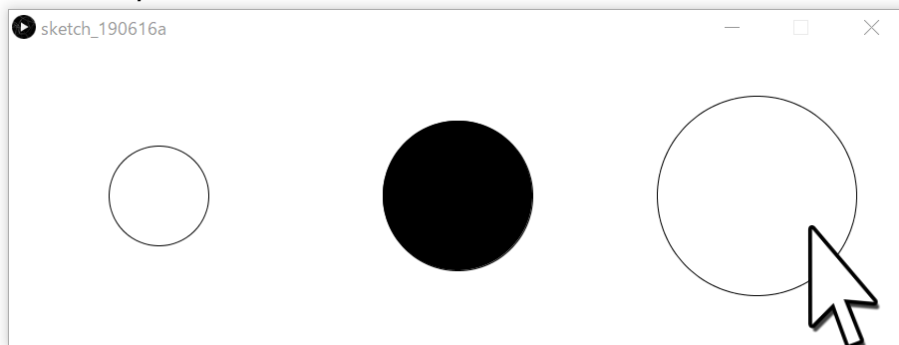
クリック!

```
Aさん: 3  
Bさん: 12  
Bさんの勝ち  
Aさん: 6  
Bさん: 5  
Aさんの勝ち  
Aさん: 6  
Bさん: 6  
引き分け
```

プログラミング演習I (第4回) 課題

• 基本課題② basic_ToggleButton

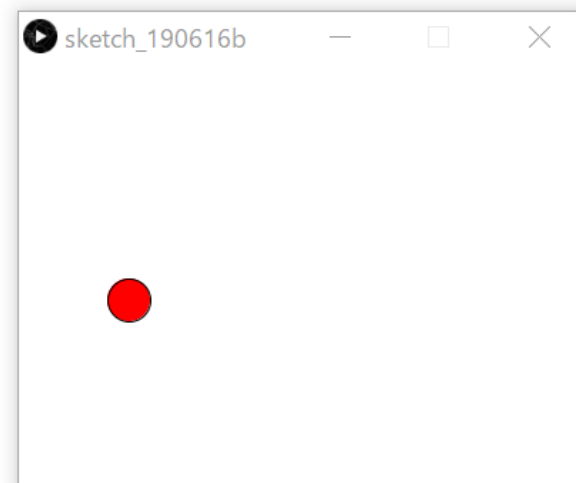
- 900x300のウィンドウを作成し, その上下中央に3つの丸いボタン (左から順に, 中心(150,150)で直径100ピクセル, 中心(450,150)で150ピクセル, 中心(750,150)で200ピクセルとする) を離しつつ並べて提示せよ
- また, そのボタンがそれぞれクリックされるたびに塗りつぶし色が白→黒→白→黒と変化するようにせよ (起動時は全部白とする)
- それぞれのボタンは独立して動作するようにせよ
- 丸の外部で反応しないようにし, また丸の内部はすべて反応するようにせよ



プログラミング演習I (第4回) 課題

• 基本課題③ basic_ColorfulBall

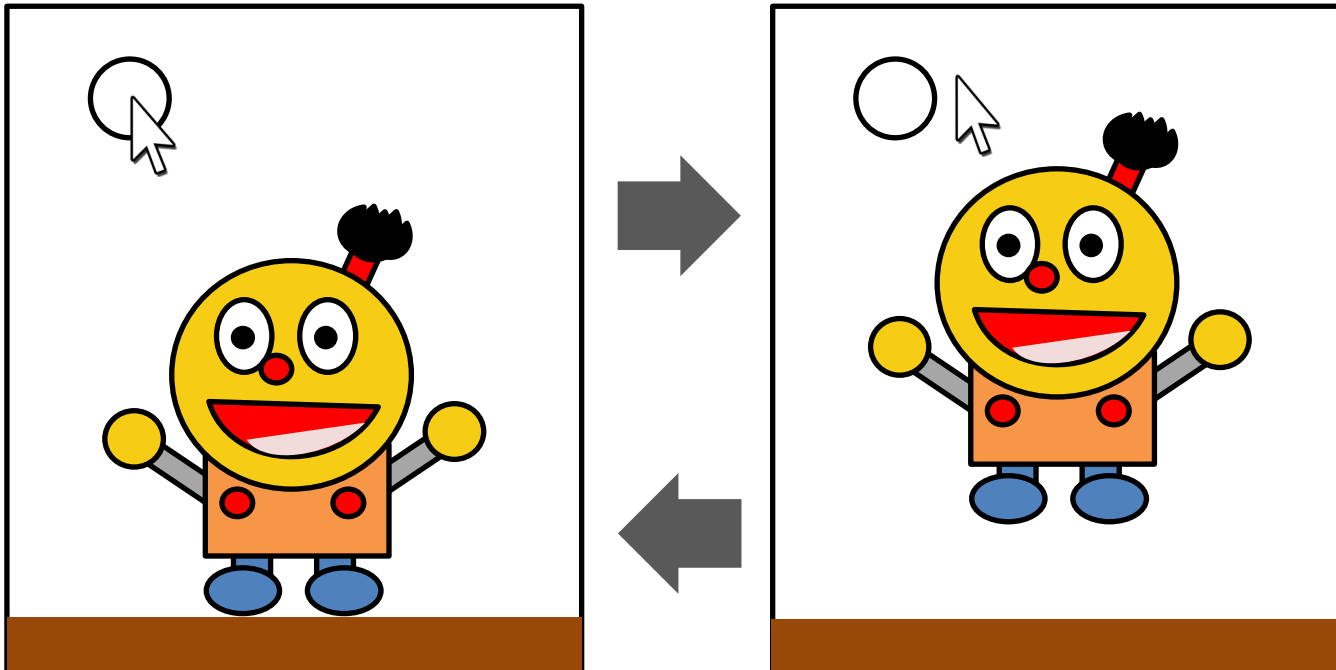
- 400x300のウィンドウ内を、毎フレームX方向に3ピクセル、Y方向に2ピクセル動く直径30ピクセルの円を作成せよ。
- 円の初期位置は画面内でランダムな位置にせよ。
- 円は上下左右の端まで来ると跳ね返るようにせよ（円の中心部が端に来た時に跳ね返るようにするだけでよい）
- 円の塗りつぶし色は、跳ね返る度に「緑→黄→赤→緑→黄→赤→」と変化するようにせよ（緑・黄・赤を繰り返すもの）



プログラミング演習I (第4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4回) 課題

発展課題② advanced_MonteCalro

- 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。ここで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（線もその色にせよ）
- 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()100回毎に標準出力せよ。

