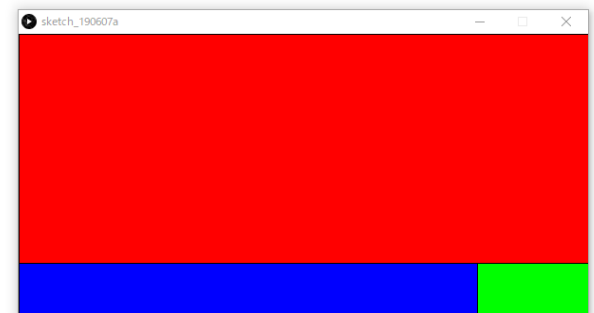
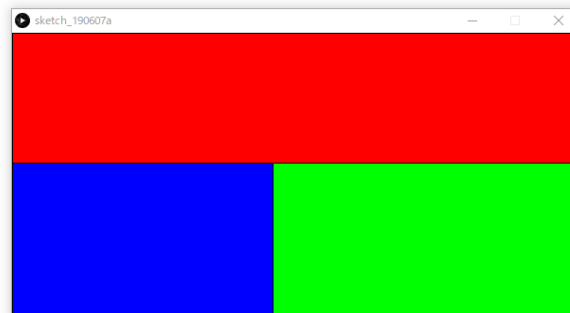


# プログラミング演習(3) 課題

## 基本課題① スケッチ名: threeAreas2

- 縦600x横600のウィンドウを作成せよ
- プログラムを起動して最初は画面全体が緑色(0, 255, 0)で描画され, 100ミリ秒に1ピクセルずつ上端から赤色(255, 0, 0)の領域が, 100ミリ秒に1ピクセルずつ左端から青色(0, 0, 255)の領域が迫りくるように描画せよ
- 起動後, 約60秒後に全面が赤色になる



# プログラミング演習I (3) 課題

## • 基本課題② リサーチユカーブ : **curve**

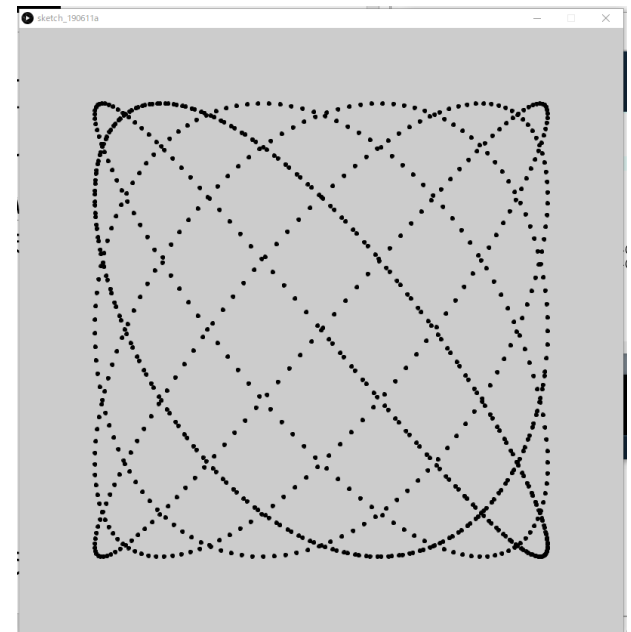
–  $x$ と $y$ の座標が $t$ によって変化する下記の数式の計算結果の座標に直径5ピクセルの黒色の円を描画せよ。ただし、 $t$ は `draw()` 毎に0から0.01ずつ増加するようにせよ。またウィンドウサイズは800x800とせよ。

- $x = 300 \sin(5t) + 400$

- $y = 300 \sin(6t) + 400$

– 5と6の値を変えると他の図形になるよ！

– 出来た人は遊んでみましょう



# プログラミング演習(3) 課題

---

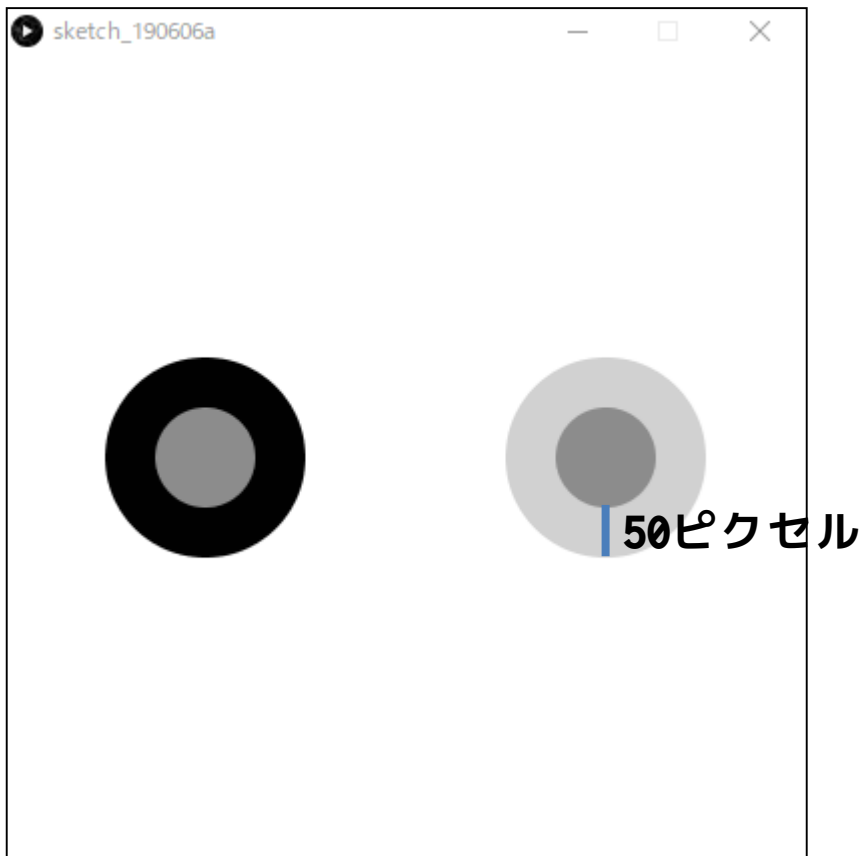
## • 基本課題③ スケッチ名: **syncIllusion2**

- 下記問題についてif/for/whileを使わず実現せよ
- 縦横400x400のウィンドウを作成せよ
- (100, 200), (300, 200)を中心に直径50ピクセルの円を2つ配置し, その2つの円の塗りつぶし色を約50フレーム毎に黒色(0, 0, 0)から白色(255, 255, 255)に変化し, さらに白色に戻るようして同期するようにせよ
- 上記の2つの円を取り囲む円を2つ描画し, その左側の円の塗りつぶし色をマウスのX座標に応じて白色から黒色に, 右側の円の塗りつぶし色をマウスのY座標に応じて白色から黒色に変更するようにせよ
  - ただし, 左上で白色, 右下で黒色となるようにせよ
- 錯覚が発生するように, 背景は白色, 線は描かないようにせよ

# プログラミング演習(3) 課題

- 基本課題③ スケッチ名: `syncIllusion2`

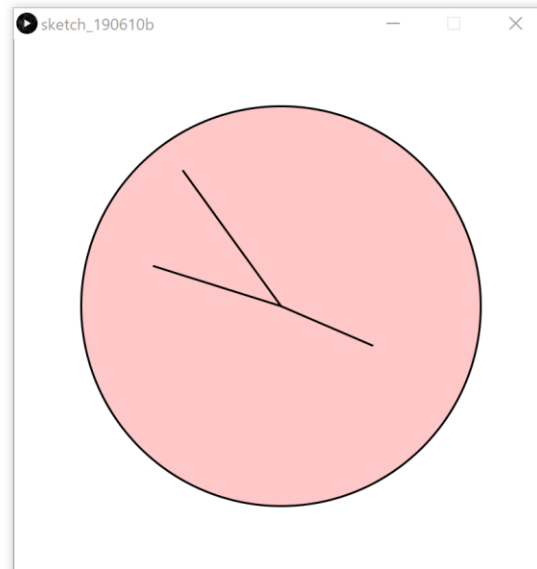
– 出力例



# プログラミング演習I (3) 課題

## • 発展課題① スケッチ名：**analogueClock**

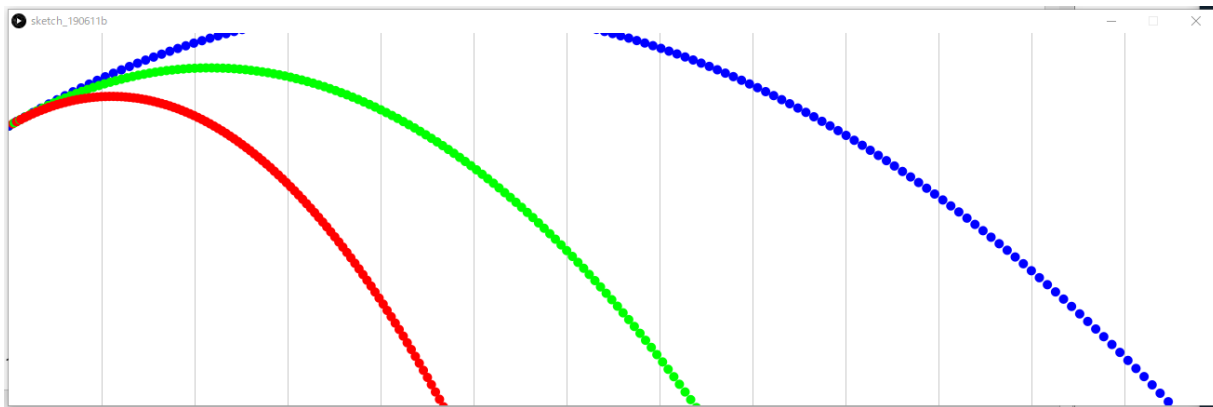
- 800x800のウィンドウの中に直径600の大きさの円を描き、その円の中心から時・分・秒の方向にそれぞれ短針・長針・秒針を表示するアナログ時計を実現せよ。また、短針・長針・秒針は識別可能なものとせよ。数字を配置するなど、表示は凝っても良い
- 短針の位置は分、長針の位置は秒を考慮せよ



# プログラミング演習I (3) 課題

## 発展課題② スケッチ名：balls

- 建物の屋上（300メートルの高さ）から斜め上30度方向にボールを投げ、落下するときの様子をシミュレーションするプログラムを作成してください（ウィンドウサイズは1300x400とせよ）。
- 3種類の投射速度（50m/s、70m/s、100m/s）で同時に投げたときの比較結果を下図のように軌跡が残るようにして示せ。
  - 50m/sを赤，70m/sを緑，100m/sを青色とせよ（輪郭の色は自由とする）
- 距離を把握しやすくするため，100mごとに灰色の縦線を描画せよ



ヒント：斜方投射の式

$$x = v_0 \cos(\theta) t$$

$$y = v_0 \sin(\theta) t - \frac{1}{2} g t^2$$

$$g = 9.8$$

※1フレームの経過時間は0.1秒とせよ。また、1ピクセル=1mと考えてください。