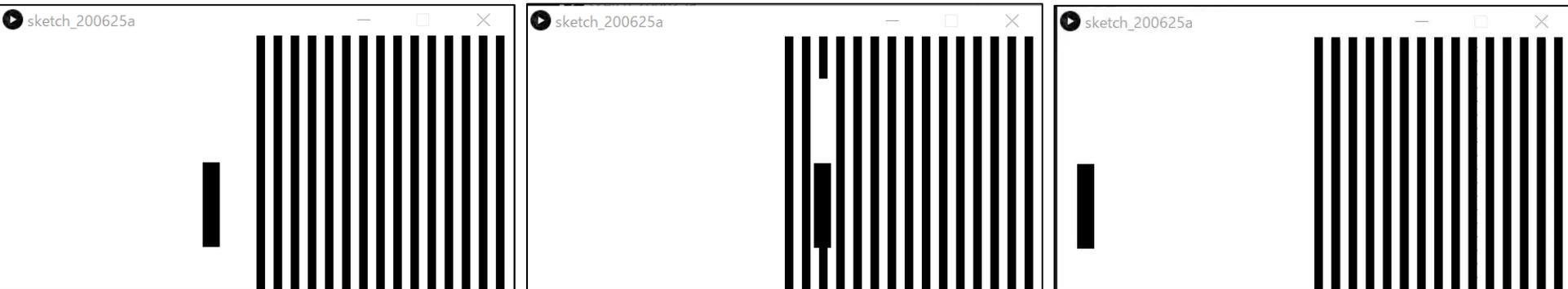


プログラミング演習(3) 課題

基本課題① スケッチ名: `basic_FootStep`

- 横600x縦300のウィンドウを作成せよ
- プログラムの画面右半分 (x座標が300のところから) 横幅10ピクセルの黒色の四角形を間を10ピクセル飛ばして画面右端まで描画するようにせよ (わかる人はfor/whileを使ってもよい)
- また、画面の左端からy座標が50ピクセルの位置に、縦100ピクセル、横20ピクセルの白色の四角形を描画し、同じくy座標が150ピクセルの位置に、縦100ピクセル、横20ピクセルの黒色の四角形を描画し、その2つの四角形をdrawのたびに1ピクセルずつ右へと移動させよ。なお、右端まで来ると左端から登場させよ
- 注意: 四角形に枠線は付けないようにせよ!



プログラミング演習I (3) 課題

• 基本課題②アルキメデスの螺旋：basic_Spiral

- 800x800のウィンドウを作成し、xとyの値が θ (theta) 【度】によって変化する、下記の数式の計算結果の座標(x, y)をpointを利用して描け
- ただし、thetaはdraw()毎に1ずつ増加するようにせよ
- thetaをcos、sinの中で使う場合はradiansで度からラジアンに変換すること

$$x = a\theta \cos\theta + b$$

$$y = a\theta \sin\theta + c$$

$a=0.1$, $b=400$, $c=400$ とする

```
int theta = 0;
void draw(){
    x,yを定義
    x,yの計算
    point( x, y );
    theta = theta + 1;
}
```

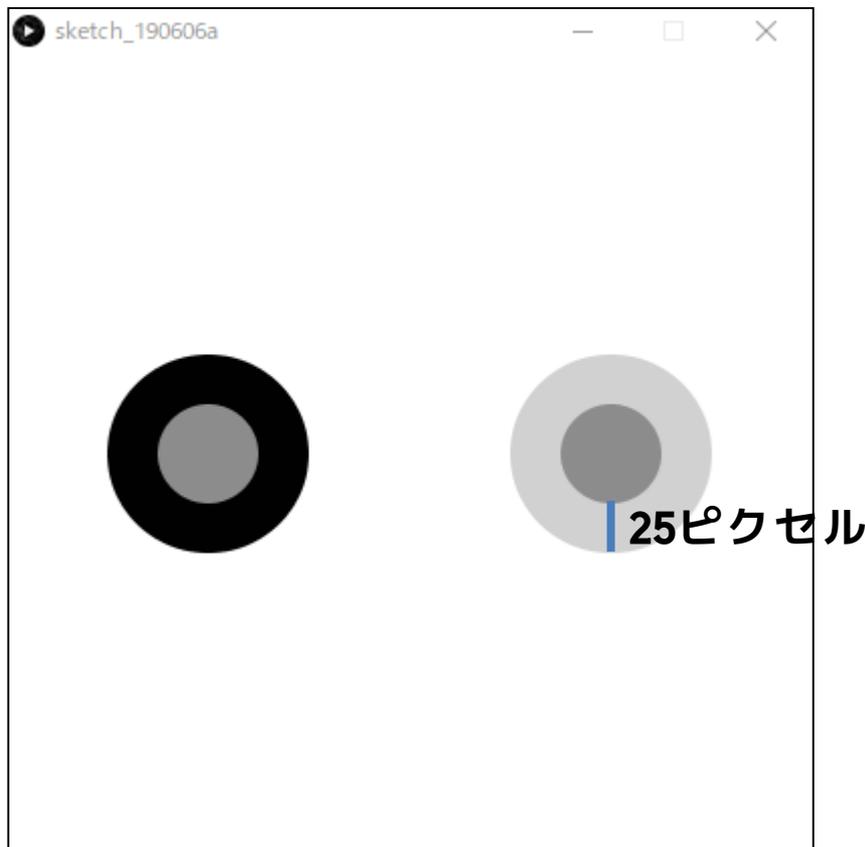
プログラミング演習(3) 課題

- **基本課題③** スケッチ名: basic_SyncIllusion
 - 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, さらに黒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(直径100ピクセル)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: `basic_SyncIllusion`

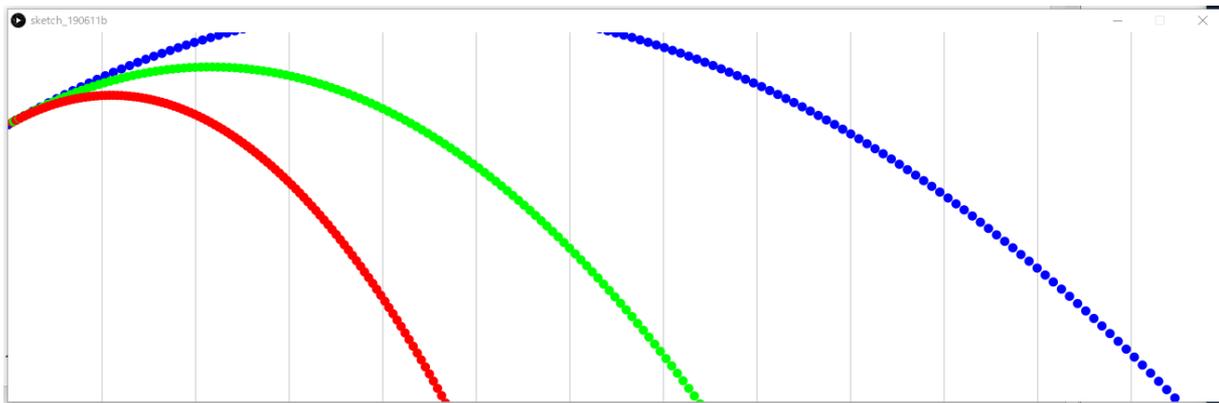
– 出力例



プログラミング演習I (3) 課題

発展課題① スケッチ名：advanced_ThrowBalls

- 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

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

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

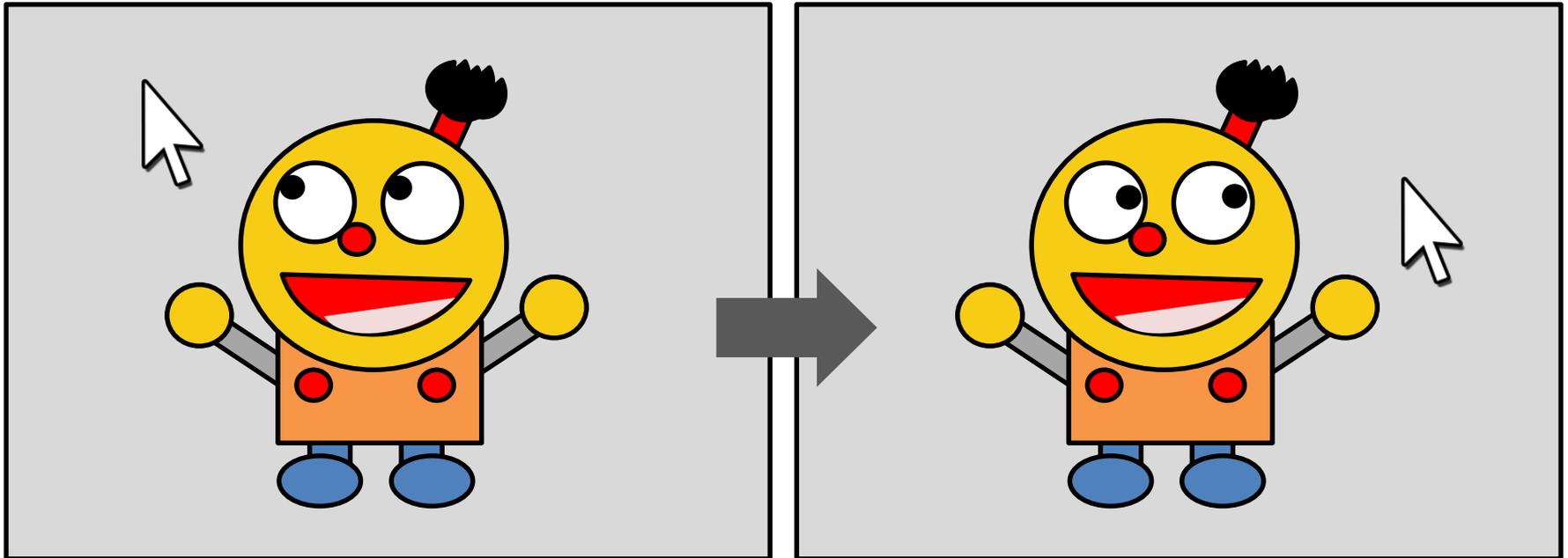
$$g = 9.8$$

※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。

プログラミング演習I (第3回) 課題

• 発展課題② スケッチ名：advanced_MoveEyes

- 前回作成したキャラクターを描くプログラムのdrawCharacterを配布プログラムにコピーして改良し、キャラクターの目（黒目がある場合は黒目）がマウスマウスカーソルのある方向を常に追いかけるプログラムを作成せよ。なお黒目がない場合は、追加してもよい。



※例えば目玉が楕円形の場合、黒目の動きは楕円軌道にこだわらなくてもよいです。

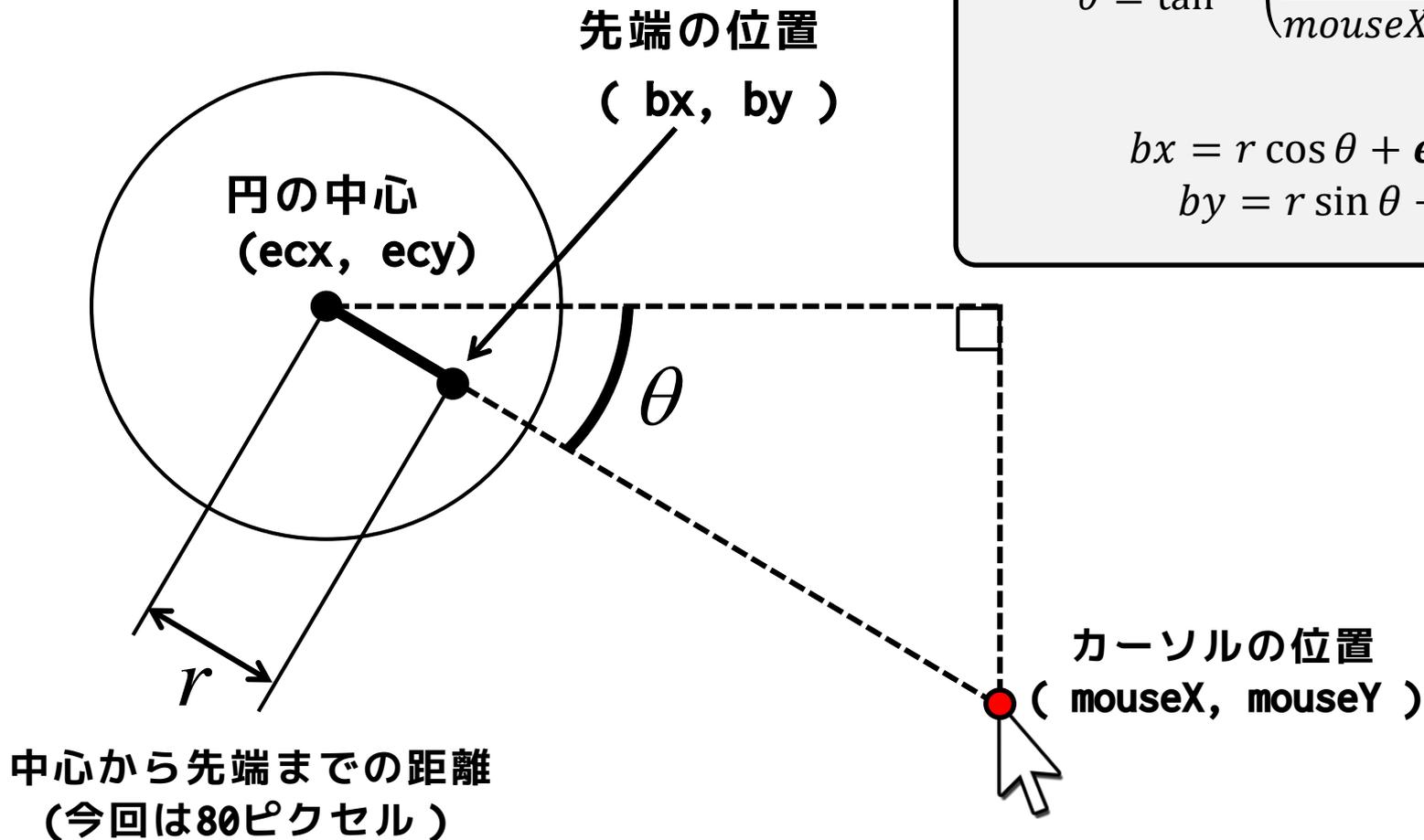
プログラミング演習I (第3回) 課題

目玉と黒目がある場合の考え方

- 目玉の中心は (x_1, y_1) で、黒目を円周上を動かすと考えた時、その半径 (または直径) は [r] である
- ある黒目の中心を ($eye1X, eye1Y$) とした時に、その目はどう表現できるかを考える
 - $eye1X, eye1Y$ を使って黒目を描画しよう
 - $eye1X, eye1Y$ には適当な値を代入しておこう (初期値等)
- ($mouseX, mouseY$) と目の中心 (x_1, y_1) のなす角度 $theta$ を計算しよう (次頁の $atan2$ を参照)
 $theta = atan2(mouseY - y_1, mouseX - x_1);$
- $theta$ を利用して $eye1X$ と $eye1Y$ の座標を計算して求めよう
 $eye1X = x_1 + r * cos(theta);$
 $eye1Y = y_1 + r * sin(theta);$
- $eye1X, eye1Y$ を利用して黒目を描画しよう (他の黒目も同様に)

プログラミング演習I (第3回) 課題

発展課題① のヒント



黒目の位置を求めるための数式

$$\theta = \tan^{-1} \left(\frac{mouseY - ecy}{mouseX - ecx} \right)$$

$$bx = r \cos \theta + ecx$$

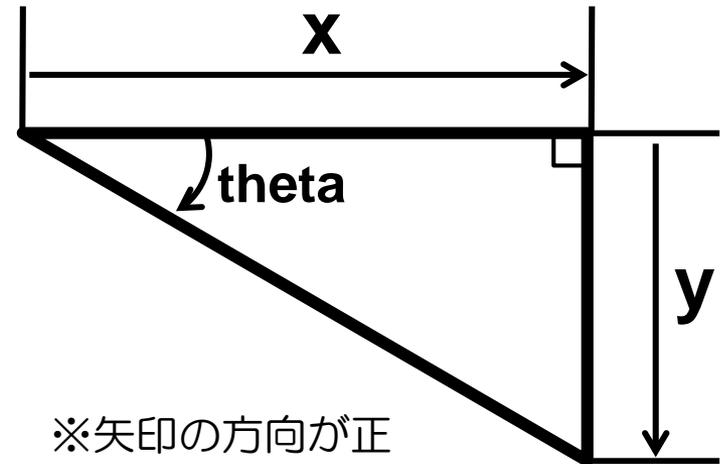
$$by = r \sin \theta + ecy$$

プログラミング演習I (第3回) 課題

• \tan^{-1} の求め方は2つある

アークタンジェントの計算には **atan()** と **atan2()** があり、それぞれ値域が異なります。なお、いずれも計算結果は実数値(float)です。

今回の課題では**atan2()** を使うとよいでしょう (atanの場合は問題が発生するが理由はわかるかな?)



$$\mathbf{theta = atan(y/x);} \quad \text{値域は} \quad -\frac{\pi}{2} \leq \mathbf{theta} \leq \frac{\pi}{2}$$

$$\mathbf{theta = atan2(y, x);} \quad \text{値域は} \quad -\pi \leq \mathbf{theta} \leq \pi$$