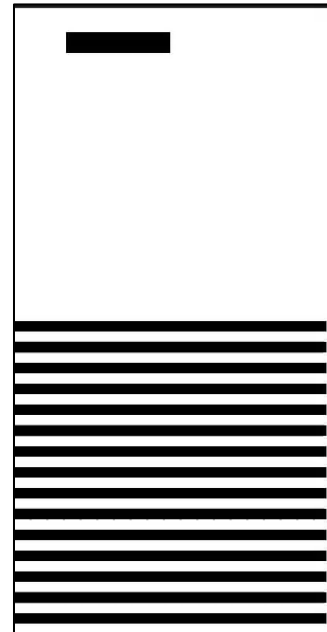
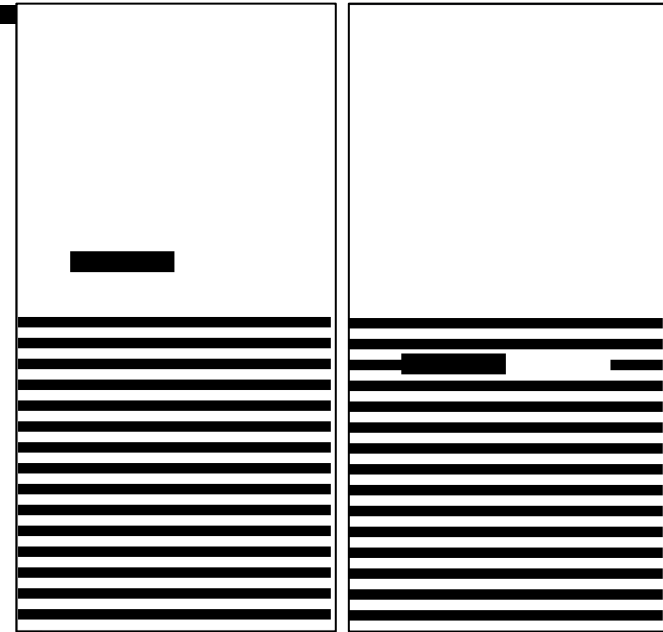


プログラミング演習(3) 課題

• 基本課題① basic_FootStep

- 横300x縦600のウィンドウを作成せよ
- プログラムの画面下半分（y座標が300のところから）に、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下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、画面の上端からx座標が50ピクセルの位置に、縦20ピクセル、横100ピクセルの黒色の四角形を描画し、同じくx座標が150ピクセルの位置に、縦20ピクセル、横100ピクセルの白色の四角形を描画し、その2つの四角形をdrawのたびに1ピクセルずつ下へ移動させよ。なお下端まで来ると上端から登場させよ
- 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(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)を中心に直径3ピクセルの円 (黒色で塗りつぶす)を描け
- ただし、thetaはdraw()毎に2ずつ増加するようにせよ (採点が面倒なので、frameRateはいじらないこと)
- 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

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

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

$$a = 0.1, b = 400, c = 400 \text{ とする}$$

- frameCountを使っても良い

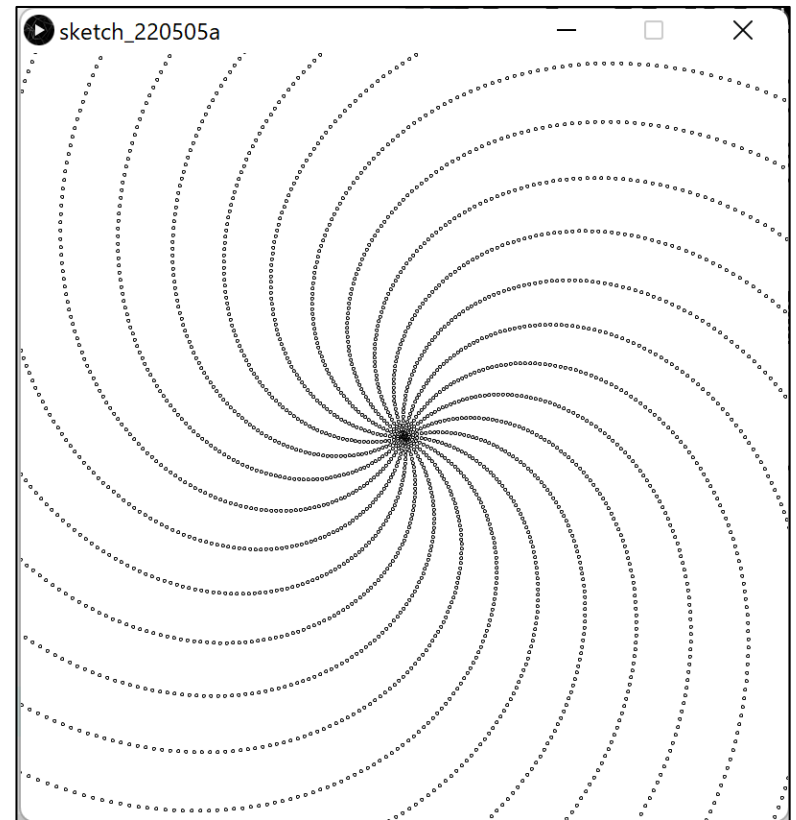
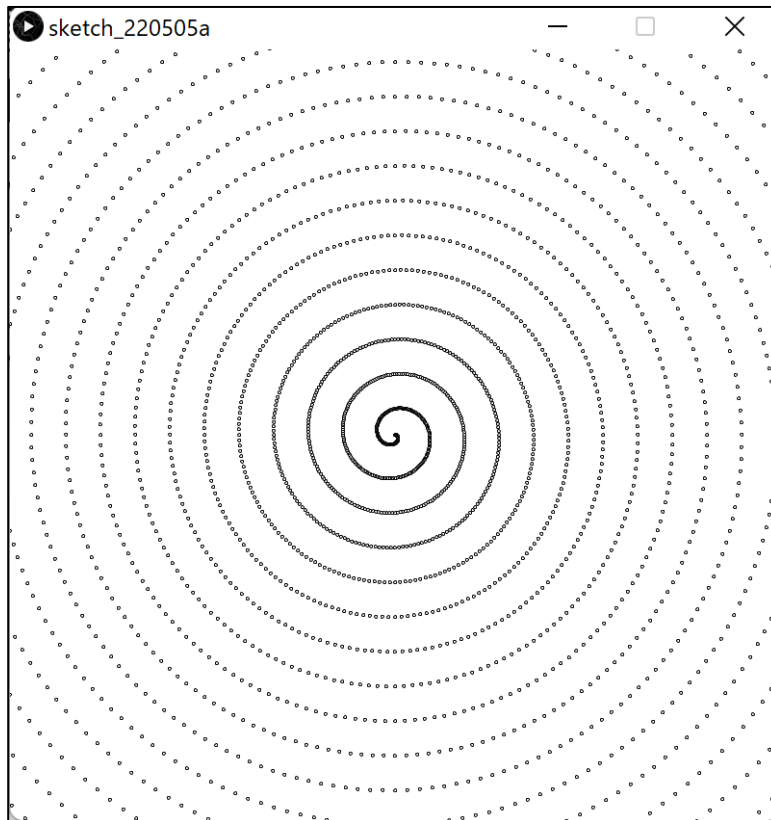
```
int theta = 0;

void setup(){
  初期化処理
}

void draw(){
  x,yを定義
  x,yの計算
  ellipse(x, y, 3, 3);
  theta = theta + 2;
}
```

プログラミング演習(3) 課題

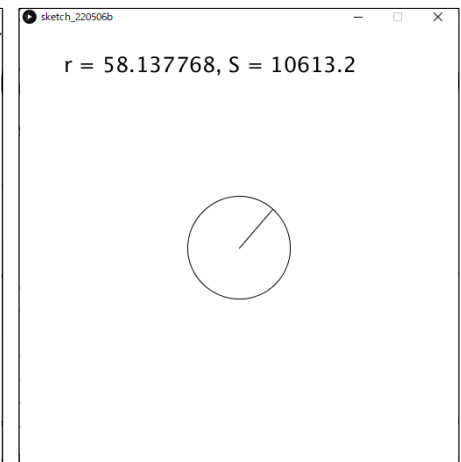
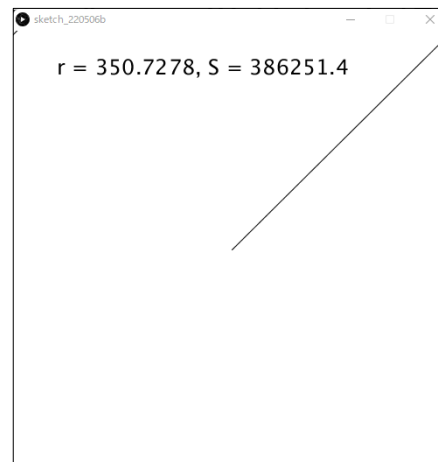
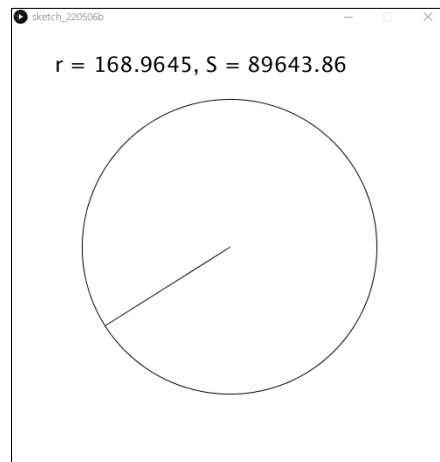
- アルキメデスの螺旋（左）、 \sin や \cos の中で使う θ をradiansでラジアン単位に変換しないと右のようになるよ！ ただこれもかっこいいので色々値を変えたりしながら楽しんでみよう！



プログラミング演習(3) 課題

• 基本課題③ スケッチ名: **basic_CircleArea**

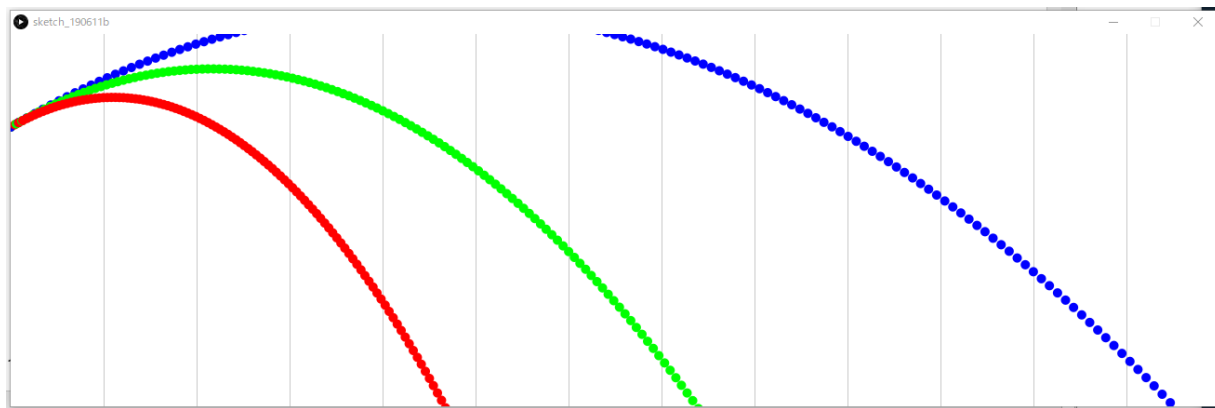
- 縦横500x500のウィンドウを作成せよ
- ウィンドウの中央から、マウスの座標(mouseX, mouseY)までの線を描け
- また、(250, 250)を中心に(mouseX, mouseY)までの距離を半径とする白色の円を描け
- さらに、円の半径rと面積Sを求め、textを利用して下図のように画面出力せよ。描画の順序に注意すること。



プログラミング演習(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

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

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

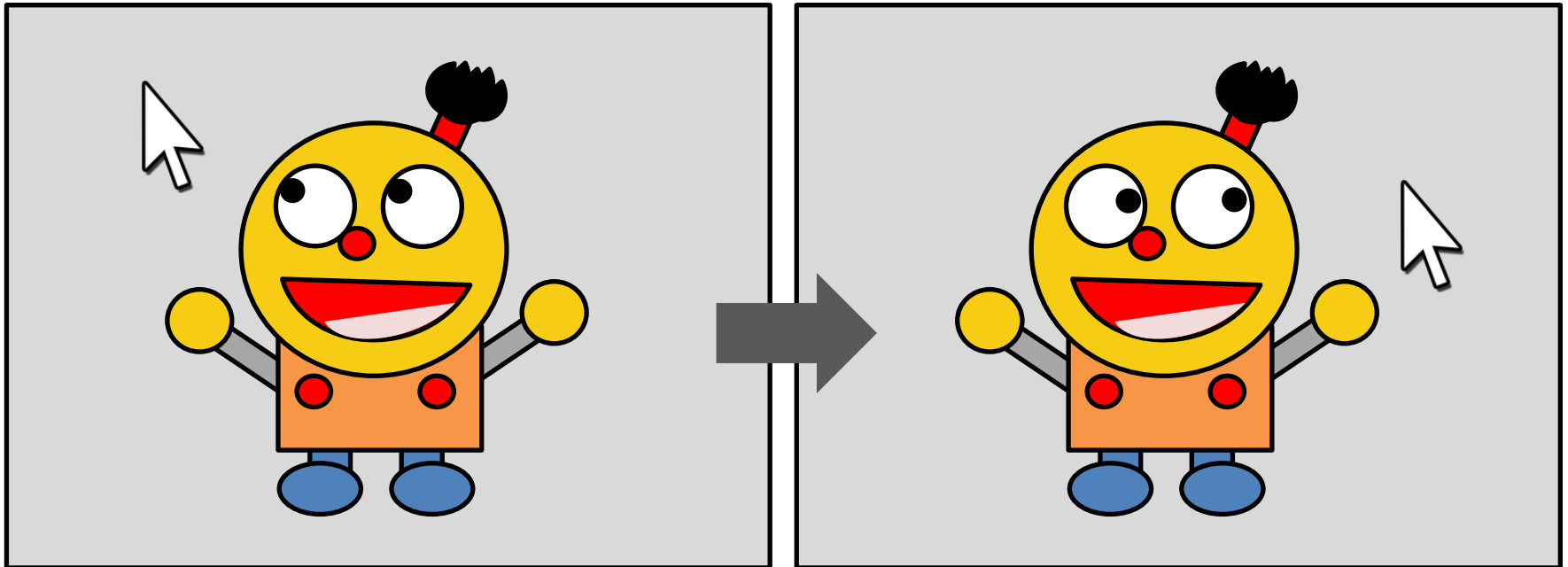
$$g = 9.8$$

※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(3) 課題

発展課題② スケッチ名：`advanced_MoveEyes`

- キャラクターを描くプログラムを改良し、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、目を追加してもよい。少なくとも1つの目が動けば正解としますが、ぜひ全部の目を動かしてみてください（白目部分はなくてもOKです）

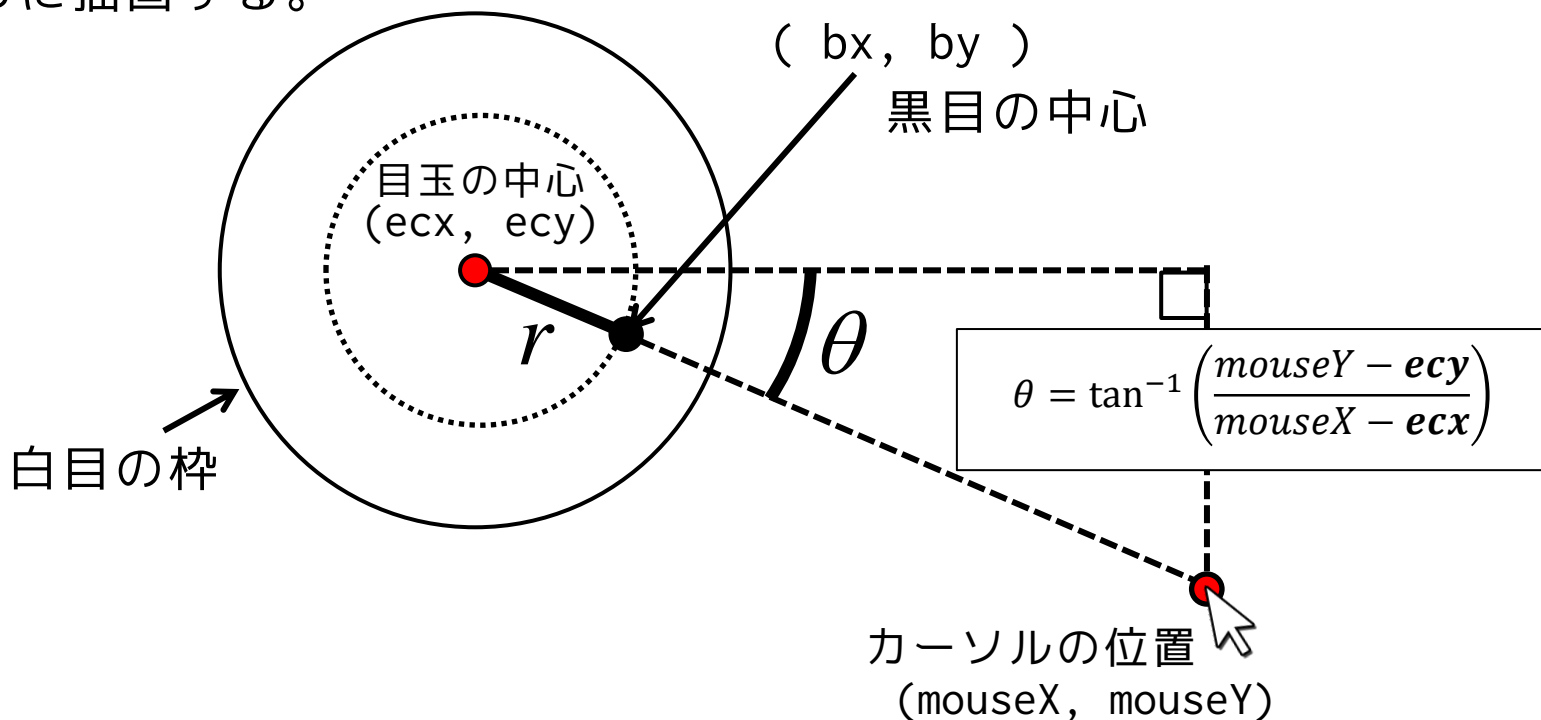


※例えば目玉が楕円形の場合、黒目の動きは楕円軌道にこだわらなくてもよいです。

プログラミング演習(3) 課題

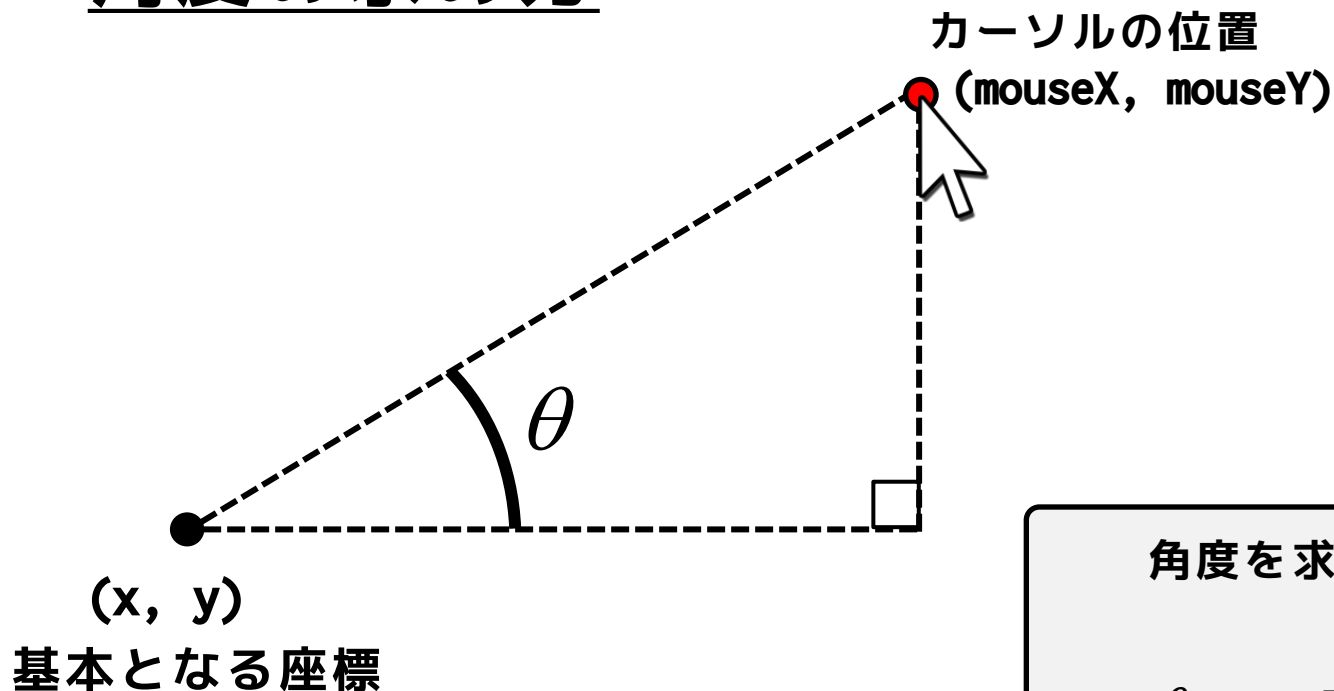
目玉と黒目がある場合の考え方 (1つの目に関する解説)

- まず、目玉の中心(ecx , ecy)を中心として、黒目(bx , by)を半径 r の円周上で動かすと考える。
 - ecx や ecy 、 r については、実際のキャラクタに基づき値を決め打ちする
- 黒目は、目玉の中心と、マウスの座標とを結ぶ線と、半径 r の円が交わるところに描画する。



プログラミング演習(3) 課題

• 角度の求め方



角度を求めるための数式

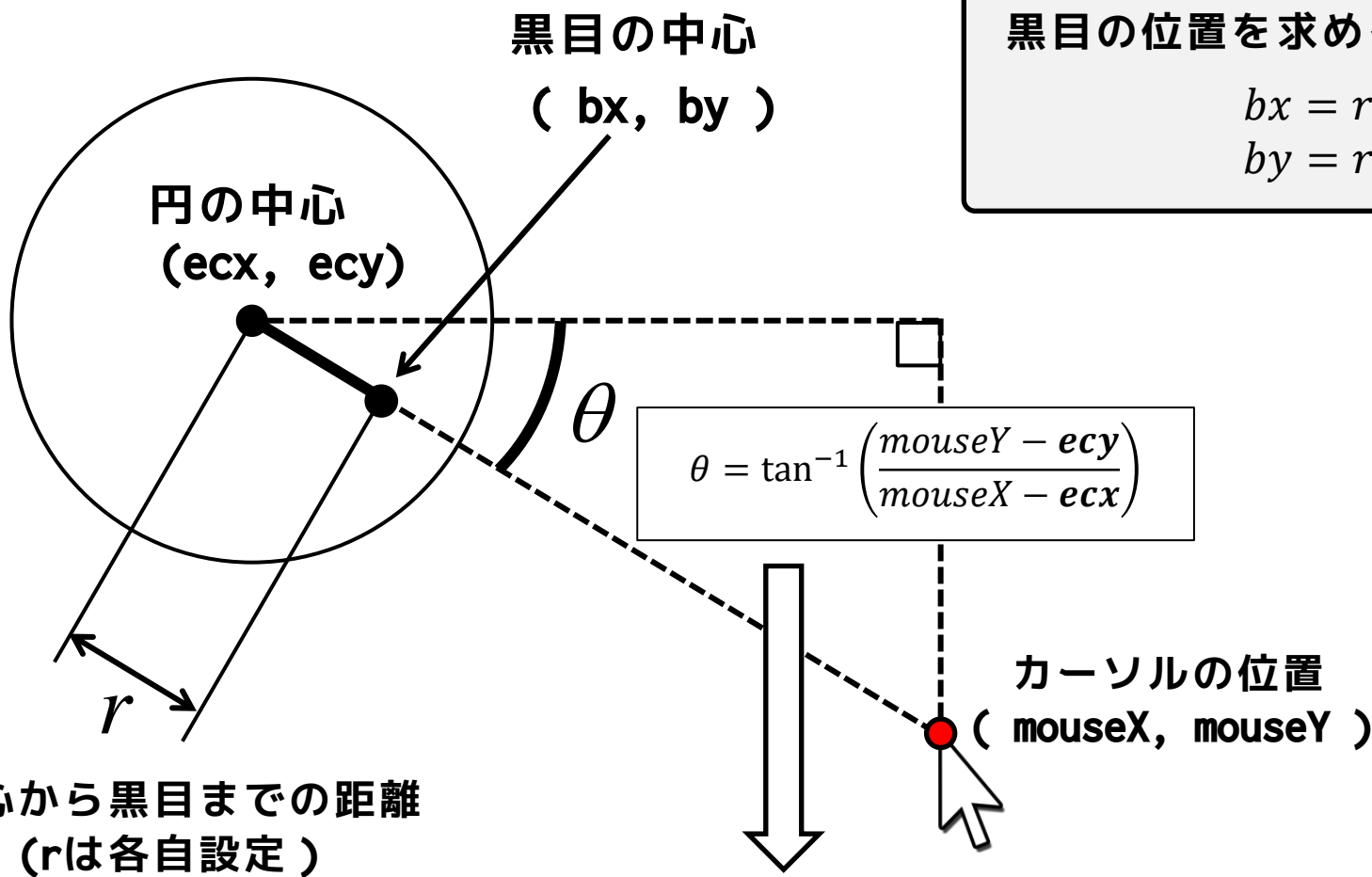
$$\theta = \tan^{-1} \left(\frac{\text{mouseY} - y}{\text{mouseX} - x} \right)$$



```
theta = atan2(mouseY-y, mouseX-x);
```

プログラミング演習(3) 課題

発展課題② のヒント



黒目の位置を求めるための数式

$$bx = r \cos \theta + ecx$$
$$by = r \sin \theta + ecy$$

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

```
theta = atan2(mouseY-ecy, mouseX-ecx);
```