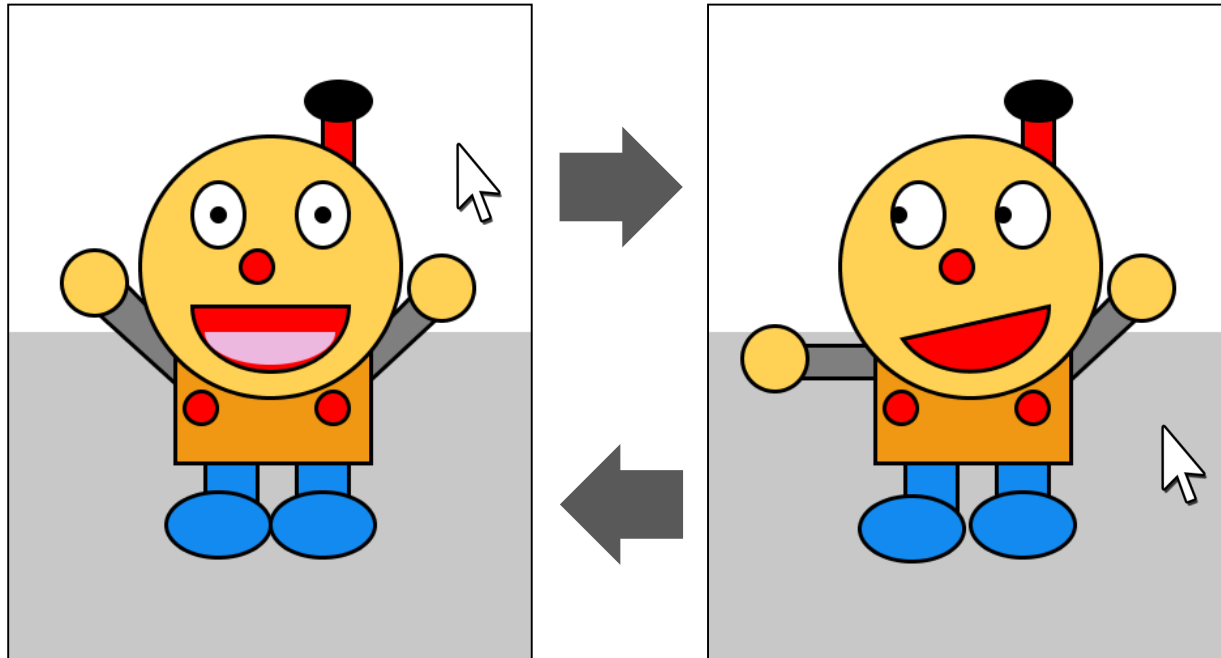


# プログラミング演習I (第4回) 課題

## • 基本課題① スケッチ名 : charapara

- キャラクタを表示し、画面の上半分、下半分にマウスマウスカーソルが来た時にそれぞれ絵がパラパラと切り変わるプログラムを作れ。
- 前々回のchara1とchara2のプログラムをうまく再利用しよう。



# 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)

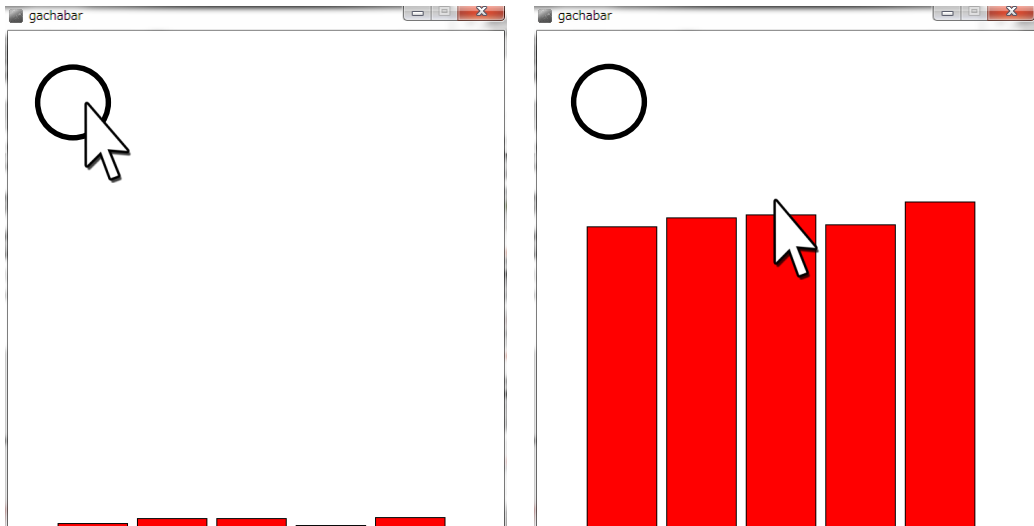
10以上20以下の整数値をランダムに発生する方法は

```
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② スケッチ名: gachachart

- 500x500のウィンドウとし左上の○ボタンがマウスクリックされるたびに5種類のカードの1種類がランダムに選ばれるプログラムを作成し, それぞれのカードが選ばれた枚数を棒グラフとして表示するプログラムを作成せよ. ただし, ○ボタン以外では反応しないようにせよ.
  - 値をランダムに生成するには, `random` を使えば良い
  - `random( 0, 3 );` で, 0以上3より小さい実数値がランダムに生成される



# プログラミング演習I (第4回) 課題

---

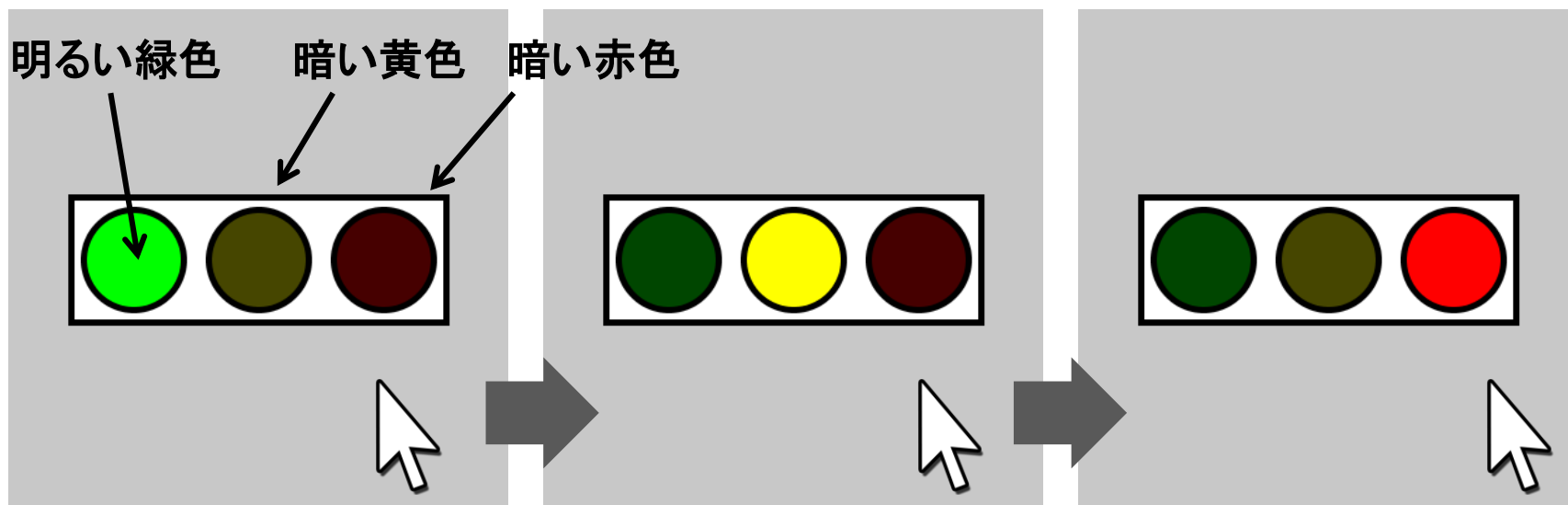
- ヒント

- それぞれの値を管理する変数を用意しよう
- card1, card2, card3, card4, card5 など
- ランダムに整数値を発生させ, その数に応じて card1 ~ card5 の変数の値を増やす
- card1 ~ card5 の値を表示する
- (配列でやってもOK)

# プログラミング演習I (第4回) 課題

## 基本課題③ スケッチ名 : signal

- 信号機のプログラムを作ってください。
- マウスクリックによって光る丸が青→黄→赤→青→... とシフトしていくようにしてください。
- クリックする場所は画面内のどこでも良いです。



画面をクリックしたら色が変わる！

赤の次はまた青に戻るように！

# プログラミング演習I (第4回) 課題

## 発展課題① スケッチ名: launch

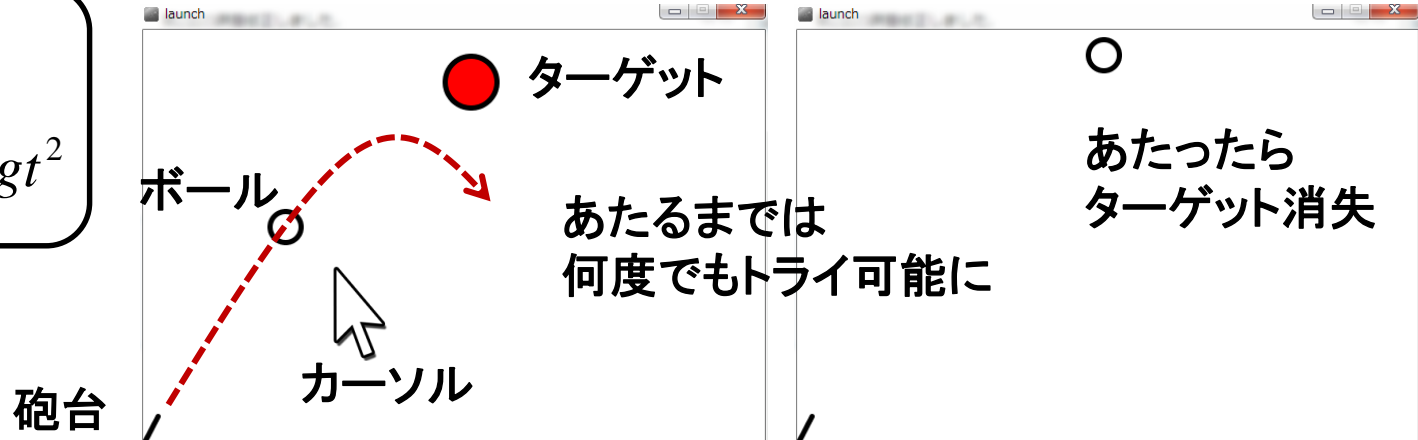
- 600x400のウィンドウ左下からマウスカーソルがある方向にボール(半径15m)を発射する砲台のプログラムを作ってください。クリックされたタイミングで投射されるようにすること。また、クリック時の砲台からカーソルまでの距離を初速とせよ。さらに、300mの距離に半径25mの赤丸のターゲットを描け。赤丸は上下に5m/フレームで動き、上端下端で折り返すようにせよ。なお、ボールがターゲットに衝突すると、ターゲットを消すようにせよ。1フレームは0.1秒とする

### 斜方投射の式

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

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

スピードが早すぎる場合はすり抜けてもよい



あたるまでは  
何度でもトライ可能に

○  
あたったら  
ターゲット消失

# ヒント

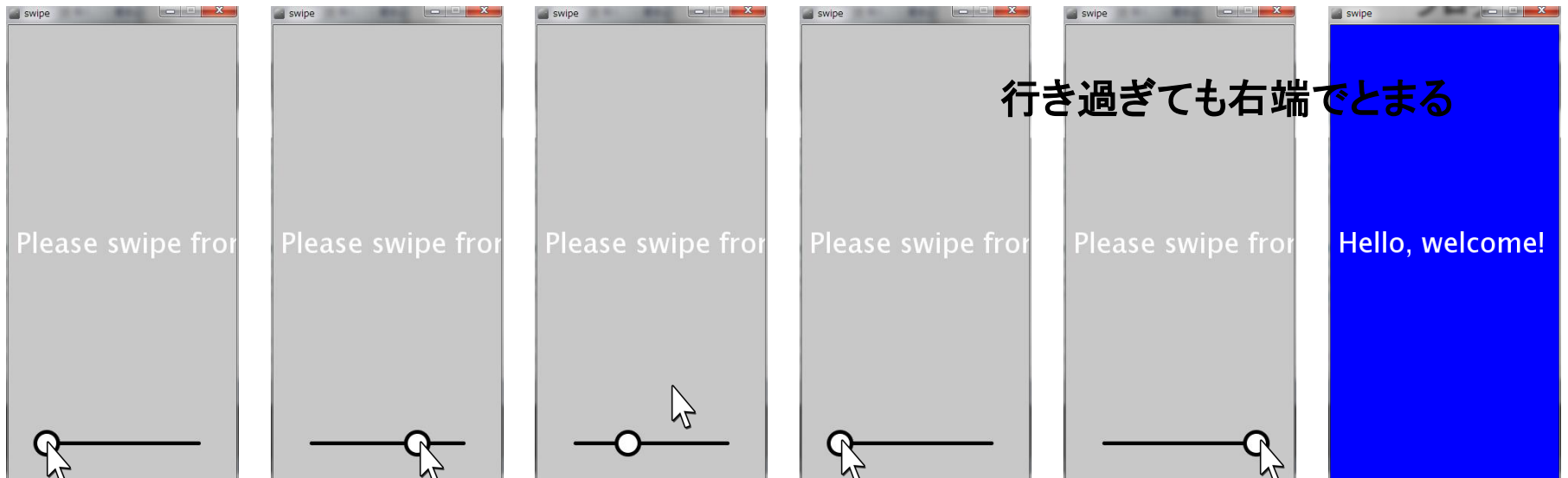
---

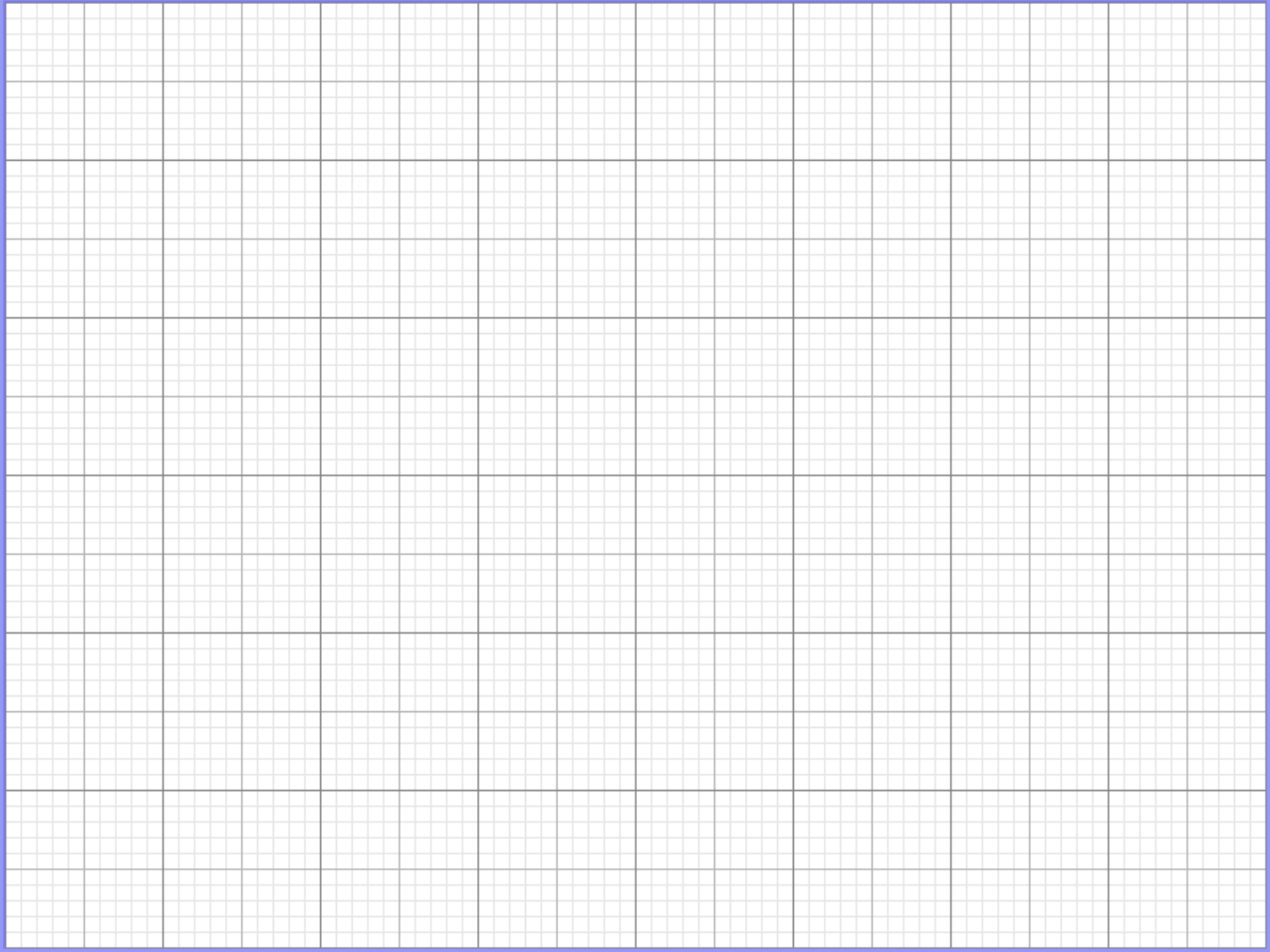
- この世界を実現する上で必要な変数(変化するものを取りあえず片っ端から)をまず列挙！
  - マウスの座標
  - 砲台の角度
  - ボールの投射角度
  - ボールの初速
  - ボールの座標
  - ターゲットの座標
  - ターゲットの速度
  - ターゲットの存在フラグ
  - 時間経過を管理する変数
  - ほかにある？
- その変数によって世界を描画してみる！
- 変数に, 計算などによって値を入れてみる
- 跳ね返るといふことはどういう計算があればよい？

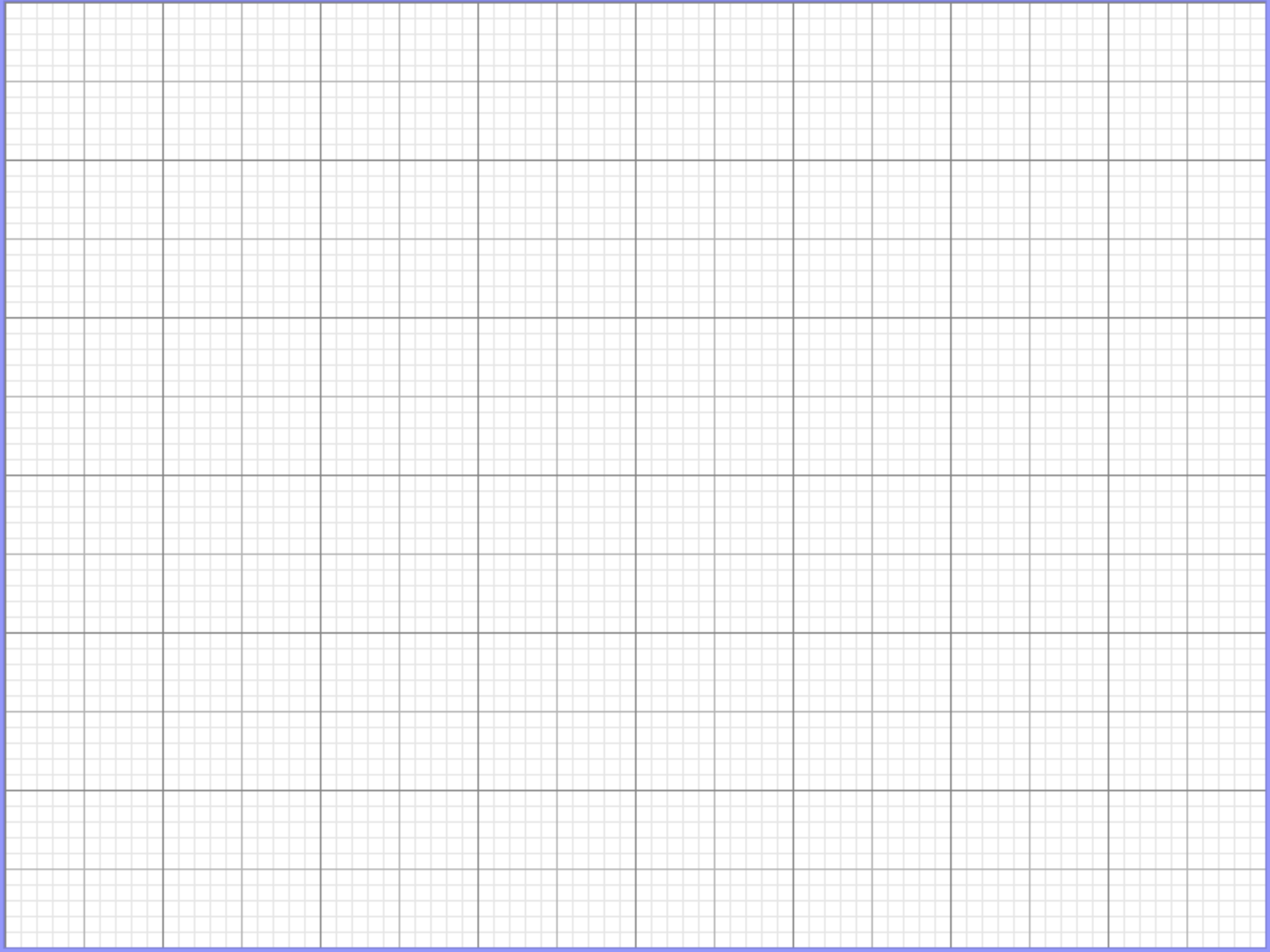
# プログラミング演習I (第4回) 課題

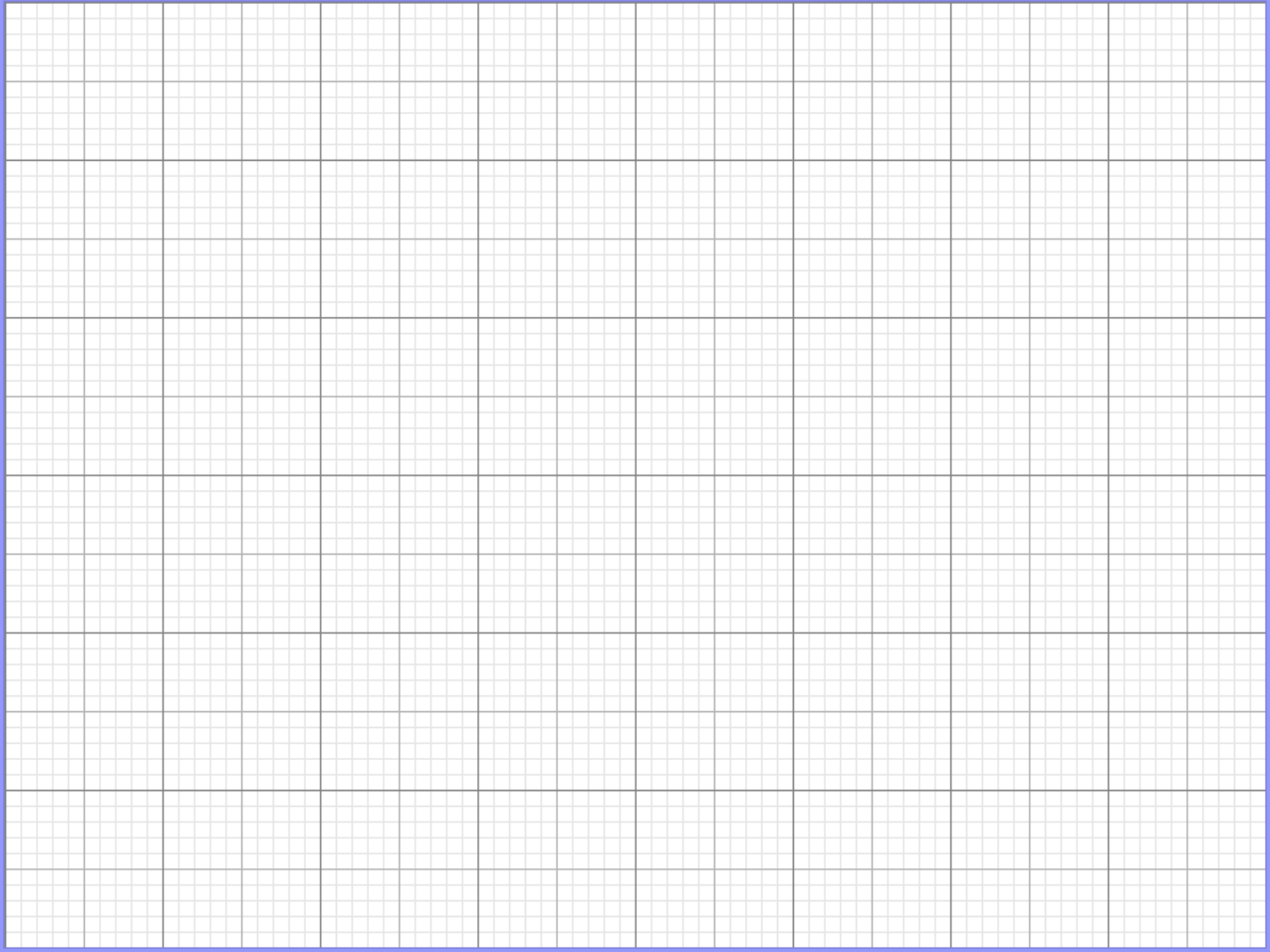
## • 発展課題② スケッチ名: unlock

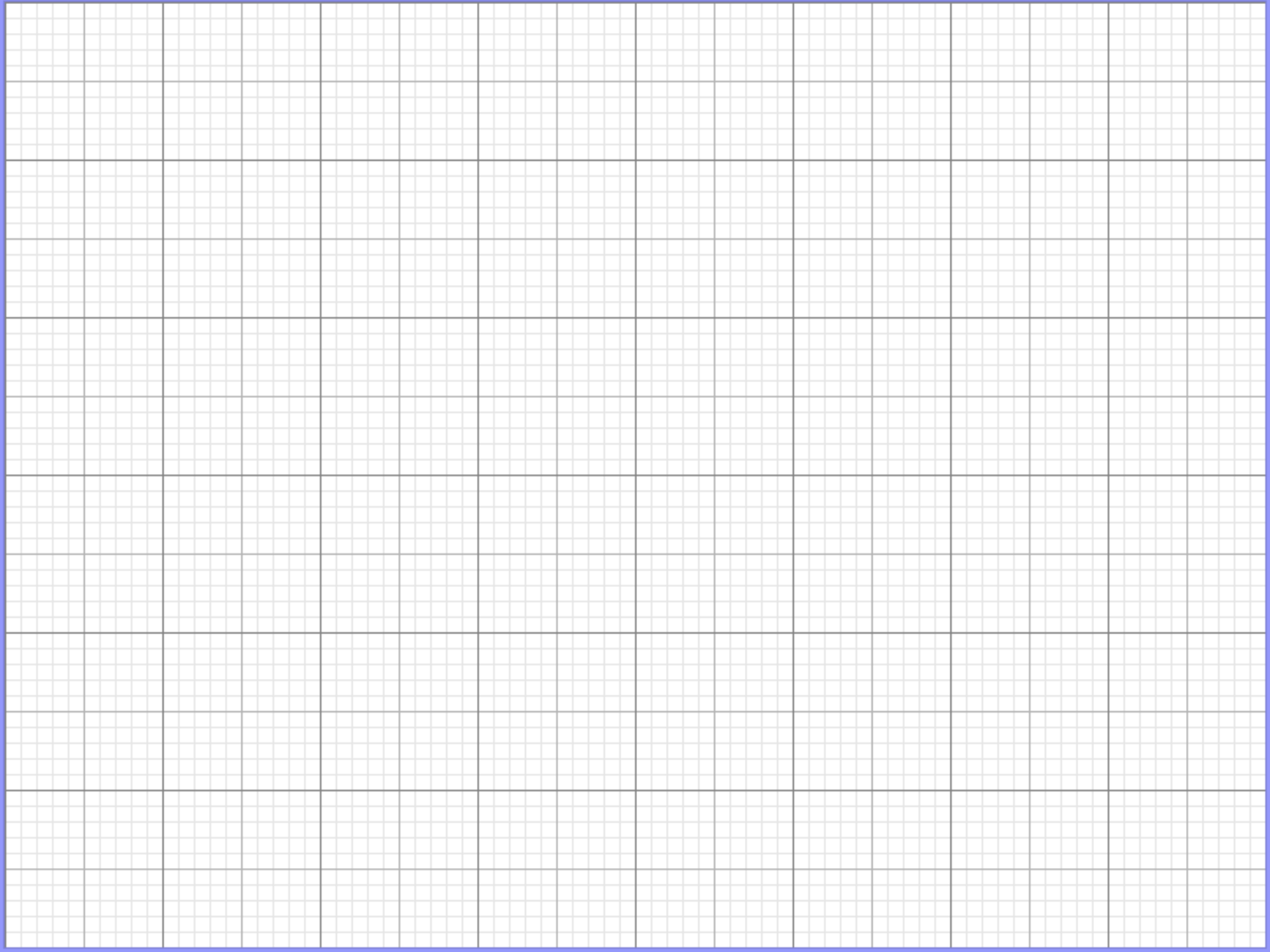
- 縦長のウィンドウの下部に左の方に丸型のものを用意し、それを右側にスライド(スワイプ)させると、ロックが解除され何らかの画面に遷移する仕組みを実現せよ(例えば、作成したキャラクタが描画されている画面など)
- なお、右端まで移動せずに手を離れた場合は、5ピクセル/フレームの速度で最初の位置に戻るようにせよ。また、指定の位置より右や左にはみ出ないようにせよ。

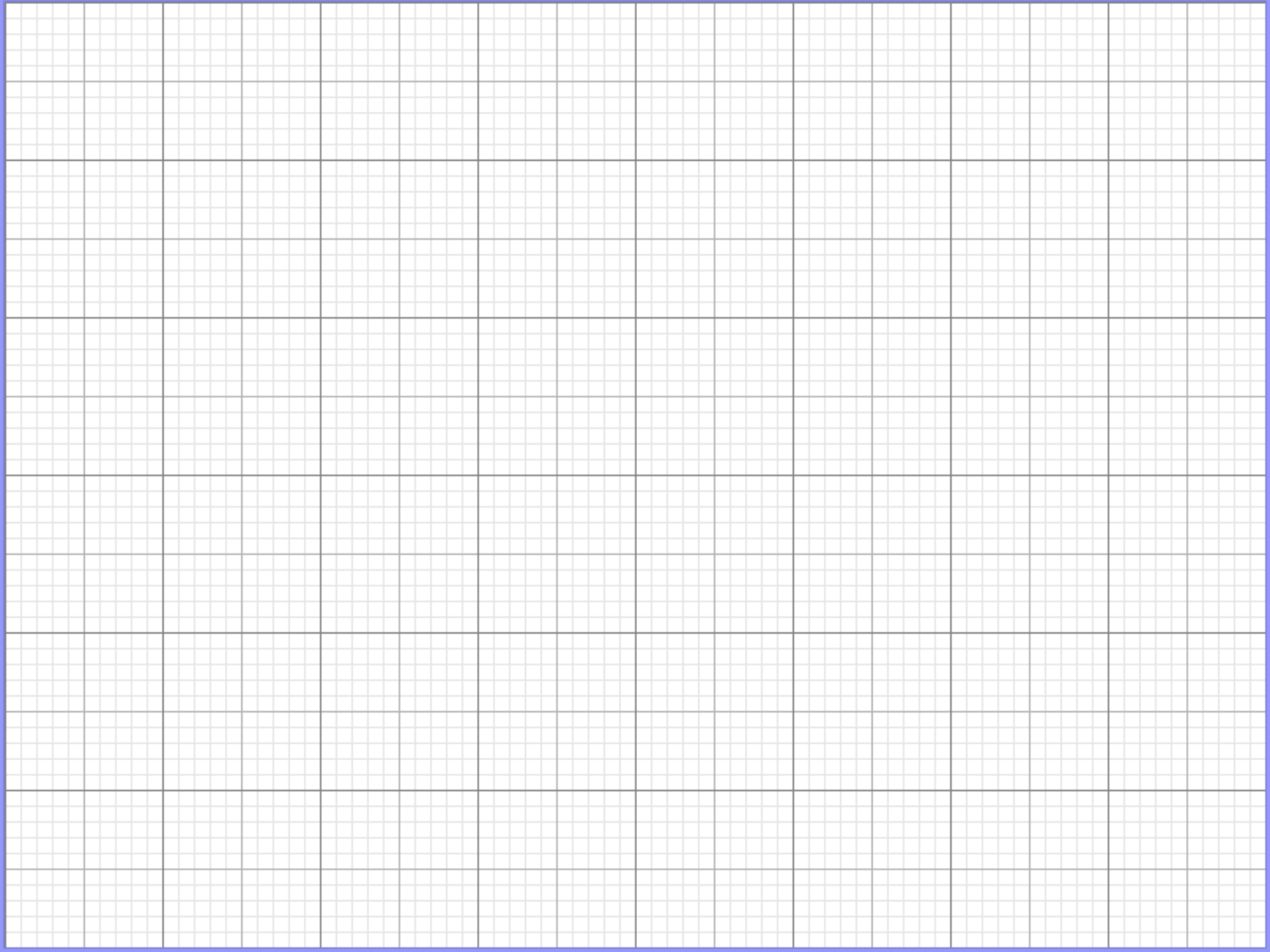


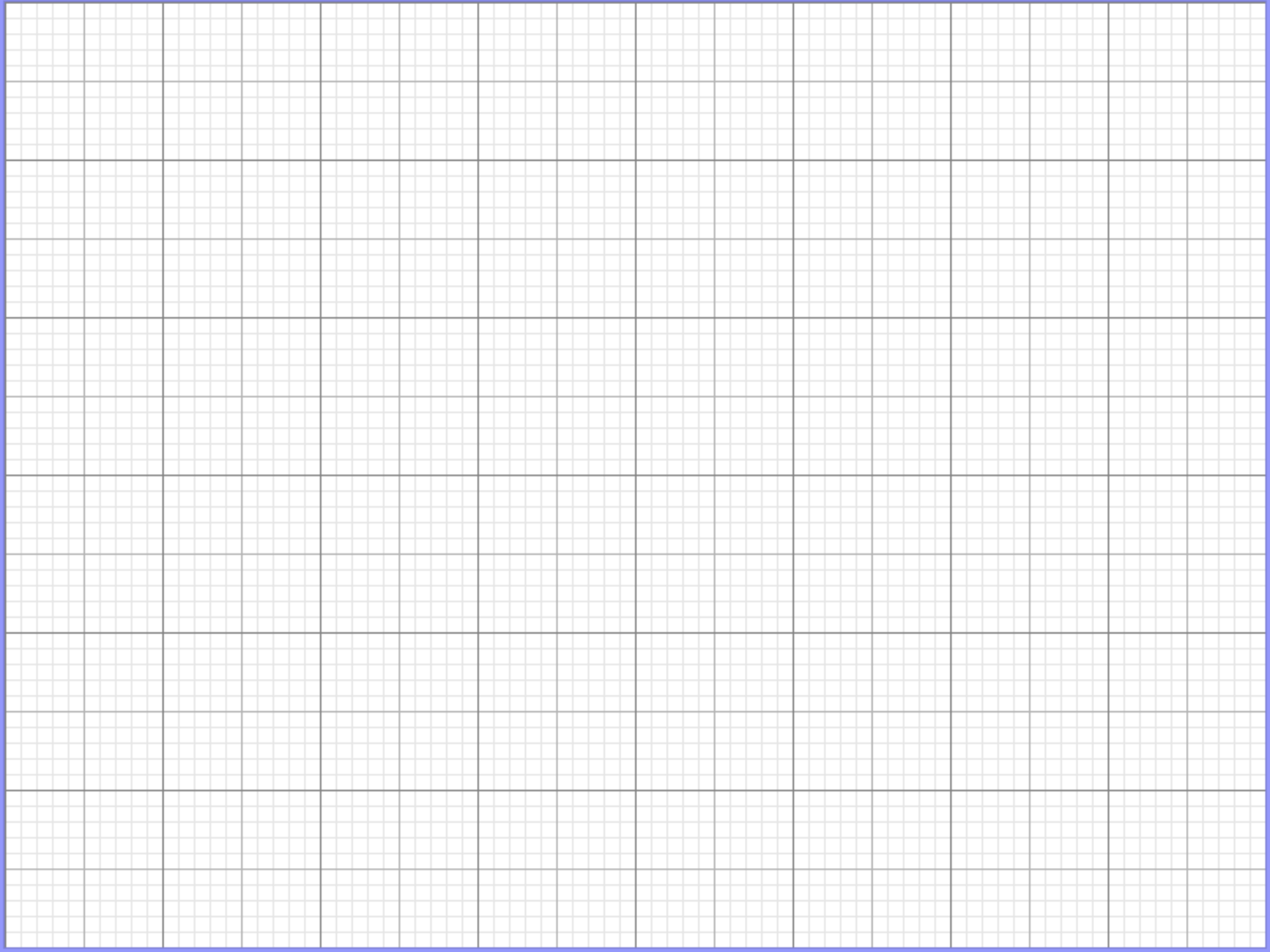












# 【重要】課題の提出方法

---

- プログラムの冒頭にコメントを必ず書くこと！
  - プログラム名とその目的(日本語でも英語でもOK)
  - プログラムの作成日(例: 2016/06/08)
  - プログラムの作成者(年-組-番号-名前)
  - 追加で他の情報を記載してもOK

```
// Olympic.pde
// Draw the Olympic rings
// 2016/06/08
// X-Y-Z Satoshi Nakamura
background( 255, 255, 255 );
fill( 255, 0, 0 );
:
```

# プログラミング演習I (第4回) 課題

---

- 必要な変数

- 丸型の部分の中心座標
- 丸型の部分を選択されているかどうかのフラグ
- 他に何かある？

- 利用するメソッド

- mousePressed() マウスのボタンを押す
  - 丸型の部分の選択の判定とフラグ付与
- mouseReleased() マウスのボタンを離す
  - 丸型の部分の選択のフラグを離す