

【重要】課題の提出方法

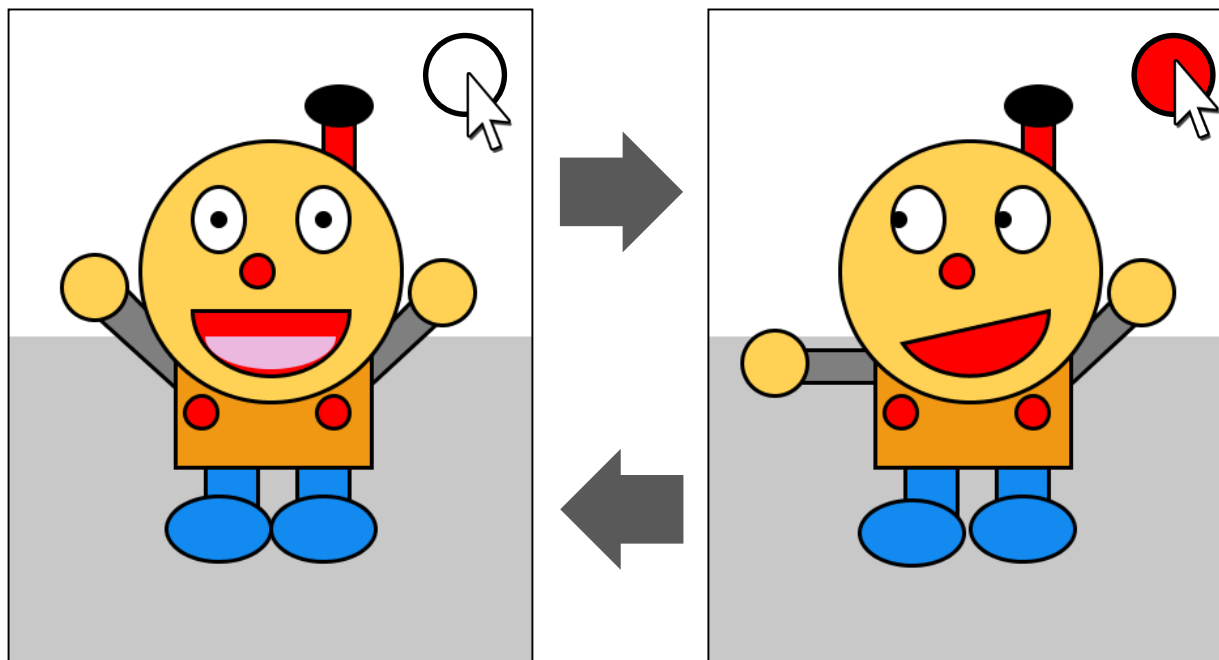
- プログラムの冒頭にコメントを必ず書くこと！
 - プログラム名とその目的（日本語でも英語でもOK）
 - プログラムの作成日（例：2015/06/08）
 - プログラムの作成者（年-組-番号-名前）
 - 追加で他の情報を記載してもOK

```
// Olympic.pde
// Draw the Olympic rings
// 2015/06/08
// 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

• 基本課題① スケッチ名 : charapara

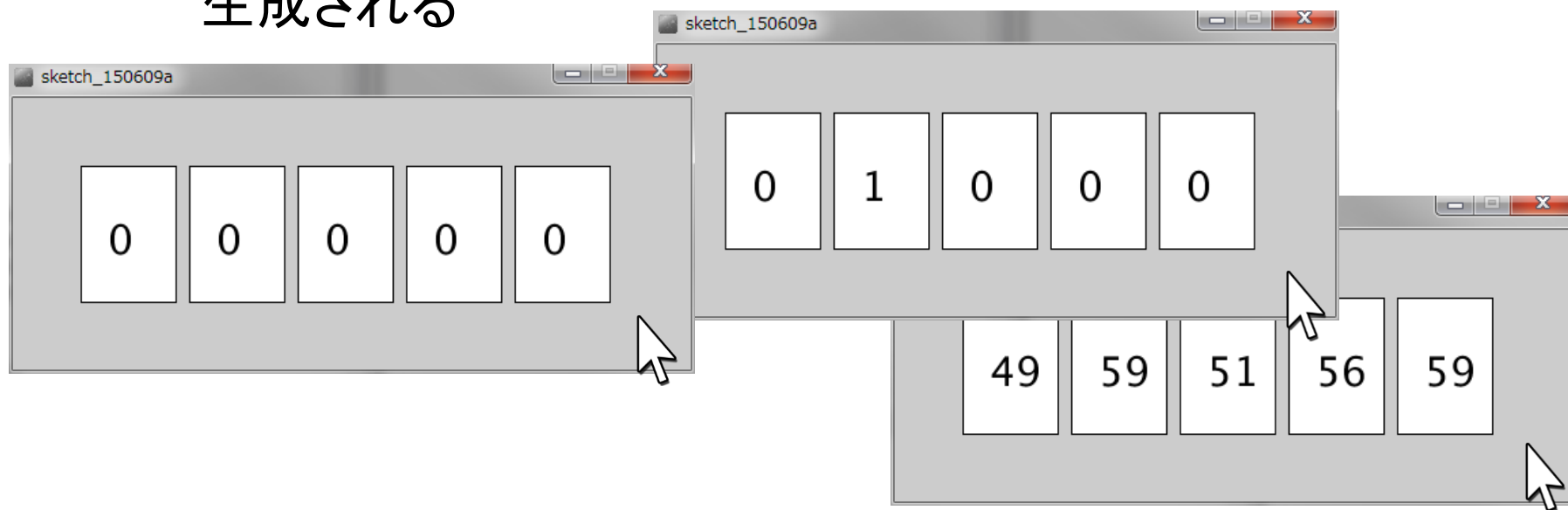
- キャラクタを表示し、画面右上に用意した直径50以下の丸ボタンをクリックすることで絵がパラパラと切り変わるプログラムを作れ。また、丸ボタンはクリックの度に白色、赤色と切り替わるようにせよ。丸ボタンの外をクリックした場合は反応しなくなるようにせよ。
- 前々回のchara1とchara2のプログラムをうまく再利用しよう。



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：gacha

- クリックする度に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表示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
 - 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



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)

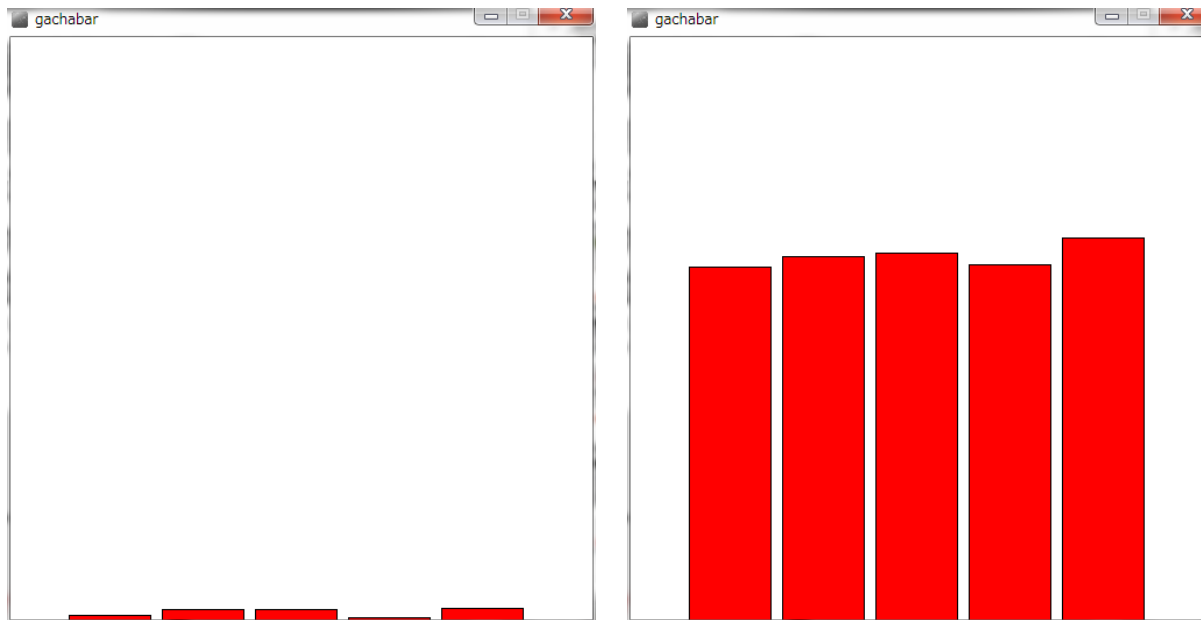
10以上20以下の整数値をランダムに発生する方法は

```
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③ スケッチ名：gachachart

- gachaプログラムを改良し，500x500のウィンドウとし、drawが呼び出される度に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



プログラミング演習I（第4回）課題

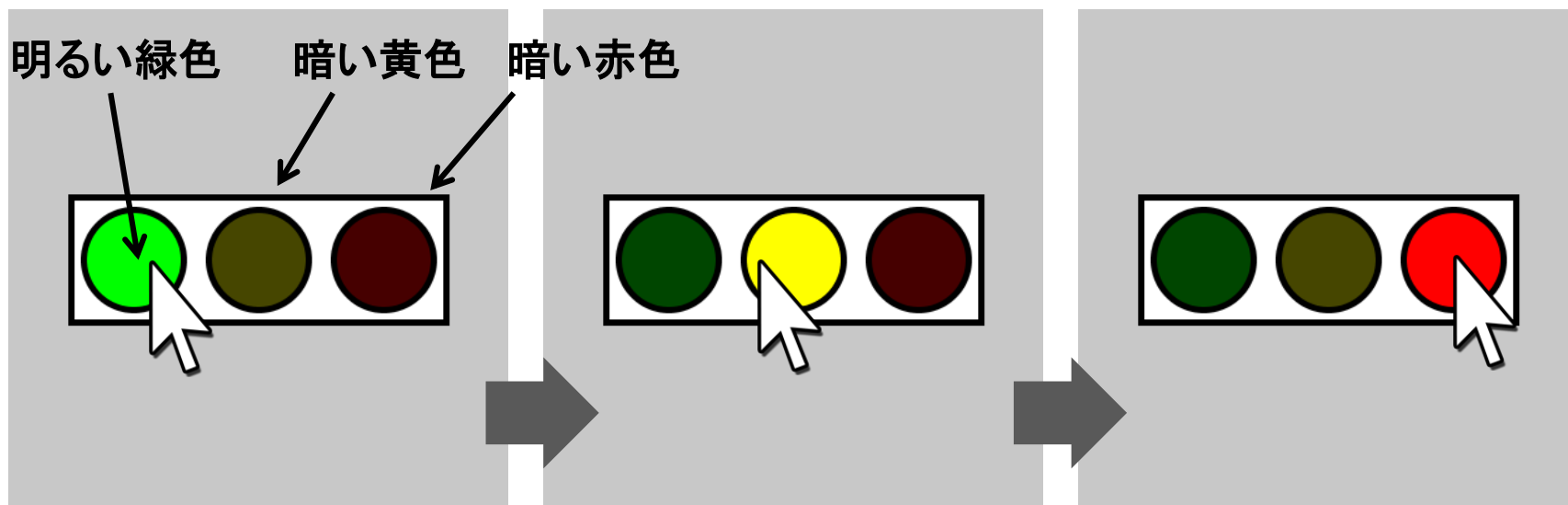
- ヒント

- 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）

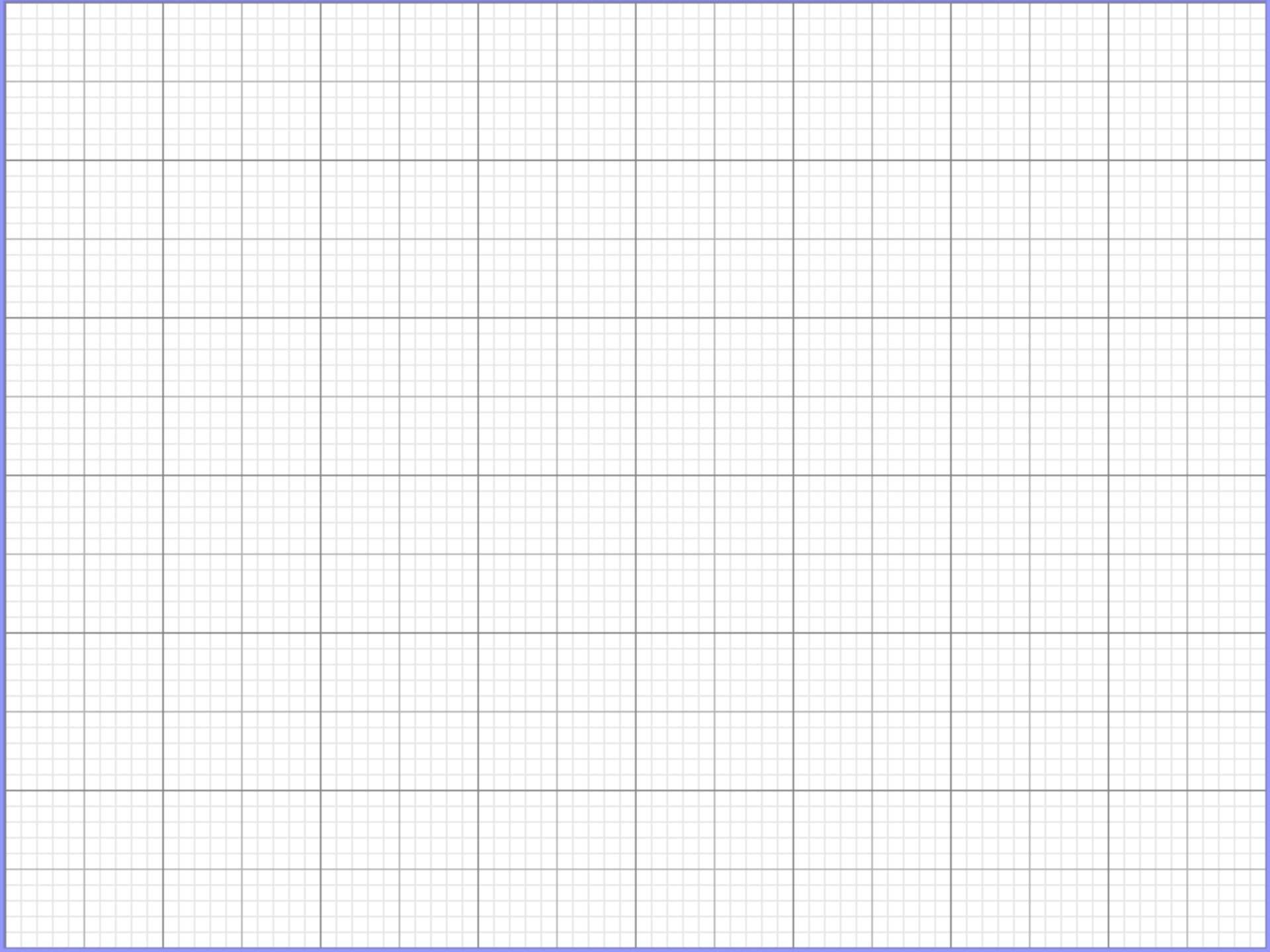
プログラミング演習I（第4回）課題

• 基本課題④ スケッチ名 : signal

- 信号機のプログラムを作成せよ。ただし、画面を左右に3等分し、マウスカーソルが左側の丸内にある場合は緑色、中央の丸内にある場合は黄色、右側の丸内にある場合は赤色が明るく灯るようにし、そうでない色は暗くなるようにせよ。



マウスカーソルの位置に応じて色が変わる！



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

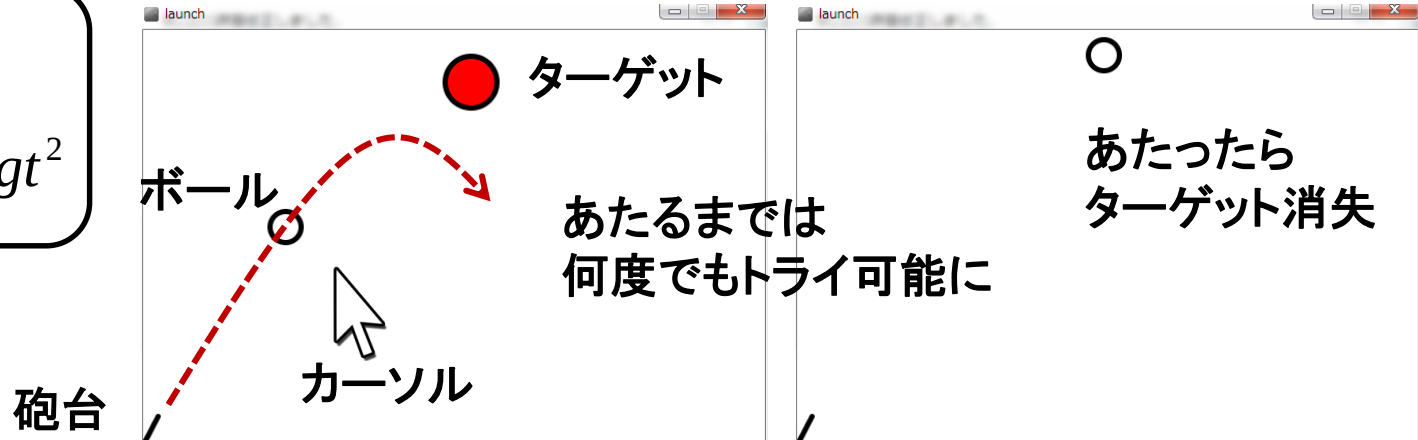
- 600x400のウィンドウ左下からマウスカーソルがある方向にボール（半径15m）を発射する砲台のプログラムを作ってください。クリックされたタイミングで投射されるようにすること。また、クリック時の砲台からカーソルまでの距離を初速とせよ。さらに、350mの高さに半径25mの赤丸のターゲットを描け。赤丸は左右に3m/フレームで動き、左端右端で折り返すようにせよ。なお、ボールがターゲットに衝突すると、ターゲットを消すようにせよ。1フレームは0.1秒とする。

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



ヒント

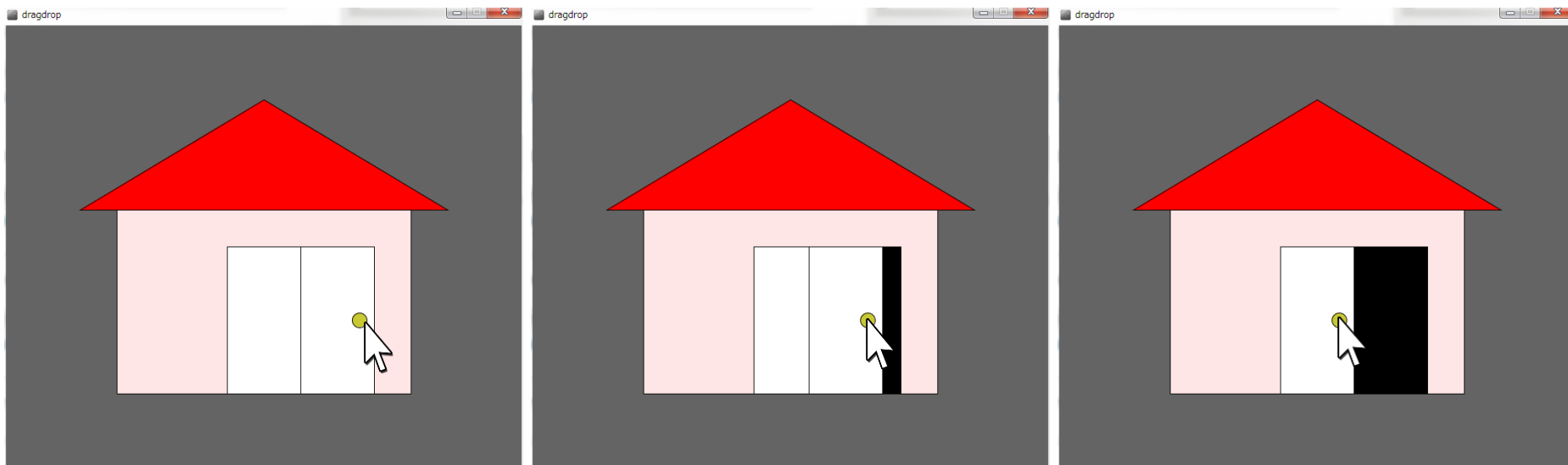
- この世界を実現する上で必要な変数(変化するものを取りあえず片っ端から)をまず列挙！
 - マウスの座標
 - 砲台の角度
 - ボールの投射角度
 - ボールの初速
 - ボールの座標
 - ターゲットの座標
 - ターゲットの速度
 - ターゲットの存在フラグ
 - 時間経過を管理する変数
 - ほかにある？
- その変数によって世界を描画してみる！
- 変数に, 計算などによって値を入れてみる
- 跳ね返るということはどういう計算があればよい？

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 : dragdrop

- 適当な大きさの灰色のウィンドウに100×200の白色の扉を2枚並べたものをもつ家を描き(家の形や扉の位置、大きさや色は任意)、片方の扉に取っ手を描き(半径60)、取っ手をカーソルで掴みながら動かすと下図のように扉をスライドして開ける事のできるプログラムを作成せよ。なお、取っ手を掴んだらそれ以降は取っ手からずれても良いものとする。また、動かす扉を最初の位置から右へや左の扉を超えて動かないようにする事

(課題案：豊崎富心)



行き過ぎても左端でとまる

プログラミング演習I（第4回）課題

- 必要な変数

- ドアの手取の中心座標
- ドアの手取が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
 - ドアの手取の選択の判定とフラグ付与
- mouseReleased() マウスのボタンを離す
 - ドアの手取の選択のフラグを離す

