明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

プログラミング演習 (11)
クラス

中村，橋本，小松，渡辺

目標

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Processing でオブジェクト指向（クラス）に挑戦
  - 配列でX座標，Y座標，それぞれの速度を管理していたのはなんだか変な感じ
  - それぞれに値をもたせて独立して動かしたい
  - 何らかの機能を持つオブジェクトを1つのクラスとしてまとめる！

オブジェクト指向とは

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- ざっくり説明すると，色々な値や機能をもつもの

  （例）シューティングゲーム上の敵

  - 現在位置（X座標，Y座標）
  - 何らかの移動機能（移動の関数）
  - 何らかの描画機能（描画の関数）

  をもっており，プログラムから移動しろ，描画しろと命令を送るだけで，その中身がどうなっているかを意識せずに利用可能

- 他人が何をどう考え実行するかを気にせず，「～をやっておいて」とお願いする感覚

クラス

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class  クラス名 {
    変数色々

    クラス名( 引数リスト ){
        // オブジェクトが作られた時にのみ実行
    }

    関数色々
}
```

XY座標をもつ円を作る

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
class Circle {
    float  posX; // x座標
    float  posY; // y座標

    // 最初にだけ呼び出される
    Circle( float x, float y ){
        posX = x;
        posY = y;
    }
}
```

クラスの定義

```
void draw(){
    background( 255 );
    ellipse( c.posX, c.posY, 20, 20 );
}
```

変数名.内部変数

```
Circle  c;
void setup(){
    size( 400, 400 );
    c = new Circle( random(400), random(400) );
}
```

new で作成する！

XY座標をもつ3つの円を作る

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
Circle  c1;
Circle  c2;
Circle  c3;
void setup(){
    size( 400, 400 );
    c1 = new Circle( random(400), random(400) );
    c2 = new Circle( random(400), random(400) );
    c3 = new Circle( random(400), random(400) );
}
void draw(){
    background( 255 );
    ellipse( c1.posX, c1.posY, 20, 20 );
    ellipse( c2.posX, c2.posY, 20, 20 );
    ellipse( c3.posX, c3.posY, 20, 20 );
}
```

new で作成する！

変数名.内部変数

クラス＋配列

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
Circle  c [] = new Circle [10];
void setup(){
    size( 400, 400 );
    int  i=0;
    while( i<10 ){
        c[i] = new Circle( random(400), random(400) );
        i++;
    }
}
void draw(){
    background( 255 );
    for( int  i=0; i<10; i++ ){
        ellipse( c[i].posX, c[i].posY, 20, 20 );
    }
}
```

配列全体をnewで作成

それぞれ new する！

配列の要素毎に描画

for でも while でも ok

XY座標をもつ円を作る

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
class Circle {
    float  posX; // x座標
    float  posY; // y座標
    float  speedX;
    float  speedY;

    // 最初にだけ呼び出される
    Circle( float x, float y, float sx, float sy ){
        posX = x;
        posY = y;
        speedX = sx;
        speedY = sy;
    }
}
```

スピードも追加してみる

移動を入れると

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室


```
Circle  c [] = new Circle [10];
void setup(){
    size( 400, 400 );
    for( int  i=0; i<10; i++ ){
        c[i] = new Circle( random(400), random(400),
                                        random(5), random(5) );
    }
}
void draw(){
    background( 255 );
    for( int  i=0; i<10; i++ ){
        c[i].posX = c[i].posX + c[i].speedX;
        c[i].posY = c[i].posY + c[i].speedY;
        ellipse( c[i].posX, c[i].posY, 20, 20 );
    }
}
```

スピードは0～5

描画の前に移動

うーん

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- あまり便利になっていない
- オブジェクト指向でクラスを使うことの良さは，移動とか，描画とかをクラス自体に任せてしまうこと

- 移動する，描画するということをクラス内で関数として定義してみる！

```
class Circle {
    float  posX; // x座標
    float  posY; // y座標
    float  speedX; // x座標方向のスピード
    float  speedY; // y座標方向のスピード
    Circle( float x, float y, float sx, float sy ){
        posX = x;
        posY = y;
        speedX = sx;
        speedY = sy;
    }
    void  move(){
        posX = posX + speedX;
        posY = posY + speedY;
    }
    void  paint(){
        ellipse( posX, posY, 20, 20 );
    }
}
```

移動のための関数

描画のための関数

クラスに関数を追加

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

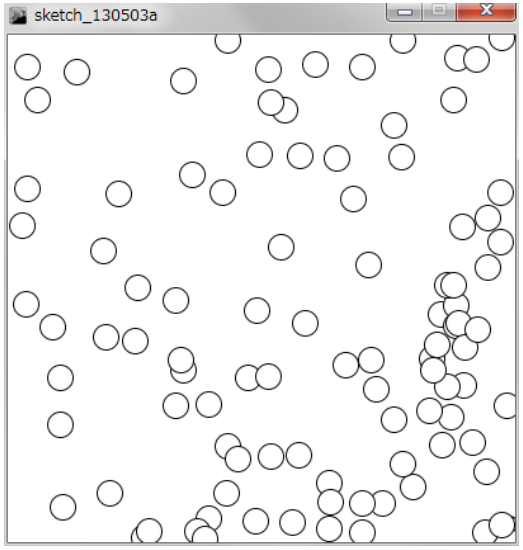

```
Circle  c [] = new Circle [10];
void setup(){
    size( 400, 400 );
    for( int  i=0; i<10; i++ ){
        c[i] = new Circle( random(400), random(400),
                                        random(5), random(5) );
    }
}
void draw(){
    background( 255 );
    for( int  i=0; i<10; i++ ){
        c[i].move();
        c[i].paint();
    }
}
```

書くのはこれだけ！

予習問題

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 現在10個の円ですが，円の数を100個に変更してみましょう

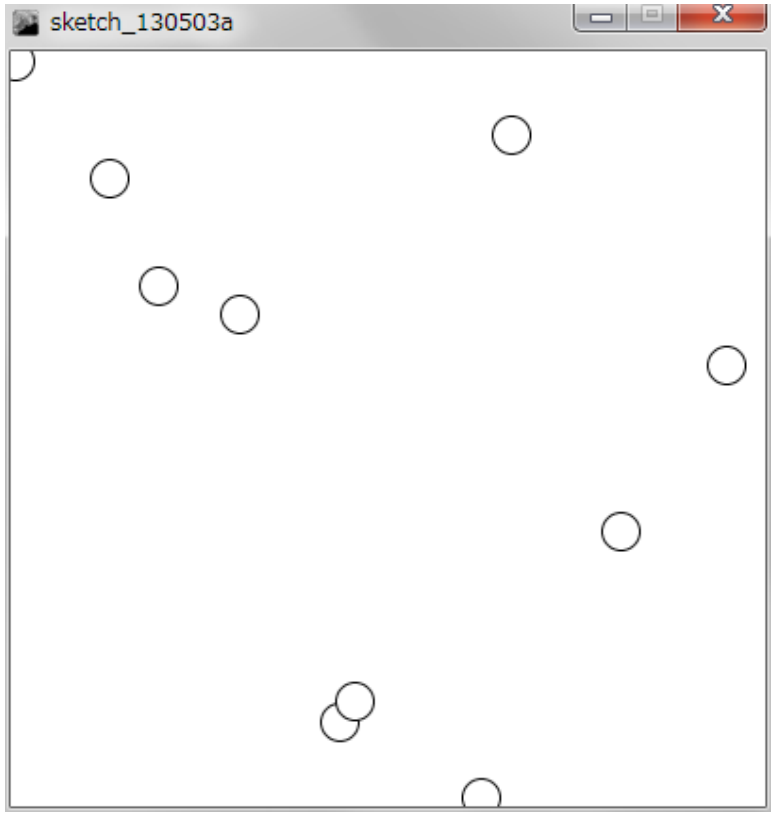


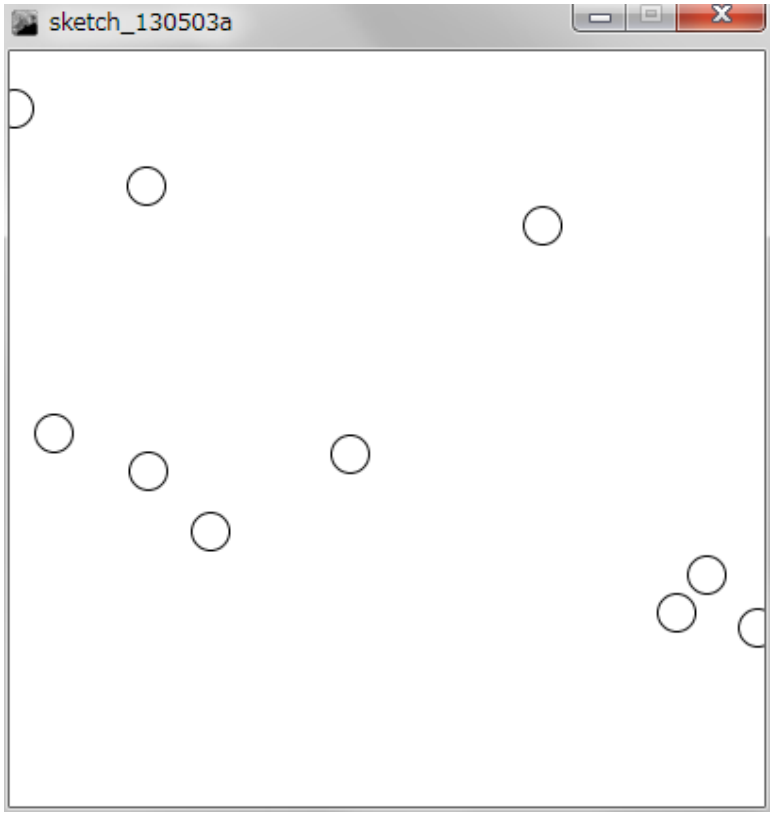
- Circleクラスで表現される円は画面の端から外に飛んでいってしまうので，端まで来たら逆から出てくるようにしよう！
  - move() で条件分岐

跳ね返りを実現する

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q) 画面端まで移動したら逆側から出てくるのではなく，右端に着いたら左に，下端に着いたら上に跳ね返るようにするには？





跳ね返りを実現する

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 考え方
  - move() の中で,
    - 画面左端(posX≦0)で右に折り返す speedX = -speedX;
    - 画面上端(posY≦0)で下に折り返す speedY = -speedY;
    - 画面右端(posX≧400)で右に折り返す speedX = -speedX;
    - 画面下端(posY≧400)で右に折り返す speedY = -speedY;

    （注意）スピードによっては端を行き過ぎることもあるため行き過ぎた分は端に戻す

    画面右端は width という変数，下端は height という変数を使うことも可能
  - あとは特に気にしないでOK!!

跳

```
void move() {
    posX += speedX;
    posY += speedY;
    if ( posX > width ) {
        posX = width;
        speedX = -speedX;
    } else if ( posX < 0 ) {
        posX = 0;
        speedX = -speedX;
    }
    if ( posY > height ) {
        posY = height;
        speedY = -speedY;
    } else if ( posY < 0 ) {
        posY = 0;
        speedY = -speedY;
    }
}
```

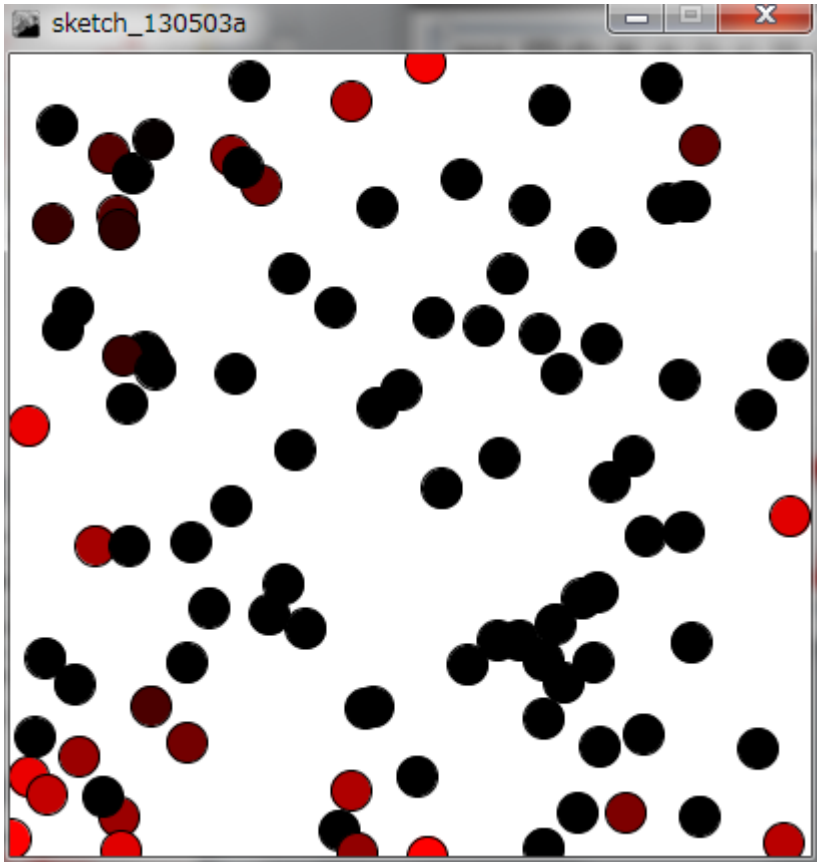
posXでの条件分岐

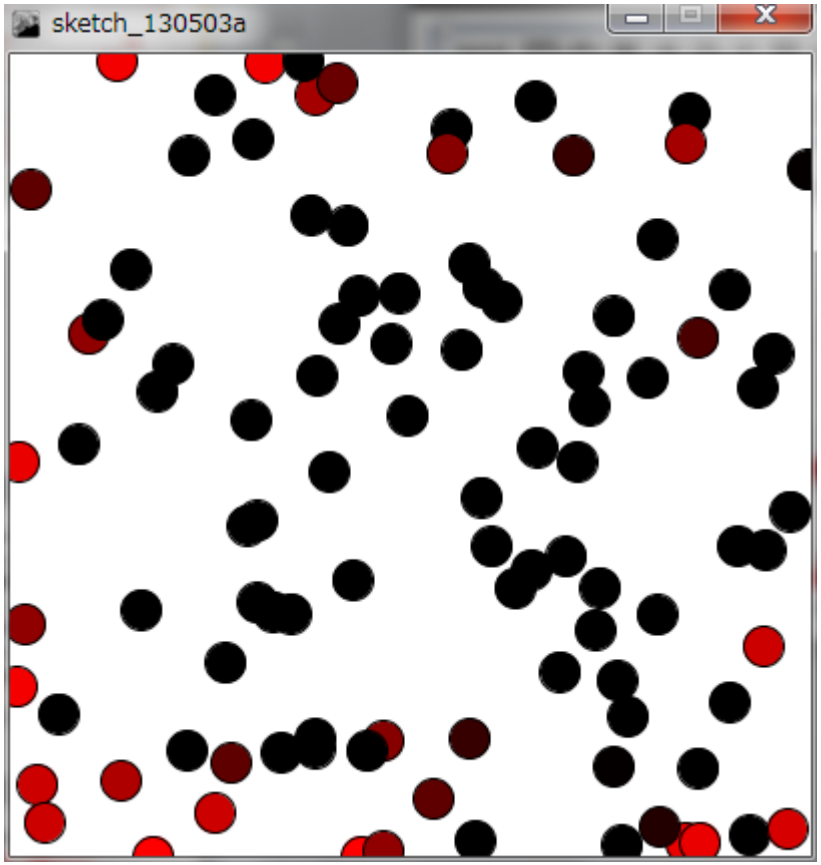
posYでの条件分岐

色を変化させる

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q) 円の基本色を黒色にし，画面の端に衝突したら塗り色を赤色に変えるようにするには？





色を変化させる

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 考え方
  - 黒色はRGBで(0, 0, 0)と赤色はRGBで(255, 0, 0)なので，衝突したらRの量を255にし，徐々に0へと近づけていく
  - 円の描画 paint() 内で fill 関数を利用して塗りつぶし色を指定する
    - fill(fillR, 0, 0);
  - 赤色の塗りつぶし量を記憶する整数型の fillR という変数を用意し，初期値は 0 にしておく
  - 衝突したら fillR の値を 255 にする
  - 描画される度に，fillR が 0 より大きければ fillR を徐々に減算する(-10ずつ，-5ずつなど)

```
class Circle {
    float  posX;
    float  posY;
    float  speedX;
    float  speedY;
    int  fillR;
    Circle( float x, float y, float sx, float sy ){
        posX = x;
        posY = y;
        speedX = sx;
        speedY = sy;
        fillR = 0;
    }
    void  paint(){
        fill( fillR, 0, 0 );
        ellipse( posX, posY, 20, 20 );
        if( fillR > 0 ){
            fillR -= 5;
        } else if( fillR < 0 ){
            fillR = 0;
        }
    }
```

```
    void move() {
        posX += speedX;
        posY += speedY;
        if ( posX > width ) {
            posX = width;
            speedX = -speedX;
            fillR = 255;
        } else if ( posX < 0 ) {
            posX = 0;
            speedX = -speedX;
            fillR = 255;
        }
        if ( posY > height ) {
            posY = height;
            speedY = -speedY;
            fillR = 255;
        } else if ( posY < 0 ) {
            posY = 0;
            speedY = -speedY;
            fillR = 255;
        }
    }
}
```

予習問題

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 先述のプログラムを改良し，端に衝突すると黒くなり，徐々に薄くなるようにしましょう
  - ヒント：白は（255, 255, 255），黒は（0, 0, 0）

- 棒人間クラスを作ってみよう
  - 棒人間クラスに移動と，描画の機能を導入しよう
  - 棒人間が端まで行くとしばらく止まり，再度別方向に動き出すようにしてみましょう
    - 動かない時間をカウントする変数 wait などを導入！

予習問題

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 過去に作ったロボットをクラス化しましょう！
  - ロボット全体をクラスにして，動かす！と命令を送るだけで，柔軟に動くようにしましょう
- 過去に作ったゲームでそれぞれのオブジェクトをクラス化しましょう！
  - 敵と見方の文理など

次回予告

明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 自由制作！