

プロセツシング 入門4

初歩のプログラミング

ウィンドウを使った関数版バブルソート

```
P b_sort_text2 | Processing 2.2.1
File Edit Sketch Tools Help

b_sort_text2

int [] a = {2,3,5,7,9,4,8,1,6,0};
int count = 0;

void setup() {
  size(250, 250);
  frameRate(1);
}

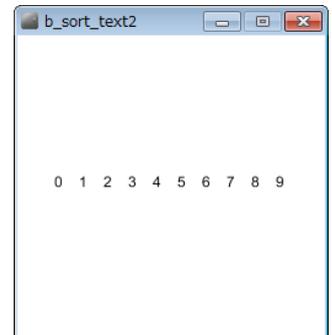
void draw() {
  bubble(count);
  background(255);
  fill(0); // character color
  for (int j = 0; j < a.length; j++) {
    text(a[j], 30+20*j, height/2);
  }
  count++;
}

void bubble(int i){
  for (int j = a.length-2; j>=i; j--) {
    if(a[j+1] < a[j]){
      int tmp = a[j];
      a[j] = a[j+1];
      a[j+1] = tmp;
    }
  }
}
```

1秒に1回表示する。でないとラウンド目のソート結果が確認できない。

ウィンドウの座標(30+20*j,height/2)にa[j]の値を表示する。

countラウンド目のバブルソートを行うという意味



ウィンドウを使った関数版バブルソート(2)

```
b_sort_graphic
int [] a = new int [10];
color [] b = new color [20];
int count = 0;

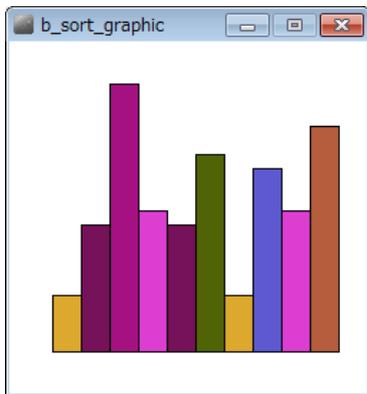
void setup() {
  size(250, 250);
  frameRate(1);
  set_data();
  set_color();
  //background(255);
}

void draw() {
  bubble(count);
  background(255);
  for (int j = 0; j < a.length; j++) {
    fill(b[a[j]]);
    rect(30+20*j, 220-10*a[j], 20, a[j]*10);
  }
  count++;
  print(count);
  if(count==a.length){
    set_data();
    count = 0;
  }
}

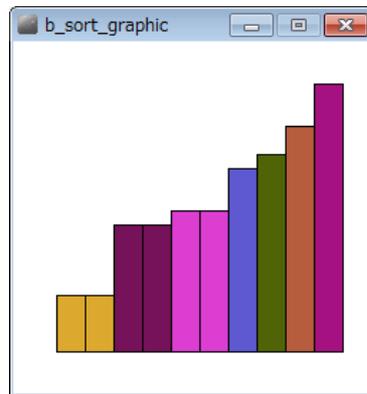
void bubble(int i){
  for (int j = a.length-2; j>=i; j--) {
    if(a[j+1] < a[j]){
      int tmp = a[j];
      a[j] = a[j+1];
      a[j+1] = tmp;
    }
  }
}
}
```

ウィンドウを使った関数版バブルソート(2)

```
void set_data(){  
    for(int i = 0; i < a.length; i++){  
        a[i] = (int)random(20);  
    }  
}  
  
void set_color(){  
    for(int i = 0; i < 20; i++){  
        b[i] = color((int)random(255), (int)random(255), (int)random(255));  
    }  
}
```



ソート前

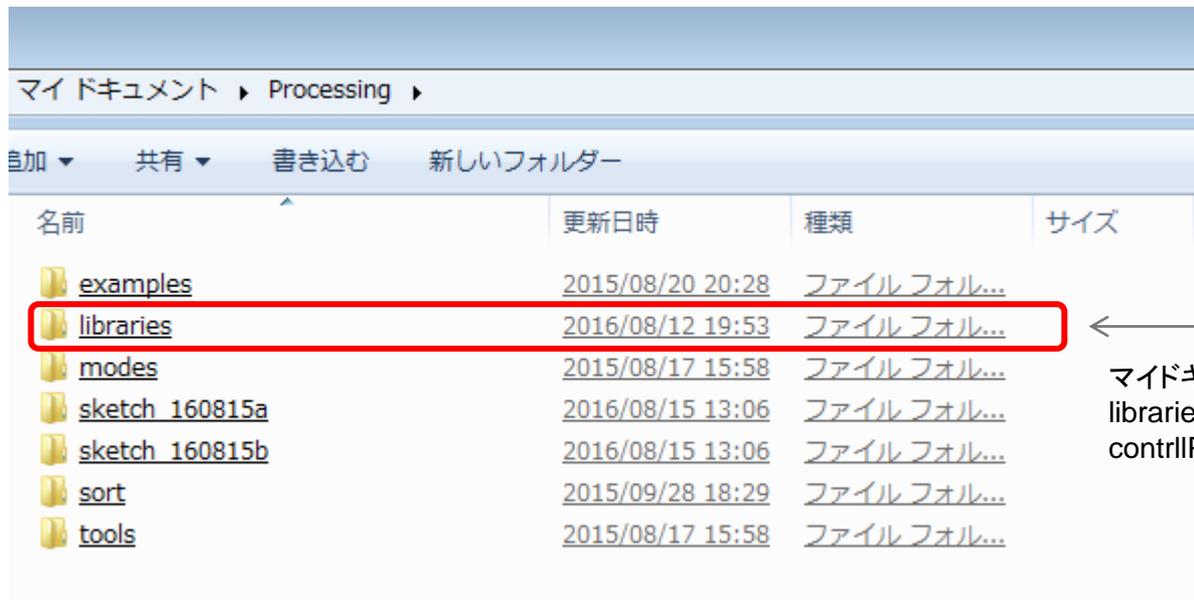


ソート後

提出期限:10月7日(金)16:30

問 データの数を100,1000にしたものを作成せよ.
framerate()は間延びしないよう適当に変更してよい.
できた人は校内メールでプログラムを送付.
基礎電磁気学演習の点に加算します.

ライブラリの使用



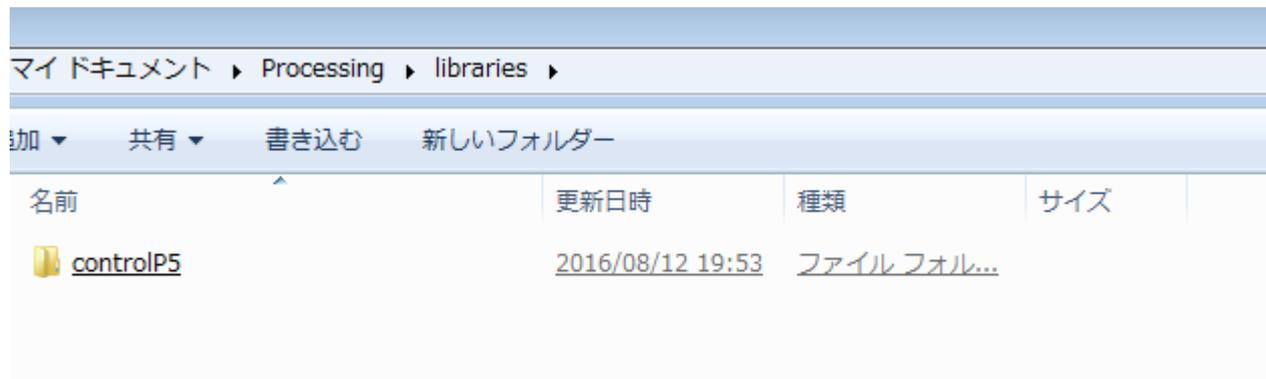
マイドキュメント ▾ Processing ▾

追加 ▾ 共有 ▾ 書き込む 新しいフォルダー

名前	更新日時	種類	サイズ
examples	2015/08/20 20:28	ファイルフォル...	
libraries	2016/08/12 19:53	ファイルフォル...	
modes	2015/08/17 15:58	ファイルフォル...	
sketch_160815a	2016/08/15 13:06	ファイルフォル...	
sketch_160815b	2016/08/15 13:06	ファイルフォル...	
sort	2015/09/28 18:29	ファイルフォル...	
tools	2015/08/17 15:58	ファイルフォル...	



マイドキュメントの下のProcessingフォルダの下のlibrariesにcontrolIP5を置きます。
controlIP5は私のページのcontrolIP5.zipからダウンロード



マイドキュメント ▾ Processing ▾ libraries ▾

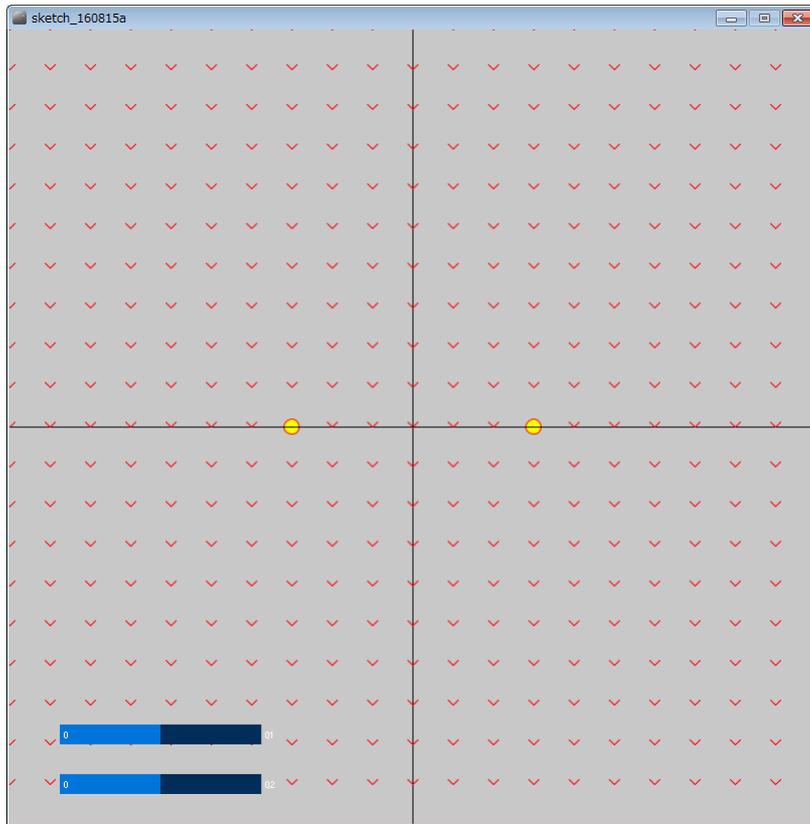
追加 ▾ 共有 ▾ 書き込む 新しいフォルダー

名前	更新日時	種類	サイズ
controlIP5	2016/08/12 19:53	ファイルフォル...	

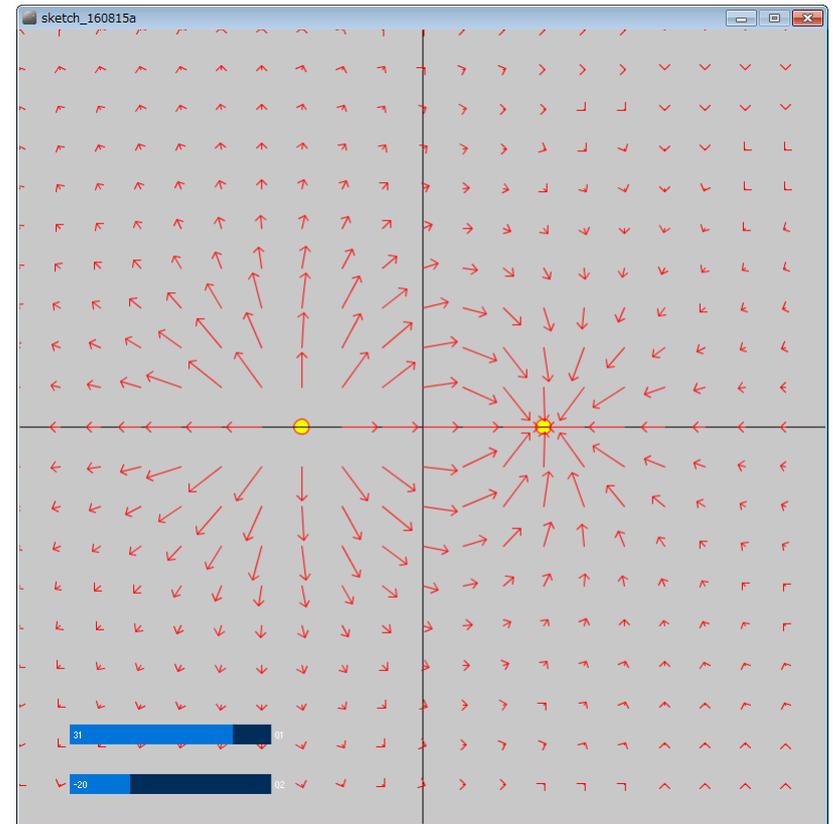
こうなっていればOK

ライブラリの使用(2)

ライブラリのスライダーをつけて2つの荷電粒子の電荷量を変化できるようにしてみます。シミュレーターっぽくなります。



初期画面 ($Q_1=Q_2=0$)



スライダー操作後

ライブラリの使用(3)

setup()とdraw()は次のように変更する. arrow()は変更の必要はない.

```
sketch_160815a
import controlP5.*;
ControlP5 cp5a,cp5b;

float Q1_f, Q2_f;
int Q1 = 0, Q2 = 0;

void setup(){
  size(800,800);
  //noLoop();
  cp5a = new ControlP5(this);
  cp5a.addSlider("Q1")
  .setPosition(50,700) ← スライダーの配置座標
  .setRange(-50,50) ← スライダーのとり値の範囲
  .setSize(200, 20); ← スライダーのサイズ
  cp5b = new ControlP5(this);
  cp5b.addSlider("Q2")
  .setPosition(50,750)
  .setRange(-50,50)
  .setSize(200, 20);
}
```

ライブラリの使用(3)

draw()は次のように変更する.

```
void draw(){
    float x,y;
    float r1,r2;
    //arrow(10,200,30,40);
    background(200);
    fill(255,255,0);
    ellipse(280, 400, 15, 15);
    ellipse(520, 400, 15, 15);
    stroke(0);
    line(0,400,800,400);
    line(400,0,400,800);

    Q1_f = (float)Q1*20000; Q2_f = (float)Q2*20000; ← スライドバーの値を利用する

    for(int i = 0; i < 20; i++){
        for(int j = 0; j < 20; j++){
            if((40*i == 280 && 40*j == 400) || (40*i == 520 && 40*j == 400)) continue;
            r1 = sqrt((float)(pow((40*i-280),2)+pow((40*j-400),2)));
            r2 = sqrt((float)(pow((40*i-520),2)+pow((40*j-400),2)));
            x = Q1_f*((float)(40*i-280)/(r1*r1*r1))
              + Q2_f*((float)(40*i-520)/(r2*r2*r2));
            y = Q1_f*((float)(40*j-400)/(r1*r1*r1))
              + Q2_f*((float)(40*j-400)/(r2*r2*r2));
            //println(x,y);
            if(x*x+y*y>35*35){
                x = 35*x/sqrt(x*x+y*y);
                y = 35*y/sqrt(x*x+y*y);
            }

            arrow(40*i,40*j,(int)(40*i+x),(int)(40*j+y));
        }
    }
}
```

ライブラリの使用(4)

問 スライダーをつけた2つの荷電粒子のシミュレーションを完成せよ. **できた人は**校内メールでプログラムを送付. 基礎電磁気学演習の点に加算します.

問 スライダーをつけた3つの荷電粒子のシミュレーションを作成せよ. 3つの配置は正三角形とする. **できた人は**校内メールでプログラムを送付. 基礎電磁気学演習の点に加算します.