



プログラミング演習2

ファイル入出力

高橋, 中村, 小林, 橋本

他人の物は極力触らない



このウイルス、どうやって感染するんだっけ？

ウイルスが入るのは目や鼻や口からである。手や足の皮膚とかから感染するものではない。だからウイルスが顔の粘膜に触れるのを避けることが大事。

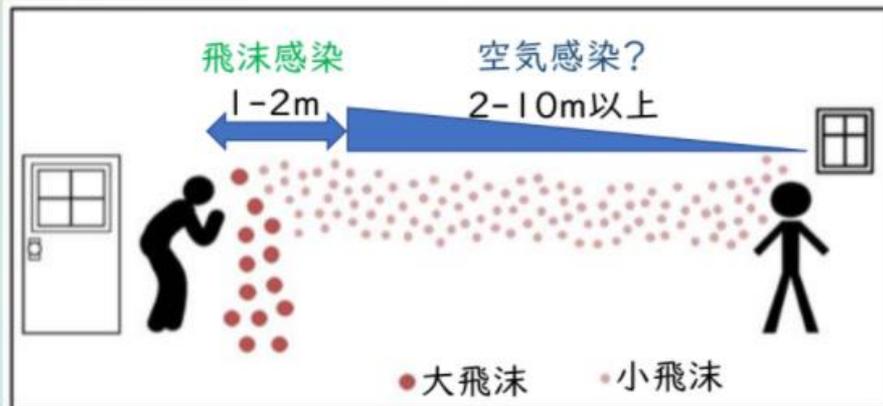
するよ

接触感染

環境のウイルスが付いている部分に触って、その手を口や鼻に触れて感染する

飛沫感染

くしゃみや咳などの飛沫を浴びて口や鼻にかかることで感染。射程距離1-2mくらい。
エアロゾル感染といって、空気の流れの悪いところでは数時間くらいウイルスが浮いてて感染する。(マイクロ飛沫も。)



しないよ

空気感染

ウイルス自体が空気中をどこまでもぶかぶか浮いてて感染する。特徴としては、全く接点のない人が感染するなど。
(注)飛沫感染との境目の解釈が曖昧なところがあります。

距離を取ろう！



飛沫感染を避けるために



くしゃみや咳などの飛沫は約2mくらいでほとんど床に落下します。
基本的に落下したものは、もう触れなければ感染性は消失しますので、飛んでいる飛沫を吸入しなければいいのです。

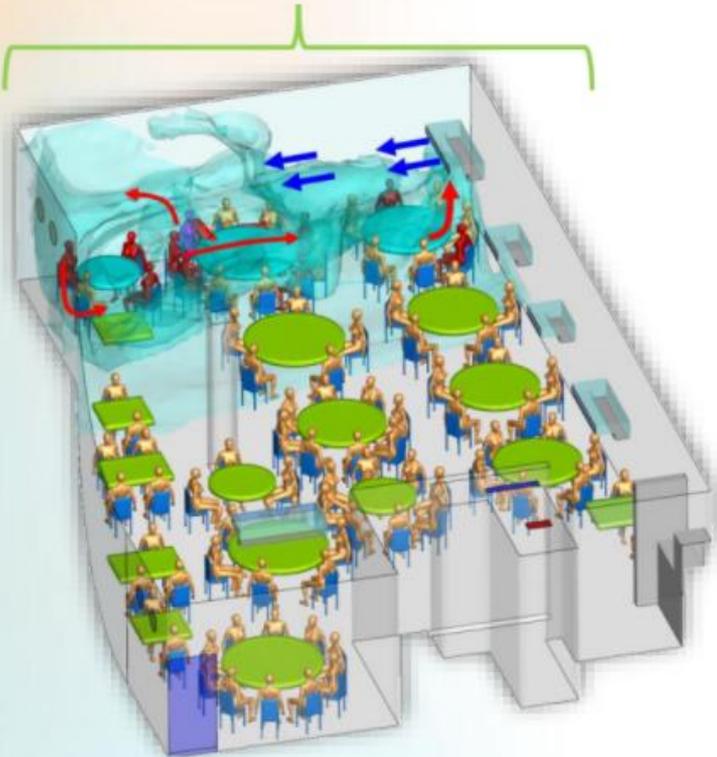
1m離れるだけでもだいぶリスクは下がります。2m離れればかなりリスクは下がります。
マスクをしていない時はその距離が保てればいいのですね。

換気も重要



さらに!換気はすごい大事

同じレストランで一部分だけ空気が滞留して患者集団発生があった。



とにかく換気!!!これが一番!
距離が取れないような部屋なら、いかに換気をするかどうかに全てかかっています。

つまり

- ・人が触るものはできるだけ触らない。触ったら手を洗う。(特に顔を触る前)
- ・人と2m位の距離を保つ。保てないならマスクをすることでリスクを下げられる。
- ・屋外ではマスクがなくても距離があれば換気がいいので感染リスクは低い

今後について



- 2020/11/09～
 - 対面で実施します（203, 206, 310, 311, 312）
- 2020/11/23（重要）
 - フィジカルコンピューティングのキットを配布しますので，カバンを忘れずに！
- 2020/11/30
 - フィジカルコンピューティング（小林先生）
- 月1限または月2限にEnglish IB（リアルタイム型のオンライン授業）を中野キャンパスで受講したい学生用に，515・516を取っていますので活用ください

今日のスケジュール



- 今後の運用（ネットワークとフィジカル）に向けて、最初に余裕をもったスケジュールにします
 - 13:30-13:50 部屋割とかもろもろ
 - 13:50-14:30 作ったもの&課題を互いに紹介
 - 14:30-15:00 今日の解説
 - 15:00-16:45 課題に取り組む時間
 - 16:45-17:00 課題解説の時間



- 最終回はグループワークの成果の発表会の予定です
- グループワークでは、研究室を複数グループ（人数によって変動）に分けますが、誰と組むことになるかはわかりませんので、みんなで助け合いレベルアップしましょう
- ということで、まずはお互いの力量を把握するため、作ったもの・課題を紹介し合しましょう（再履修の学生さんたちは今日は現研究室に入ってください）
 - 自分の研究室がもしなければ教員・TAまで



- 作った模倣を研究室で発表してください
 - 感染リスクを考え、発表者がみんなから2m離れるよう移動し、Discordで画面を共有して発表
 - Discordで画面共有できない場合はディスプレイをみんなに向ける
 - 他のメンバーはDiscordの画面共有を見ながら、話を聞く。
 - 発表が終わったら拍手！
 - だいたいひとりあたり3分程度で回して行ってください。時間が余ったら、作った課題についても紹介して行ってください！

今日やること



- 前回やった記録をどう再現する？
- ハイスコアをどう記録する？
- 実験などの記録をどう再現する？

どうやって再現する？



- 前回起動していたときの状態を引き継ぐには、どうしたらよいだろうか？

おきのどくですが
ほうけんのしょ 1ばんは
きえてしまいました。

記録したり呼び出したり



- これまで

- 変数に値を保存（代入）し，変数を使うことで値を取り出し表示したりしていた

- Ballのx, y座標と, スピード
- ゲームのscore
- 電光掲示板のライトのON/OFF情報

アプリケーションを再起動したら消えてしまう



ファイルに記録して再起動後にも使う！

ファイルを経由する



- プログラムとは別のファイルに一旦情報を記録しておき、そのファイルからまた情報を読み出す！
- セーブ&ロードしよう！
 - 記録する
 - 何かの値をファイルに記録しておく（ハイスコアや状態など）
 - 読み込む（再生する）
 - 記録した値をファイルから呼び出す（ハイスコアや状態など）

ファイルから読み込む



```
String[] lines = loadStrings("ファイル名");
```

- ファイルの中身を1行毎にString配列に格納
 - 1行目の値は `lines[0]` に, 2行目の値は `lines[1]` に入っている
- **int(文字列)** で, 文字列を整数に変換

```
String[] lines = loadStrings("list.txt");  
int name = lines[0];  
int cost = int( lines[1] );
```

ファイルから読み込む



Windows

C:¥ どこかのフォルダ **¥** どこかのフォルダ **¥** なんかのファイル

Mac OS

/ Users **/** nakamura **/** どこかのフォルダ **/** なんかのファイル

```
lines[0] = こんにちは みさなん おんげき ですか？  
lines[1] = わしたは げんき です。  
lines[2] = この ぶんよしう は いりぎす の ケブンツリジ  
lines[3] = だがいく の けゆきんう の けっか  
lines[4] = にんんげ は もじ を にしんき する とき その  
lines[5] = さしいよ と さいご の もさじえ あいてっれば  
lines[6] = じばんゆん は めくちちゃや でも ちんやと  
lines[7] = よめる という けゆきんう に もづいとて わざと  
lines[8] = もじの じんばゆん を いかれえて あまりす。  
lines[9] = どでうす？ ちんやと よやちめう でしょ？  
lines[10] =  
lines[11] = ちんやと よためら はゆしくを よしろく
```

上から順に読み込む

ファイルに書き込む



```
saveStrings("ファイル名", String型の配列);
```

- String型の配列の内容を, 1行毎にファイルに書き込む
 - 例えば, Stringの配列をlinesとしたときに, lines[0]は1行目に, lines[1]は2行目に保存される
 - **str(何らかの値)** で整数などを文字列に変換

文字列保存演習



- 百人一首からひとつ歌をもってきて、プログラムで5,7,5,7,7を1行ずつuta.txtに保存するプログラムを作る
 - 5つの要素からなる文字列型の配列 `saveLines` を作成し、`uta.txt` に保存しましょう

<code>saveLines[0]</code>	花の色は
<code>saveLines[1]</code>	移りにけりな
<code>saveLines[2]</code>	いたづらに
<code>saveLines[3]</code>	わが身世にふる
<code>saveLines[4]</code>	ながめせし間に

文字列保存演習



```
// 5行分なので5つの要素からなる文字列配列を作る
```

```
String[] saveLines = new String[5];
```

```
// 1行ずつ値を放り込む
```

```
saveLines[0] =
```

```
saveLines[1] =
```

```
saveLines[2] =
```

```
saveLines[3] =
```

```
saveLines[4] =
```

```
// saveStringsでsaveLinesを1行ずつuta.txtに保存します
```

```
saveStrings("uta.txt", saveLines);
```

数値を保存する



- 現在の状態をファイルに保存
 - $x = 10$
 - $y = 20$
- ファイル (data.txt) に例えば下記のように保存

10

20

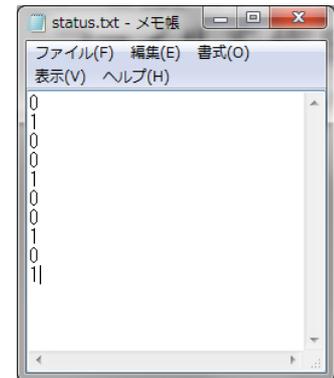
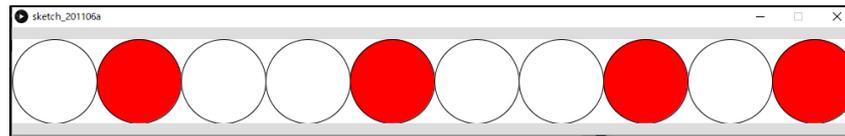
- 1行目は x , 2行目は y の値として読み込む

電光掲示板



1000x1000のウィンドウに横に10個，縦に1個並んだ電光掲示板について，丸をクリックする度に赤色，白色と塗りつぶしの色が入れ替わるようにせよ．またファイルから赤白の状態を読み込むようにせよ

- メモ帳で，1行に0または1だけを書いた10行のファイルを作成する（ファイル名はstatus.txt）
 - PDEと同じフォルダに保存する
- status.txt をプログラムで読み込み，1行目をlights[0]に，2行目をlights[1]にと値を順に割り当てる
- lightsの値に応じて描画





```
int[] lights = new int[10];
```

```
void setup()
```

```
{  
  size(1000, 100);  
  for(int x=0; x<10; x++){  
    lights[x] = 0;  
    x++;  
  }  
}
```

```
void draw()
```

```
{  
  background( 255 );  
  for(int x=0; x<10; x++){  
    if(lights[x] == 1){  
      fill(255, 0, 0);  
    } else {  
      fill(255);  
    }  
    ellipse(100*x+50, 50, 100, 100);  
  }  
}
```

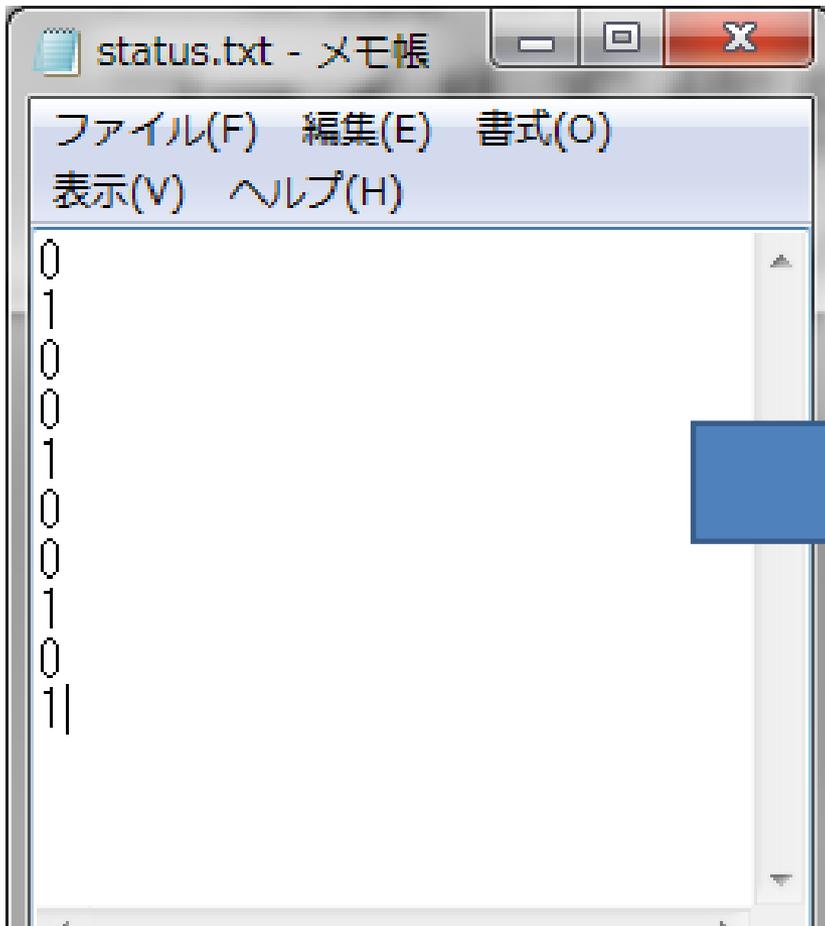
```
void mousePressed(){  
  for(int x=0; x<10; x++){  
    if(dist(100*x+50,50,mouseX,mouseY) <= 50){  
      if(lights[x] == 1){  
        lights[x] = 0;  
      } else {  
        lights[x] = 1;  
      }  
    }  
  }  
}
```

電光掲示板プログラム



- ファイルから読み込む

```
String [] lines = loadStrings("status.txt");
```



配列	値
lines[0]	"0"
lines[1]	"1"
lines[2]	"0"
lines[3]	"0"
lines[4]	"1"
lines[5]	"0"
lines[6]	"0"
lines[7]	"1"
lines[8]	"0"
lines[9]	"1"

電光掲示板プログラム



setup()のみを変更してファイルから読み込む

```
int [] lights = new int [10];

void setup() {
  size( 300, 100 );
  String[] lines = loadStrings( "status.txt" );
  lights[0] = int( lines[0] );
  lights[1] = int( lines[1] );
  lights[2] = int( lines[2] );
  lights[3] = int( lines[3] );
  lights[4] = int( lines[4] );
  lights[5] = int( lines[5] );
  lights[6] = int( lines[6] );
  lights[7] = int( lines[7] );
  lights[8] = int( lines[8] );
  lights[9] = int( lines[9] );
}
```



```
int[] lights = new int [10];

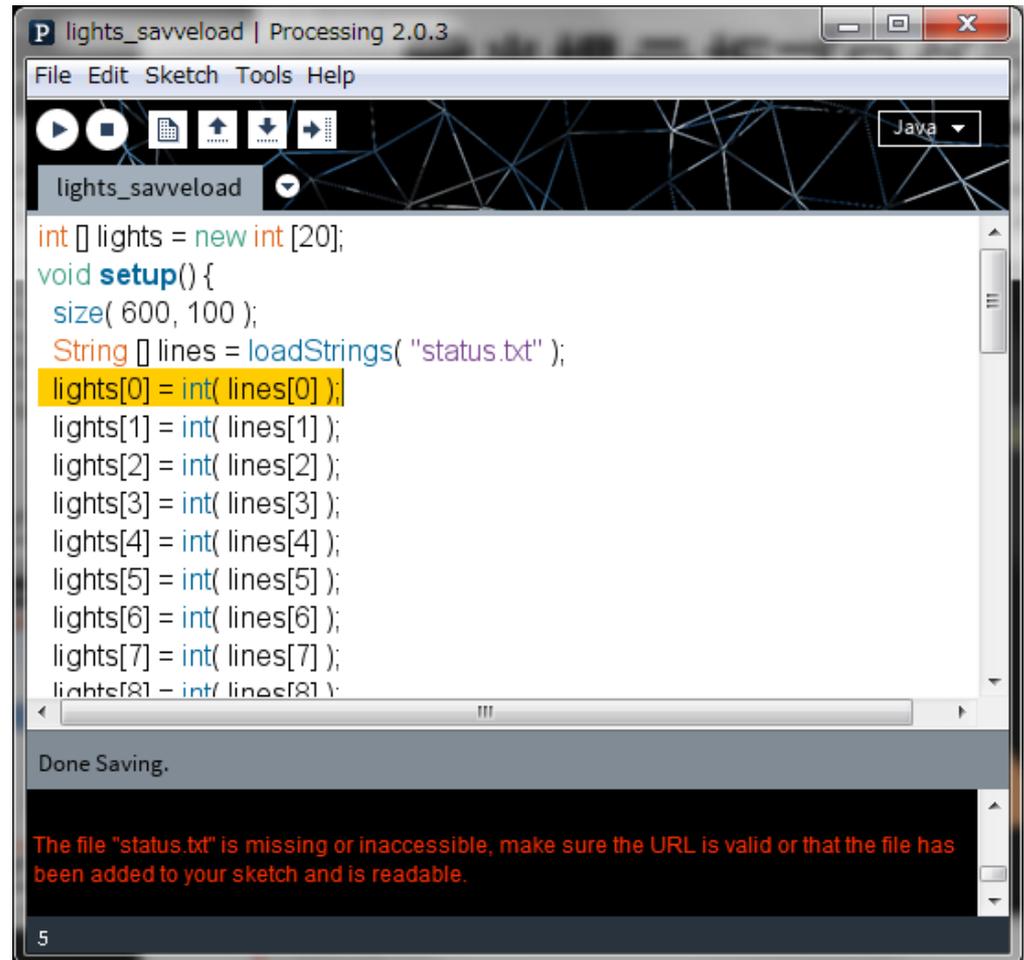
void setup() {
  size( 300, 100 );
  String [] lines = loadStrings( "status.txt" );

  for(int x=0; x<10; x++){
    lights[x] = int( lines[x] );
  }
}
```

こんなエラーが出たら



- status.txt がPDEと同じフォルダに入っていないということ



```
lights_savveload | Processing 2.0.3
File Edit Sketch Tools Help
lights_savveload
int [] lights = new int [20];
void setup() {
  size( 600, 100 );
  String [] lines = loadStrings( "status.txt" );
  lights[0] = int( lines[0] );
  lights[1] = int( lines[1] );
  lights[2] = int( lines[2] );
  lights[3] = int( lines[3] );
  lights[4] = int( lines[4] );
  lights[5] = int( lines[5] );
  lights[6] = int( lines[6] );
  lights[7] = int( lines[7] );
  lights[8] = int( lines[8] );
  lights[9] = int( lines[9] );
  lights[10] = int( lines[10] );
  lights[11] = int( lines[11] );
  lights[12] = int( lines[12] );
  lights[13] = int( lines[13] );
  lights[14] = int( lines[14] );
  lights[15] = int( lines[15] );
  lights[16] = int( lines[16] );
  lights[17] = int( lines[17] );
  lights[18] = int( lines[18] );
  lights[19] = int( lines[19] );
}

Done Saving.

The file "status.txt" is missing or inaccessible, make sure the URL is valid or that the file has
been added to your sketch and is readable.

5
```

電光掲示板プログラム



```
void mousePressed(){
  for(int x=0; x<10; x++){
    if(dist(100*x+50, 50, mouseX, mouseY) <= 50){
      if(lights[x] == 1){
        lights[x] = 0;
      } else {
        lights[x] = 1;
      }
    }
  }
}

String[] saveLines = new String[10];
for(int x=0; x<10; x++) {
  saveLines[x] = str( lights[x] );
}
saveStrings("status.txt", saveLines);
}
```

電光掲示板プログラム



```
void mousePressed(){
  for(int x=0; x<10; x++){
    if(dist(50+100*x, 50, mouseX, mouseY) <= 50){
      // 1,0,1,0と切り替えるならこの方法でもOK
      lights[x] = 1 - lights[x];
    }
  }
}

String[] saveLines = new String[10];
for(int x=0; x<10; x++) {
  saveLines[x] = str( lights[x] );
}
saveStrings("status.txt", saveLines);
}
```

課題1: basic_click_count



ウィンドウ上でクリックした回数を表示するプログラムを作成せよ。なお、クリックした回数を保存し、次回起動した時にそのクリック回数から増やしていくようにせよ

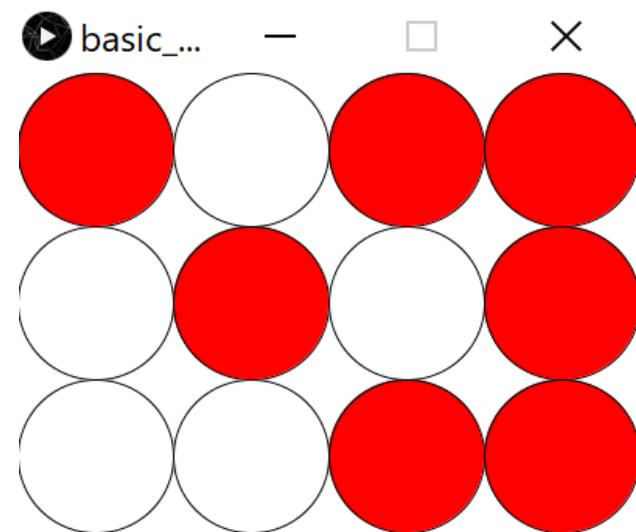
- `count.txt` というファイルを作成
- `click_count` という変数を用意して、`mousePressed()` の度に値を増やす
- `mousePressed()` の度に、`click_count` の値を `count.txt` に保存する
- 起動時 (`setup`) の時に、`count.txt` の値を `click_count` に読み込む

課題2 basic_lights43



電光掲示板のプログラムを改良して縦に3個、横に4個となるように変更し、左下のように記述したstatus.txtから状態を読み込み下図のように出力せよ。また、値を変えて動作を確認せよ。さらに、クリックのたびに保存し、次回の起動時に結果を引き継ぐようにせよ

```
1,0,1,1
0,1,0,1
0,0,1,1
```



課題2 basic_lights43



ヒント

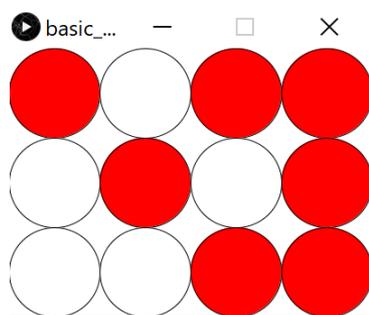
- `status.txt` というファイルをまず用意する（半角であることに注意！）
- `lights` という名前の4x3の配列を作って電光掲示板として実現
- 起動した際にファイル `status.txt` から1行ずつ取得し、そのそれぞれの値を配列に代入していく
 - 1行の値を読み込む時は、各値がカンマ区切りであることを考慮し、カンマごとに区切る場合は `split` を使う
- クリックされるたびに保存することにし、保存する際にも、4x3の配列の値をString型の配列に放り込んでいく

ヒント



- 0行目: `lines[0]`は1,0,1,1という値
 - カンマで区切ったとき、1と0と1と1という値が続いていることになるが、この0番目、1番目、2番目がそれぞれ左上、中左上、中右上、右上に対応つまり0番目は`lights[0][0]`、1番目は…
- 1行目: `lines[1]`は0,1,0,1という値
- 2行目: `lines[2]`は0,0,1,1という値

```
1,0,1,1
0,1,0,1
0,0,1,1
```



`lights[0][0]`

`lights[1][0]`

`Lights[3][0]`

`lights[0][1]`

`lights[1][1]`

`lights[3][1]`

`lights[0][2]`

`lights[1][2]`

`lights[3][2]`

カンマ区切りの取得



```
String[] 文字列配列 = split(文字列, '区切り文字');
```

```
// linesという配列に1行ずつ文字列を読み込む
String[] lines = loadStrings( "status.txt" );

// lines.lengthですべての行数を取得できる
for(int y=0; y<lines.length; y++ ){
    // line[i] ( i行目 ) の文字列をカンマで分割
    String[] data = split( lines[y], ',' );
    // 0番目、1番目、2番目の値を整数に変換して代入
    lights[ ? ][ ? ] = int( data[0] );
    lights[ ? ][ ? ] = int( data[1] );
    lights[ ? ][ ? ] = int( data[2] );
    lights[ ? ][ ? ] = int( data[3] );
}
```

カンマ区切りで保存！



str()で文字列に変換してカンマでつなぐ

```
// 3行分なので3つの要素からなる文字列配列を作る
String[] saveLines = new String[3]
// 1行ずつ値をカンマ区切りで代入していく
for(int y=0; y<lights.length; y++){
    // str()で文字列に変換して、「,」で4つの値をつなぐ
    // 4つだけなので直指定してますが、forで繰り返してもOK
    saveLines[ ? ] = str( ??? ) + ","
                    + str( ??? ) + ","
                    + str( ??? ) + ","
                    + str( ??? );
}
// saveStringsでsaveLinesを1行ずつstatus.txtに保存します
saveStrings("status.txt", saveLines);
```

課題3 Fitts' Law



スケッチ名: basic_fitts_law

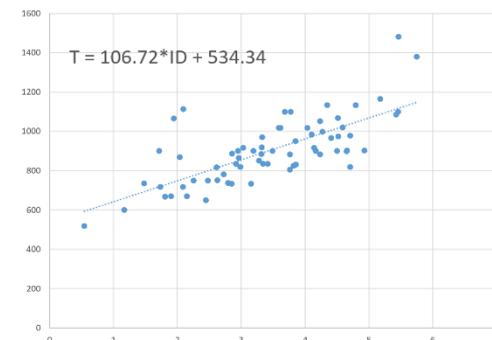
ある開始位置からターゲットの中央までの距離をD、ターゲットの大きさをW、ターゲットをクリックするまでの時間をTとしたとき、Tは下記の式で表現できる

$$T = a + b \cdot \log_2\left(\frac{D}{W} + 1\right)$$

これを確認するため、画面をクリック後に、画面にランダムに円を表示し、そのクリックまでの時間をT、開始位置（前回のクリック位置）からターゲットの中央までの距離をD、ターゲットの直径をWとしてひたすらファイルに記録するプログラムを作りたい

results.csvに $(\log_2(\frac{D}{W} + 1), T) = (ID, T)$ の値を保存していき、Excelで散布図を表示し、線形近似の直線をグラフに書き入れ確認せよ（本来これは1次元用なので、少しフィット率は下がりますがそれっぽくなります）
研究室の誰かにやってもらいましょう！

	A	B
1	2.797643	737
2	2.730543	780
3	4.707959	817
4	5.468897	1481
5	3.860189	830
6	5.458824	1099
7	3.765372	804
8	5.750654	1378
9	2.962717	864
10	3.318669	884



底が2はどう求める？



- Processingのlogは自然対数（底がe）

$$\log_2 x = \frac{\log_e x}{\log_e 2}$$

- なので、普通に $\log(x)/\log(2)$ とするだけでOK！

ファイルに追記



```
import java.io.FileWriter;

void file_println(String filename, String text){
    // ¥n はMacでは\nになります
    file_print(filename, text + "¥n");
}

void file_print(String filename, String text){
    // プログラムがあるところにファイルを作る
    filename = sketchPath("") + filename;
    // try-catch構文といってtryに失敗したらcatchに行く
    try { // まずはこの下の行を実行していく
        FileWriter fw = new FileWriter(filename, true);
        fw.write(text);
        fw.close();
    } catch (Exception ex) { //例外
        ex.printStackTrace();
    }
}
```

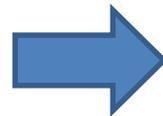
宿題 hw_questionnaire



電光掲示板のプログラムを参考にして、アンケートを作成せよ。アンケートでは縦に3題の質問を並べ、その右に各質問に対応した1から5までの5段階の回答を示すボタンを表示せよ。最初はすべて3とし、クリックで切り替えられるようにし、Saveボタンで回答がカンマ区切りで保存され、すべての回答が3に戻るようにし、次の人が回答できるようにせよ（問題は日本語で良いよ）

	1	2	3	4	5
0 : I think that I would like to use this system frequently.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
1 : I found the system unnecessarily complex.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
2 : I thought the system was easy to use.	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

save



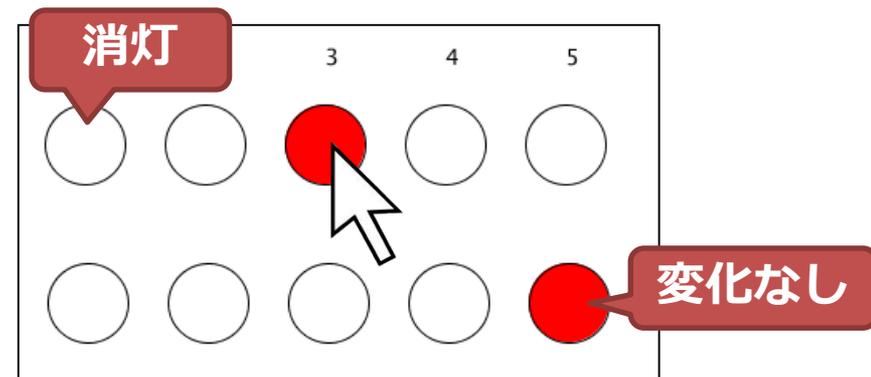
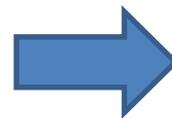
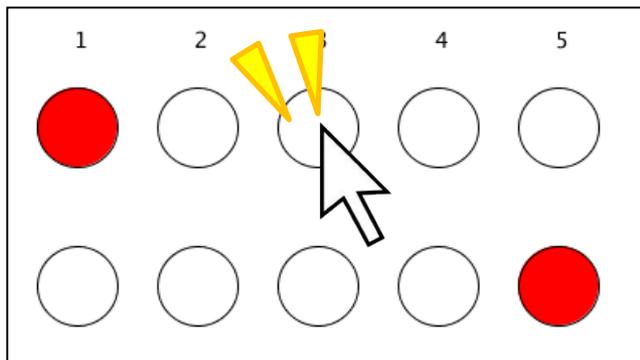
1,5,3

宿題 hw_questionnaire



ヒント

- basic_lights43と同じように配列を用意
 - 質問数 x 回答の段階数 (5で良い)
 - 配置や質問内容は任意
- 回答のボタンは常に1つだけ点灯
 - クリックしたボタン以外は "0" に
 - 別の質問の回答には影響しないように

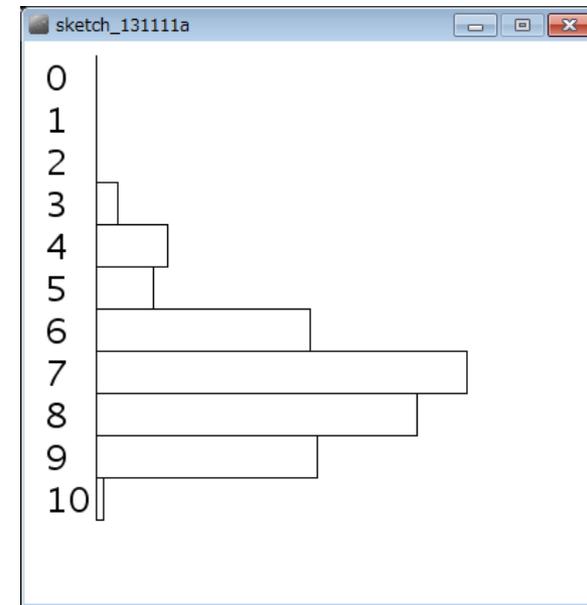


おまけ：成績グラフ作成



- <http://nkmr.io/lecture/2020/scores.txt> という180人分の成績データ（架空）を読み込んで点数毎の頻度分布を作ってみましょう

- 0-10までの数を数える配列を作成する
- ファイルから読み込んだ値に応じて、配列のそれぞれの数を1ずつ増やしていく



おまけ: ネットからロード



```
String[] lines = loadStrings("URL");
```

```
// 0から10なので11個の要素からなる配列を作る
```

```
int[] scores = new int[11];
```

```
void setup(){
```

```
    size( 400, 400 );
```

```
    textSize( 25 );
```

```
    for(int i=0; i<11; i++){
```

```
        scores[i] = 0;
```

```
    }
```

```
    String [] lines = loadStrings( "http://nkmr.io/lecture/2020/scores.txt");
```

```
    for( int i=0; i<lines.length; i++ ){
```

```
        scores[ int( lines[i] ) ]++;
```

```
    }
```

```
}
```