

インデントしていないプログラムは却下

- プログラムは読みやすく記述しましょう！
 - インデントを意識して構造をわかりやすくするCtrl+Tを押して、プログラムの階層構造(どことどこがグループなのかということ)を把握する
 - **インデントのないプログラムは採点 & QA対象外とします**
 - 思考過程をコメントにしっかり書いておく！ **コメントのないプログラムはTAさんにとって厳しいものなので、QA対象外とします！**

```
void setup(){
size(400,400);
}
void draw(){
background(255,255,255);
if( mouseX<200 ){
fill(255,0,0);}
else {
fill(0,0,0);}
ellipse(200,200,100,100);
}
```

```
void setup(){
    size(400,400);
}
void draw(){
    background(255,255,255);
    // マウスのx座標で色を変える
    if( mouseX<200 ){
        fill(255,0,0);
    } else {
        fill(0,0,0);
    }
    ellipse(200,200,100,100);
}
```

プログラミング演習I（第8回）課題

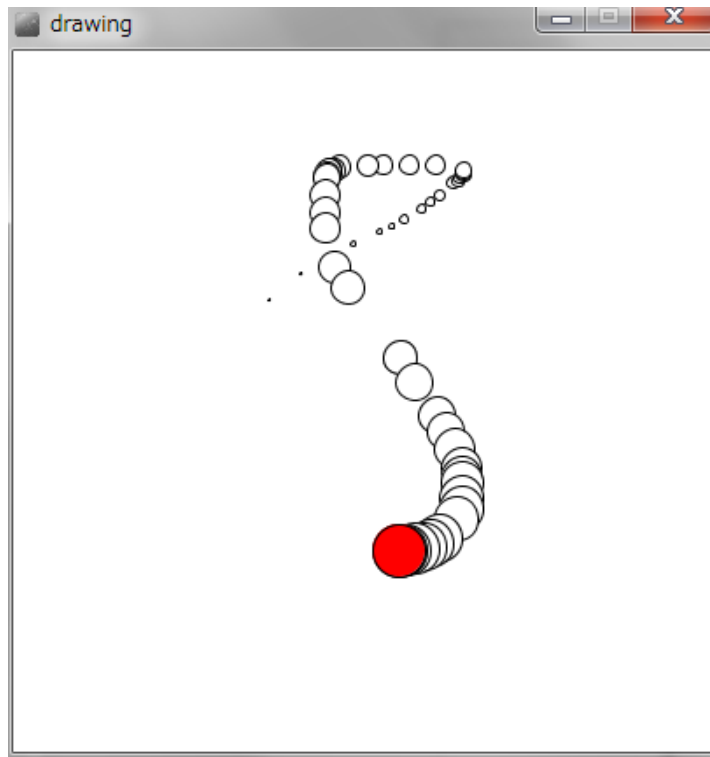
- 基本① スケッチ名 : parapara

- 何らかの画像を【5枚以上】用意し、画像をパラパラと切り替えるプログラムを作成せよ。
- 自作のキャラクタを画像化して利用してもよい。
- 画像の更新頻度 (frameRate) は適切にせよ。

プログラミング演習I（第8回）課題

• 基本② スケッチ名 : drawing

- マウスカーソルの軌跡に過去50点分、直径が30ピクセルから1ピクセルまで徐々に小さくなる白色の円を描画するプログラムを作ってください（カーソルを追尾する円）。また、新しい円を手前に表示するようにせよ
- さらに一番新しい点のみ赤色で塗りつぶすようにせよ
- Bashで値を確認しつつプログラムせよ。



[step1]

まずは予習テキストにあるマウスカーソルの軌跡を円で表示するプログラムを作って、動作を確認しよう。

[step2]

それをもとに $[i]$ 番目の点と $[i-1]$ 番目の点との間に線を描く処理を加えてみよう。

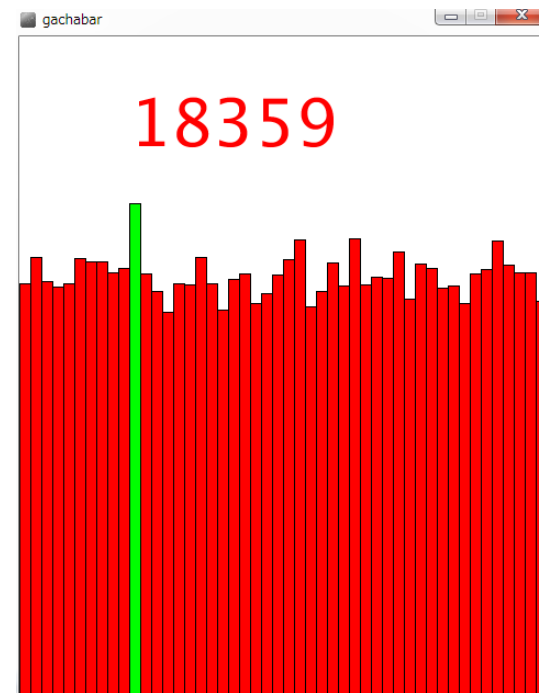
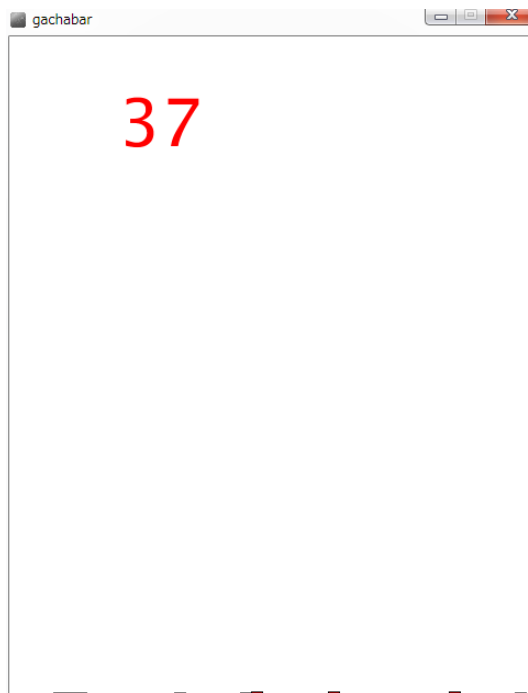
[step3]

終端位置に円を描く処理を加えてみよう。終端位置の座標値は配列の何番目の要素に入っているだろうか？

プログラミング演習I（第8回）課題

- 基本③ スケッチ名： gachachart48

- gachachartプログラムを改良し、480x600のウィンドウとし、drawが呼び出される度に48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
- また、引いたカードの数についても表示し、最大値をもつグラフを緑色にせよ



プログラミング演習I（第8回）課題

• 発展① スケッチ名 : game

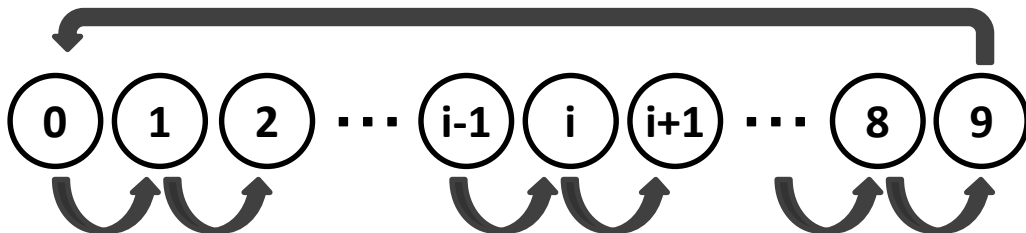
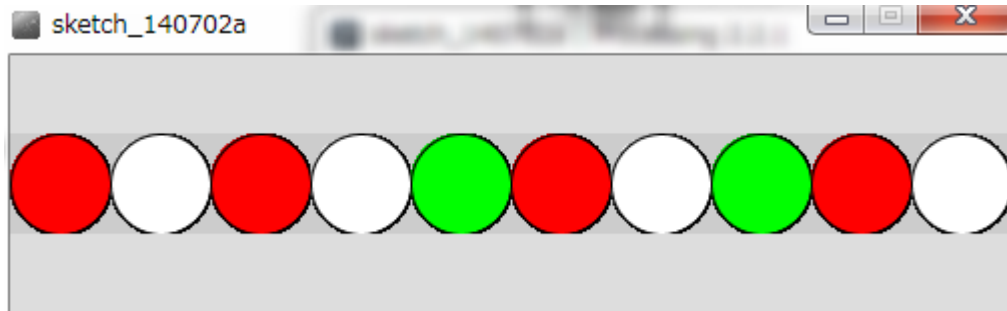
- 600x500の画面内にX座標Y座標ともにランダムに配置された30個の直径20の赤色の円を描き、それぞれのX、Y方向へのスピードで移動するようにせよ（速度は正負の両方向として、初期化時にランダムに設定する）
- また、マウスカーソルの下に直径20の緑色の円を描き、その緑色の円と赤色の円が衝突するとゲームオーバーとなり、そこまで生き残った時間をスコアとして表示せよ
- 円は端に行くと逆側から出てくるようにせよ



プログラミング演習I（第8回）課題

・ 発展② スケッチ名 : oneline

- 直径50の円を横に10個並べたボードを作成せよ。
- 円の内部をクリックする度に、その円の色が【白→赤→黄→緑→白】と変化させるようにすること。
- また、キーボードの【右】キーを押すと、色が順に左から右へ移動するようにせよ。さらに右端まで来ると左端から出てくるようにせよ。



[step1]

円を敷き詰め、各円の状態を保存する配列を作ろう。状態を示す値は、各自定義してみよう。

[step2]

円をクリックしたら、その円の値が変わり、その値に基づき色を塗ろう。

[step3]

右キーを押した時に値が左から右に移動されるようにしよう。

[step4]

右端の値を左端に代入するようにしよう。値を代入するにはどうするか？

a, b, cの最大値を求めるときはどうする？

aとbとcの関係を考える

色々条件分岐を考え
複雑になってしまう

```
if( a >= b ){  
    if( a >= c ){ println( a ); }  
    else { println( c ); }  
} else {  
    if( b >= c ){ println( b ); }  
    else { println( c ); }  
}
```

最大値をもつ変数を作る

max_valueという変数が、
最大値を常に持つように
する。つまり、max_value
より大きい値を発見すると
差し替える

```
int max_value = a;  
if( max_value < b ){  
    max_value = b;  
}  
if( max_value < c ){  
    max_value = c;  
}  
println( max_value );
```

今日のテクニック

配列の最大値や最小値を求めるときはどうする？

max_valueという変数を用意し、そのmax_valueという変数が最大値を常にもつようにする！

min_valueという変数を用意し、そのmin_valueという変数が最小値を常にもつようにする！

【最小値の例】

```
int min_value = value[0];
for( int i=0; i<value.length; i++ ){
    if( min_value > value[i] ){
        min_value = value[i];
    }
}
println( "最小値は" + min_value + "です" );
```