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(4)
条件分岐(1)

中村，高橋
小林，橋本

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

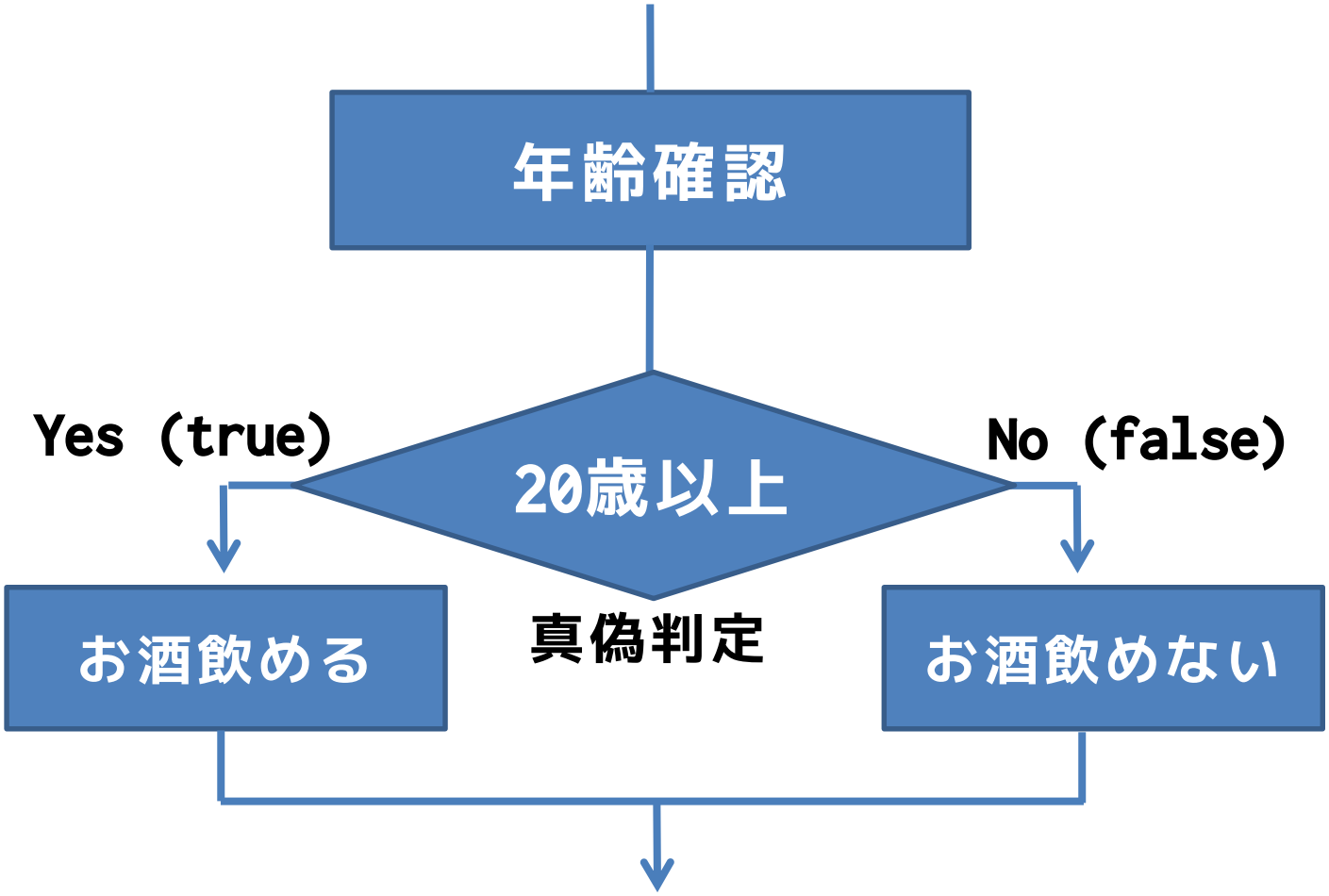
- 条件分岐を理解する
  - Processing で当たり判定
    - 何らかの条件を満たした時に色を変える！
    - マウスカーソルと動いている円がぶつかったら終了
  - 条件に応じて挙動を変える
    - 12月の次は年が1つ増えて1月になるなど
    - カレンダーを作成するため7個出力したら改行する

- 課題：
  - Processing でゲームを作ろう！
  - 占いを作ってみよう

フローチャートと条件分岐

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

- プログラムの流れ



制御文

よくあるミス
if(条件A);{ 条件A の処理 }

```
if(条件A)
{
  // 条件Aの時の処理内容
}
else if(条件B)
{
  // 条件Aでなく，条件Bの時の処理内容
}
else
{
  // 条件AおよびB以外の時の処理内容
}
```

論理値，真偽値（boolean）

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

- true か false かを値として持つ
  - それ以外の値は持たない
- 制御文で「条件Aを満たす時」というのは「条件Aが真(true)である」と同じ意味
- 制御文で「条件Aを満たさない時」というのは「条件Aが偽(false)である」と同じ意味
  - x > y の条件をみたす場合， x > y は true，みたさない場合は false
  - x == y は x と y が同じ値の場合 true に，違う値の場合に false となる

条件の記述方法

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

演算子	意味	プログラム上
x > y	x が y より大きい	左記の時にtrue それ以外でfalse
x < y	x が y より小さい	同上
x >= y	x が y 以上	同上
x <= y	x が y 以下	同上
x == y	x と y が等しい	同上
x != y	x と y が等しくない	同上
!x	x は false？	同上

条件の結果

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

- 条件の結果はtrue（真）と false（偽）になる
- さて，以下の結果はそれぞれ何になる？
  - println(5 < 3);
  - println(5 > 3);
  - println(5 == 3);
  - println(5 != 3);
  - println(5 == 5);
  - println(5 != 5);

(Q) ぞろ目判定

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

プログラムを起動した時に，1～6までのさいころを2つ振ったときの目をそれぞれ表示し，同じ目が出た時に「ぞろ目です」と表示せよ

```
1つ目は3
2つ目は3
ぞろ目です
```

- さいころを振るのは random を使う
- 1～6の目がランダムに出るようにするには下記のように書く（1以上，7より小さい）

```
int dice = (int)random(1, 7);
```

(A) ぞろ目判定

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

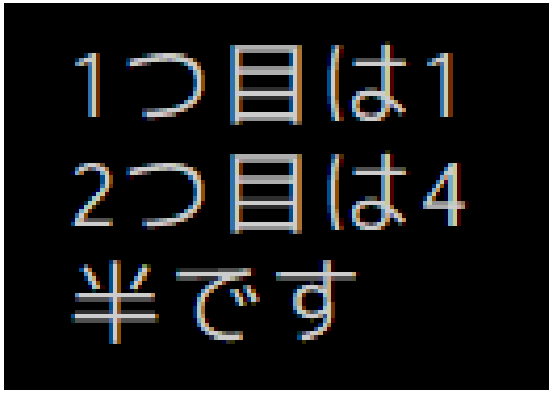

```
// 2つのサイコロを振る
int diceA = (int)random(1, 7);
int diceB = (int)random(1, 7);

// それぞれの目を出力する
println("1つ目は" + diceA);
println("2つ目は" + diceB);

// 判定する
if(diceA == diceB)
{
  println("ぞろ目です");
}
```

(Q) 丁半判定

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) プログラムを起動したときに，1～6までのさいころを2つ振ったときの目をそれぞれ表示し，その和が偶数の場合は「丁です」，奇数の場合は「半です」と出力するようにせよ

- 出力例



小テスト

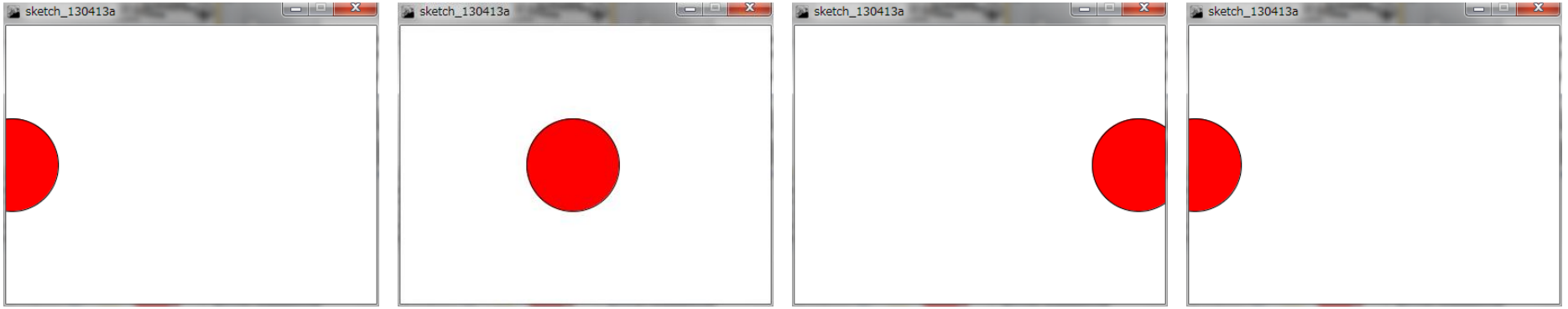
(A) 丁半判定

```
// 2つのサイコロを振る
int diceA = (int)random(1, 7);
int diceB = (int)random(1, 7);

// 出力する
println("1つ目は" + diceA);
println("2つ目は" + diceB);

if((diceA + diceB) % 2 == 0)
{
  println("丁です");
}
else
{
  println("半です");
}
```

アニメーションで戻る

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) 400x300のウインドウで，左から右へ移動する円が右端に来たら，左端に戻るように



アニメーションで戻る

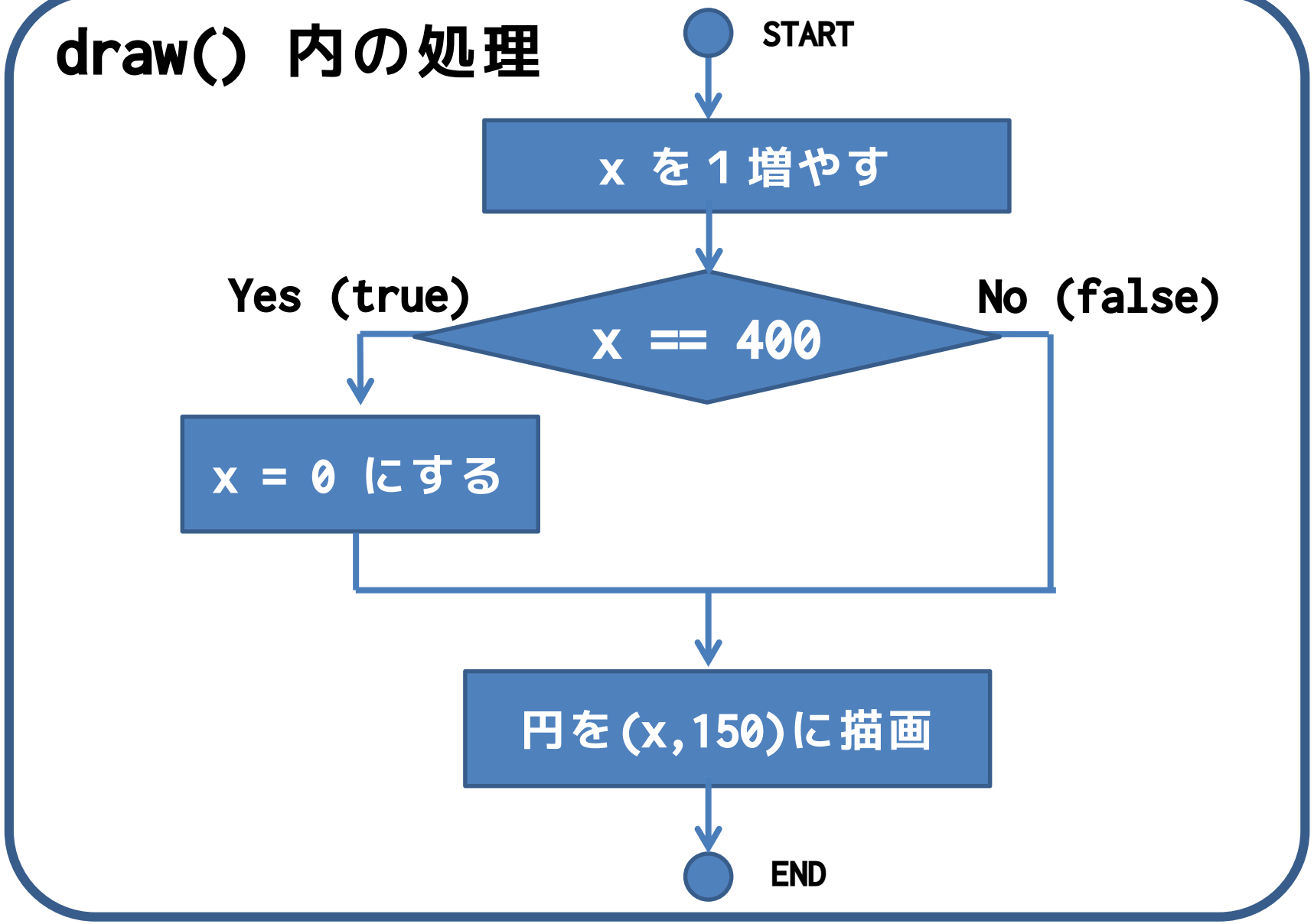
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(A) 400x300のウインドウで，左から右へ移動する円が右端に来たら，左端に戻るように

- 円の中央を変数xを使って(x,150)とする
- xは最初0にセットし，drawする度に1増やす
- 画面の端に円が来た時のxの値は400
- xが400(またはwidth)になったらxを0に!!

フローチャートと条件分岐

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



アニメーションで戻る

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

```
int x = 0;

void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
  fill(255, 0, 0);
  x = x + 1;
  if(x == 400)
  {
    x = 0;
  }
  ellipse(x, 150, 100, 100);
}
```

x == 400 の判定
x が 400 の時に true
となり括弧内が実行される

アニメーションで戻る

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

```
int x = 0;

void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
  fill(255, 0, 0);
  x = x + 1;
  if(x == width)
  {
    x = 0;
  }
  ellipse(x, 150, 100, 100);
}
```

x == width で判定しても同じこと！

アニメーションで戻る

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

```
int x = 0;

void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
  fill(255, 0, 0);
  x = x + 1;
  println(x);
  if(x == 400)
  {
    println("return");
    x = 0;
  }
  ellipse(x, 150, 100, 100);
}
```

値の変化を確認
してみましょう

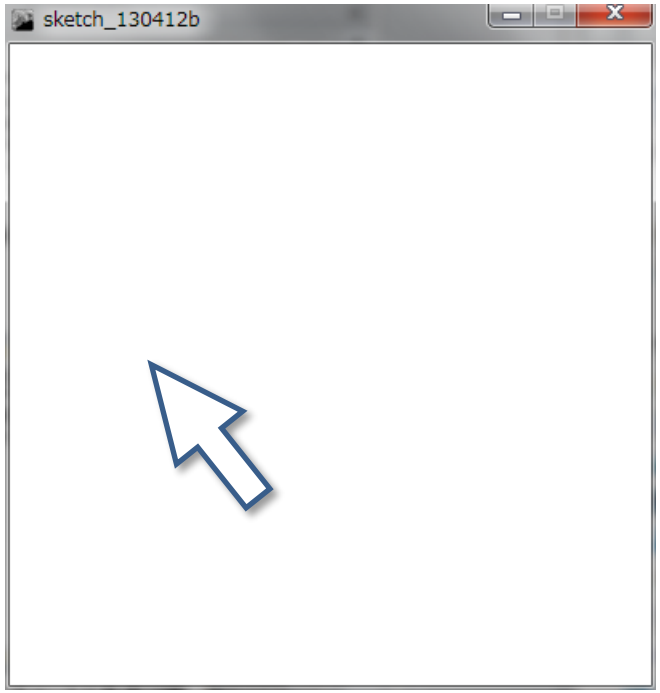
x の値の変化を表示

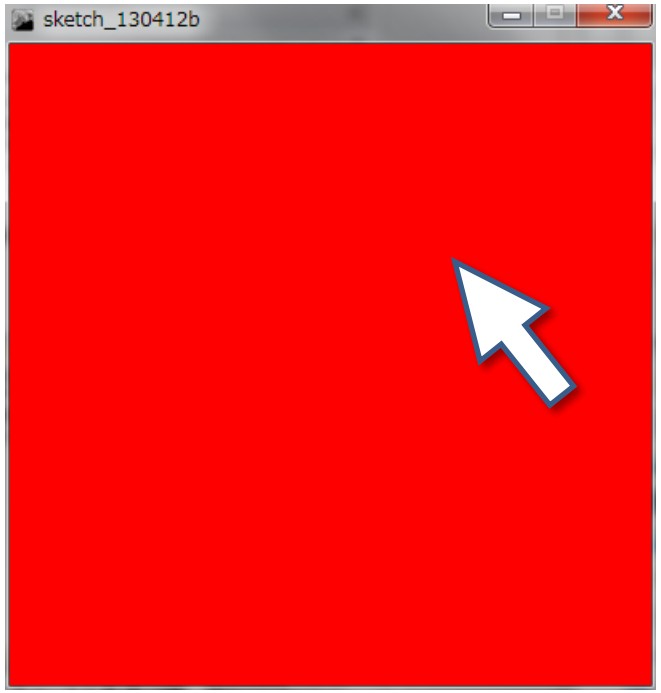
returnと表示

場所で色を変える

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) マウスが400x400のウインドウの左半分にあれば白背景，右半分であれば赤背景にする



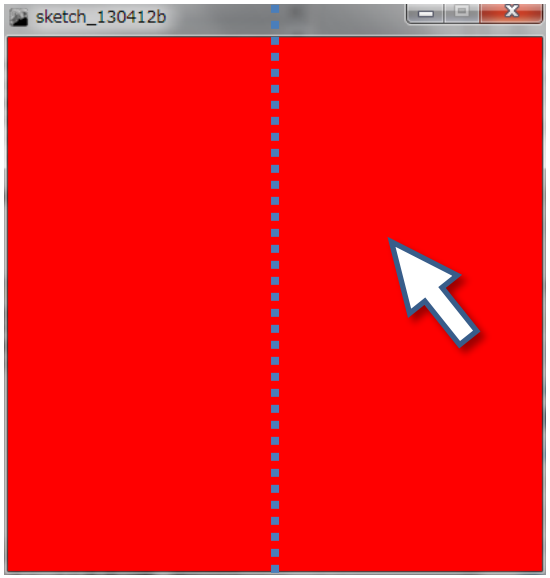


場所で色を変える

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

(A) マウスが400x400のウインドウの左半分にあれば白背景，右半分であれば赤背景にする

- マウスのX座標を利用（mouseX）
- 右半分はX座標が200より大きいという条件
- mouseX > 200 なら赤に，違えば白にしたい




```
void draw()
{
  if(                          )
  {
    background(255, 0, 0);
  }
  else
  {
    background(255, 255, 255);
  }
}
```

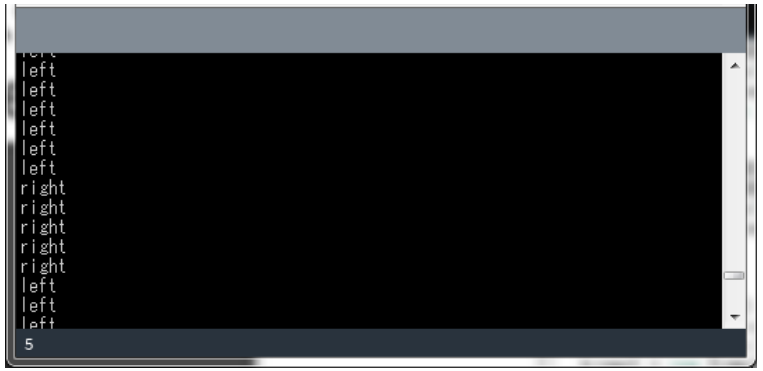
printlnで確認する！

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

(A) マウスが400x400のウインドウの左半分にあれば白背景，右半分であれば赤背景にする

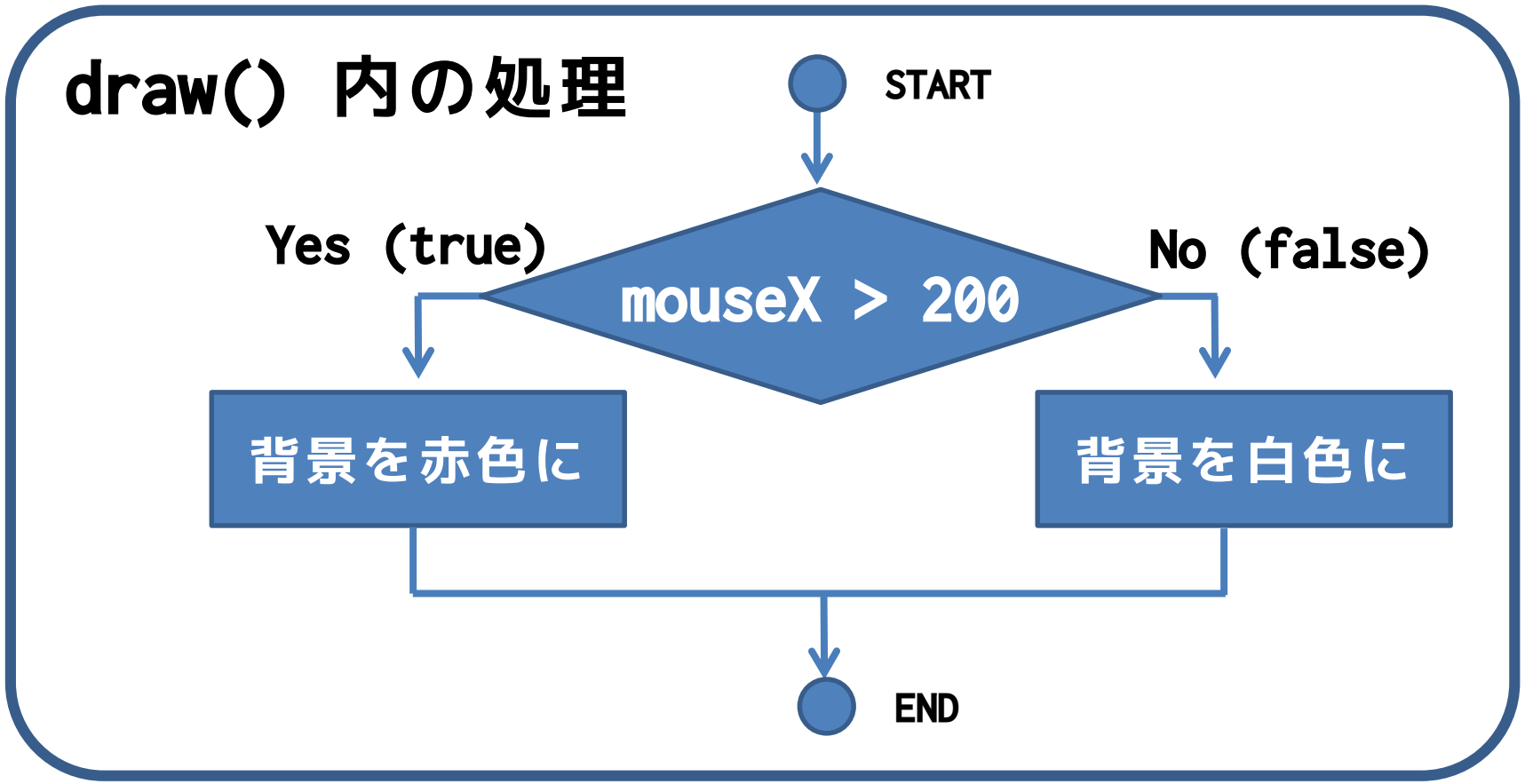
- 条件の中に入っているのかどうかを判断するため，println で left や right と表示

```
void draw()
{
  if(                          )
  {
    background(255, 0, 0);
    println("right");
  }
  else
  {
    background(255, 255, 255);
    println("left");
  }
}
```



フローチャートと条件分岐

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

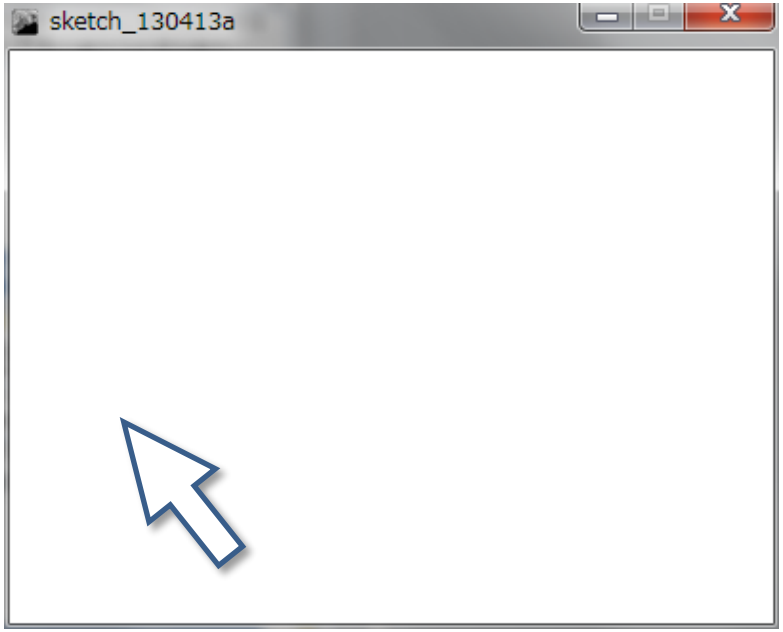


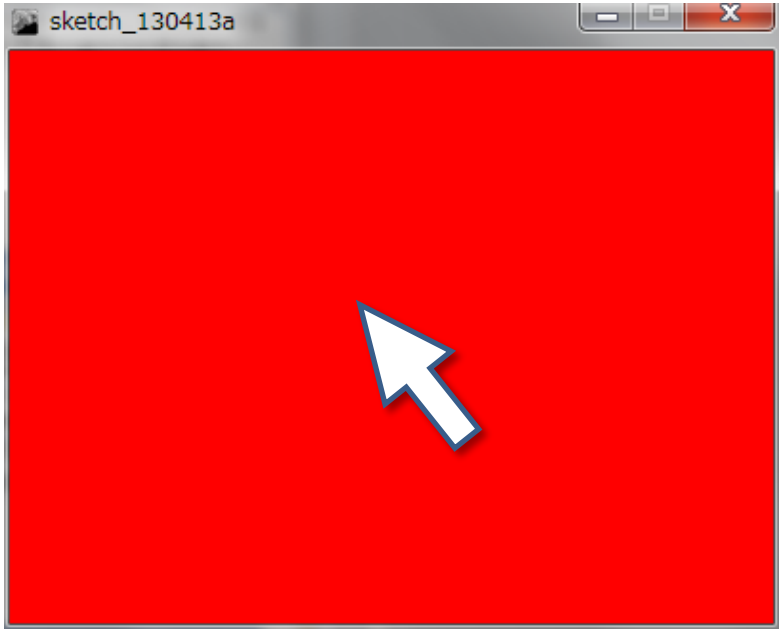
- プログラムの流れ

距離で色を変える

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) 400x300のウインドウの中心から20ピクセルの距離に入ったら赤色に塗りつぶすには？





どんな条件分岐が必要か？

距離で色を変える

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

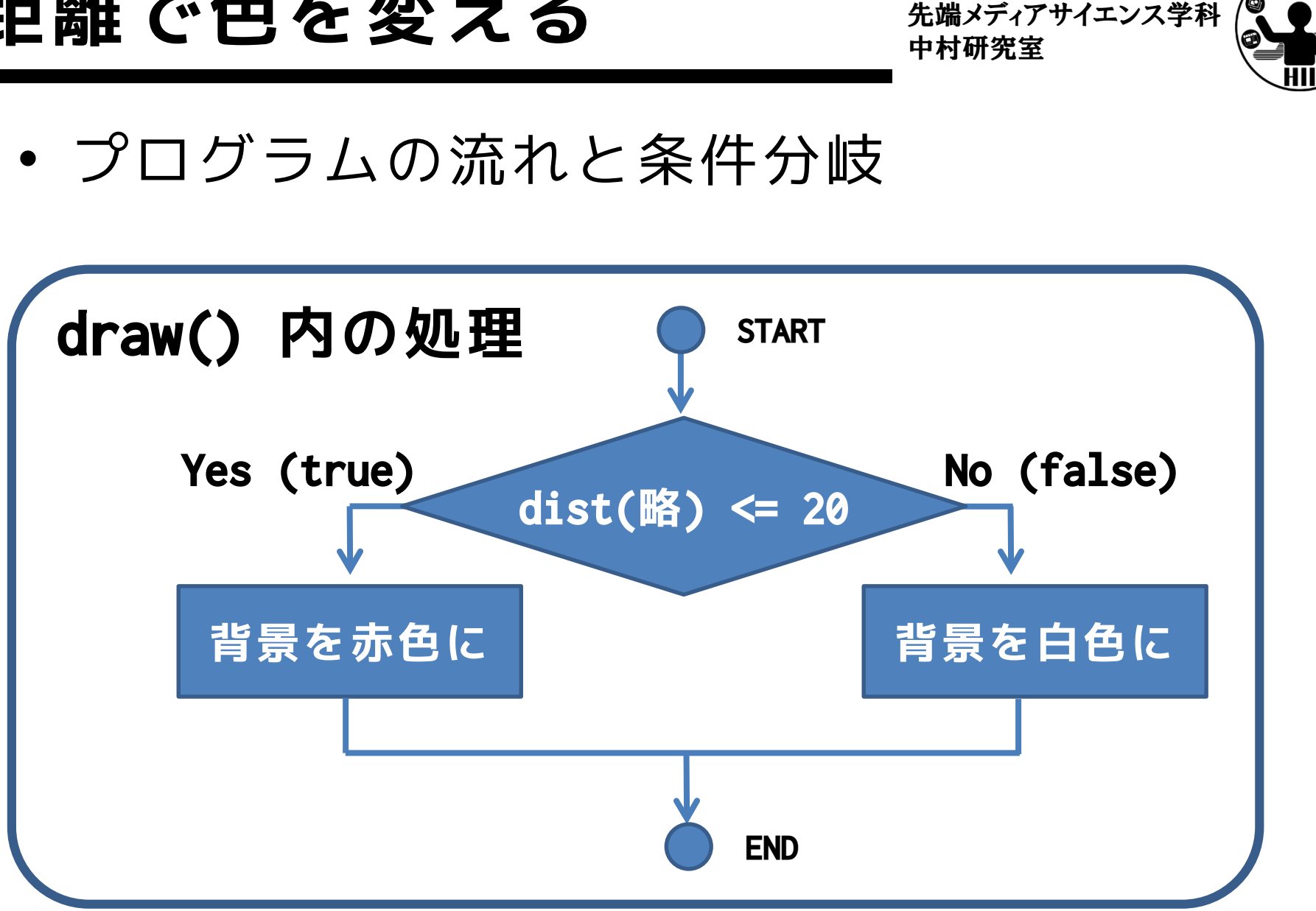
- 距離の計算は dist(x_1,y_1,x_2,y_2);
- 中心(200,150) マウス位置(mouseX,mouseY)
- 中心からマウスまで dist(200,150,mouseX,mouseY);
- dist(200,150,mouseX,mouseY)<=20 なら赤色！

```
void draw()
{
  if(dist(200, 150, mouseX, mouseY) <= 20)
  {
    background(255, 0, 0);
  }
  else
  {
    background(255, 255, 255);
  }
}
```

setup は省略

距離で色を変える

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

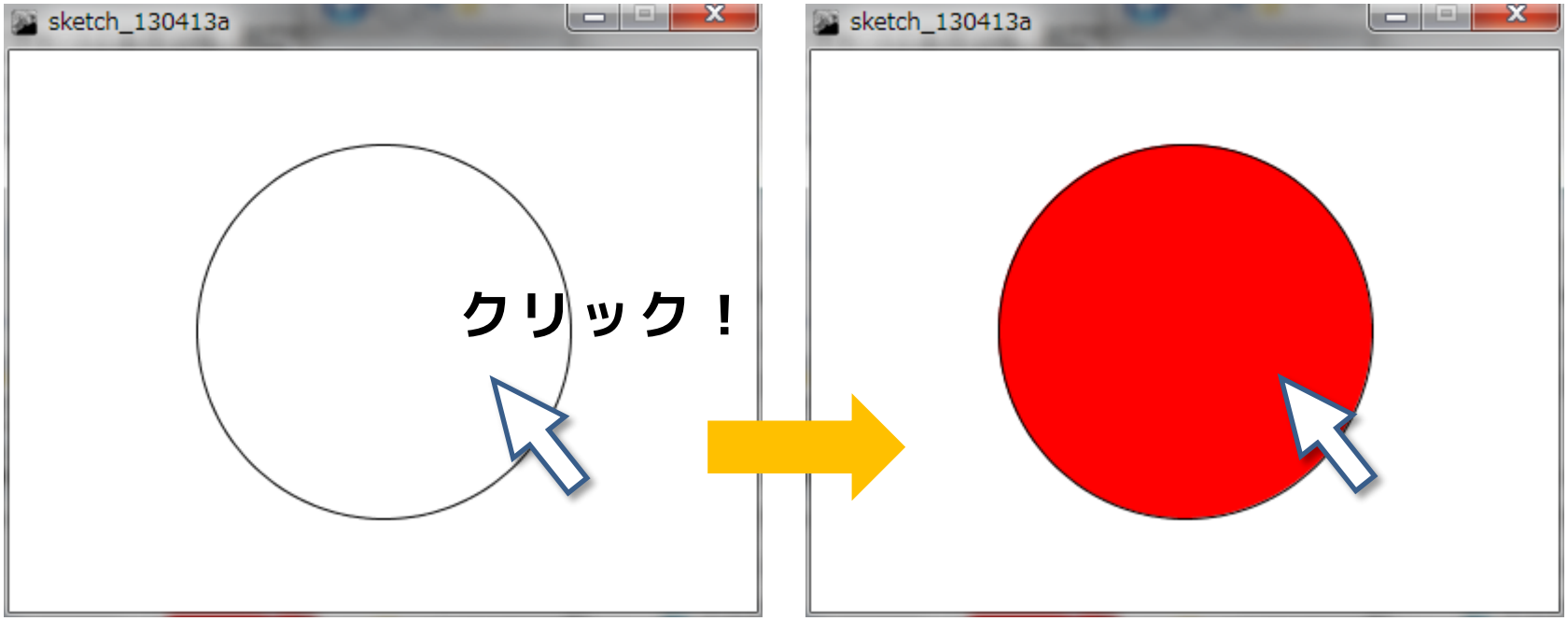


- プログラムの流れと条件分岐

クリックした場所の色変更

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) 400x300のウインドウで半径100の円を描き，円の中でクリックされたら円を赤色，外なら円を白色にするには？



クリックの取得

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

- マウスのクリックは void mousePressed(){ ... }
- クリックされた場所は mousePressed の{}内のmouseXとmouseYで取得可能
- （例） ボタンが押されたところに円を描く

```
void setup()
{
  size(400, 300);
  background(255, 255, 255);
}

void draw()
{
}

void mousePressed()
{
  ellipse(mouseX, mouseY, 10, 10);
}
```

クリックの取得

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

- 下記のようにしてしまうとクリックされた場所に丸が表示されない．なぜ？

```
void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
}

void mousePressed()
{
  ellipse(mouseX, mouseY, 10, 10);
}
```

（答）background で全部を再描画してしまうため！

クリックした場所の色変更

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

- マウスのクリックは `void mousePressed(){ ... }`
- クリックされた場所が円の中心(200,150)から，100の距離以内であれば塗り色を赤色に！
  - 100の距離以内だったらフラグを1にする（フラグを立てる）

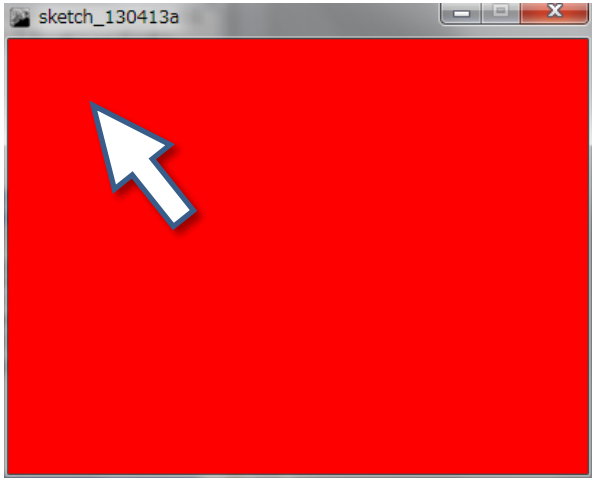
```
int flag = 0;
int r = 100;
int centerX = 200, centerY = 150;

// setupで400x300のウインドウを作る（略）

void mousePressed()
{
  if(dist(centerX, centerY, mouseX, mouseY) < r)
  {
    flag = 1;
  }
  else
  {
    flag = 0;
  }
}
```

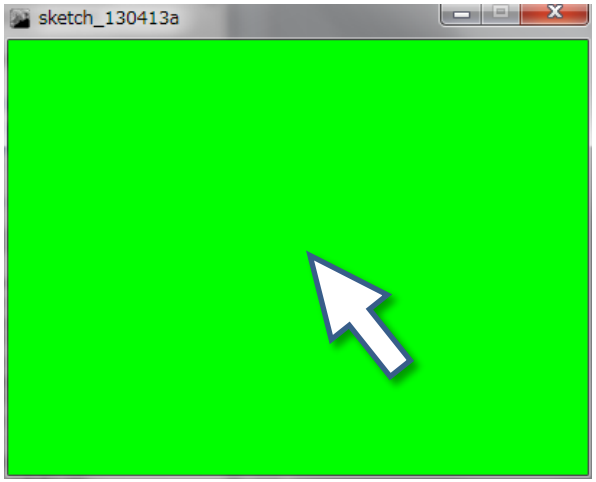
```
void draw()
{
  background(255, 255, 255);
  if(flag == 1)
  {
    fill(255, 0, 0);
  }
  else
  {
    fill(255, 255, 255);
  }
  ellipse(centerX, centerY, r*2, r*2);
}
```

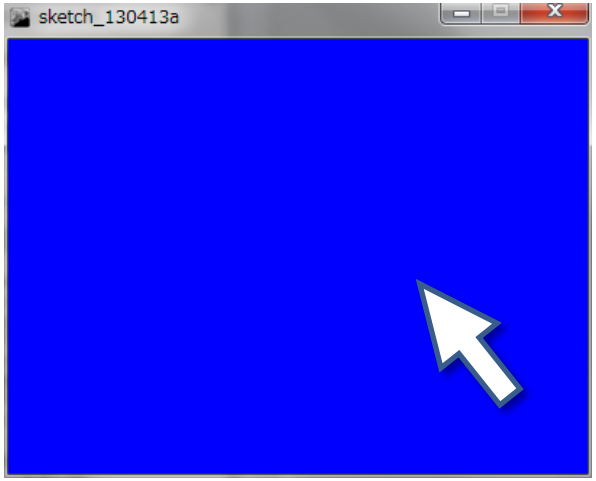
色々な条件に挑戦

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) 600x400のウインドウでマウスが画面の左で赤背景，中で緑背景，左で青背景にする



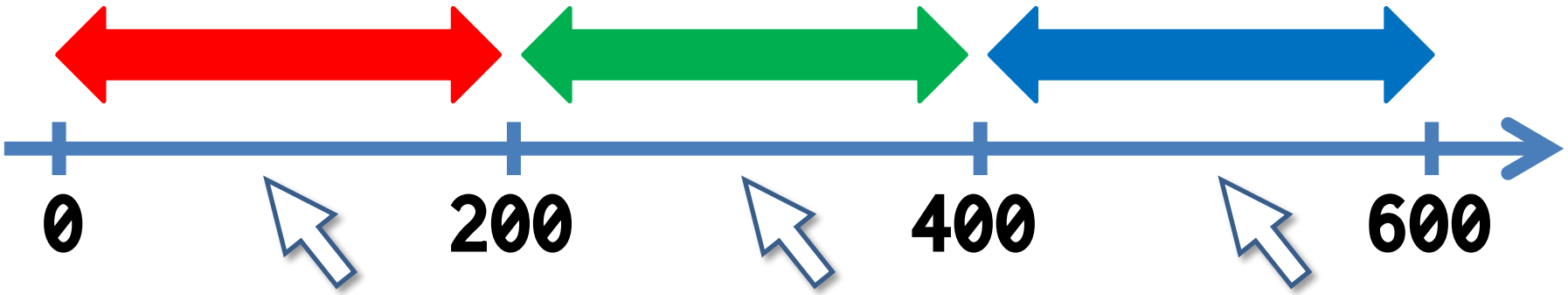




色々な条件に挑戦

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

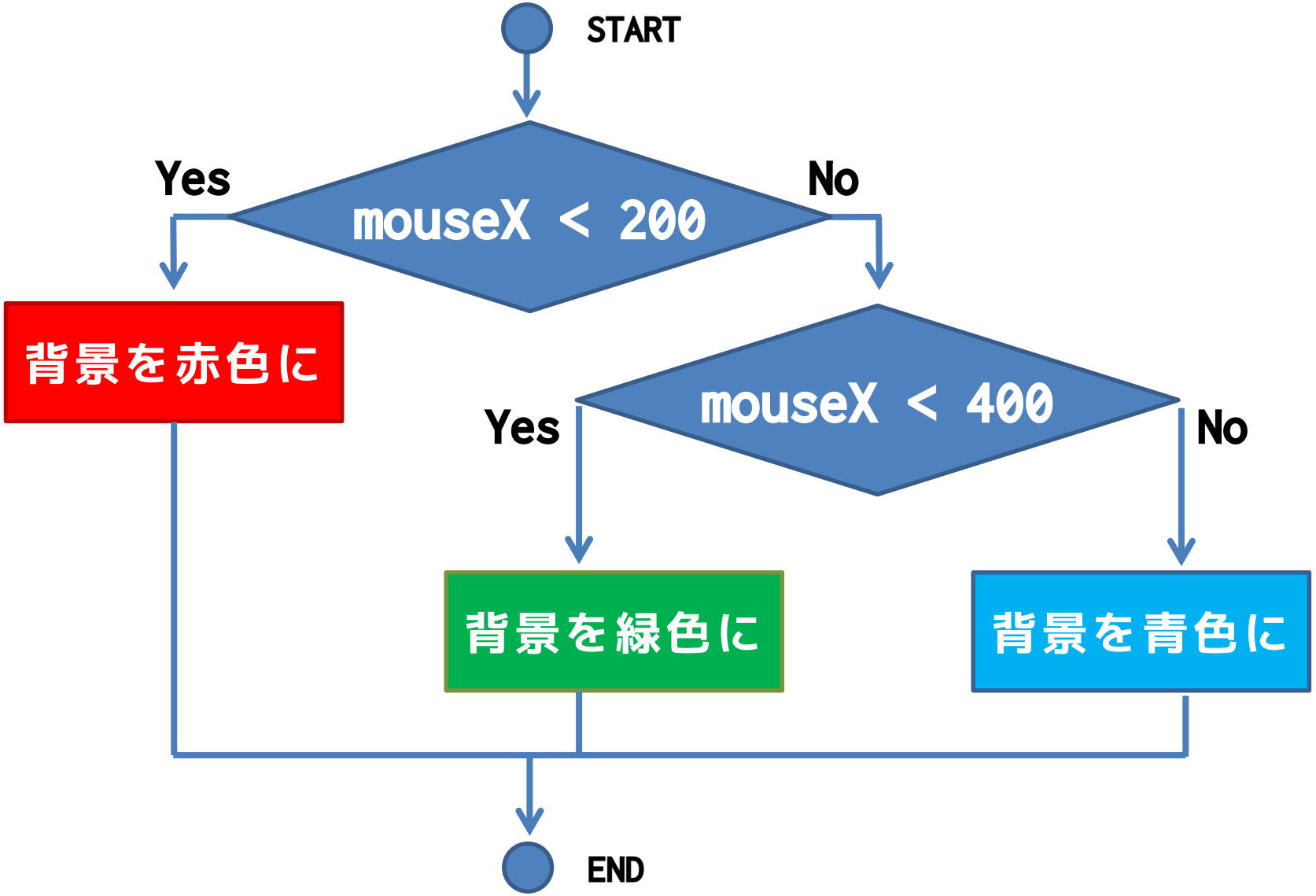
- マウスのX座標 mouseX に応じて色を変更
- 左は0～200，中は200～400，右は400～600



- もし，mouseXが0～200なら赤色，200～400なら緑色，400～600なら青色
- 言い換えると，mouseX<200なら赤色，そうでなくてmouseX<400なら緑色，そうでなければ青色

色々な条件に挑戦

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



色々な条件に挑戦

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


```
void setup()
{
  size(600, 400);
}

void draw()
{
  if(mouseX < 200)
  {
    background(255, 0, 0);
  }
  else if(mouseX < 400)
  {
    background(0, 255, 0);
  }
  else
  {
    background(0, 0, 255);
  }
}
```

色々な条件に挑戦

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

```
void setup()
{
  size(600, 400);
}

void draw()
{
  if(mouseX < width / 3)
  {
    background(255, 0, 0);
  }
  else if(mouseX < width * 2 / 3)
  {
    background(0, 255, 0);
  }
  else
  {
    background(0, 0, 255);
  }
}
```

(Q) 占い

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

プログラムを起動したときに，0〜9までの値を作成し，その値に応じて占いの結果を表示せよ

0〜3の時は「凶」

4〜6の時は「吉」

7〜9の時は「大吉」

と表示するようにせよ

0から9までの数字を出力する場合は…

```
int kuji = (int)random(0, 10);
```

(A) 占い

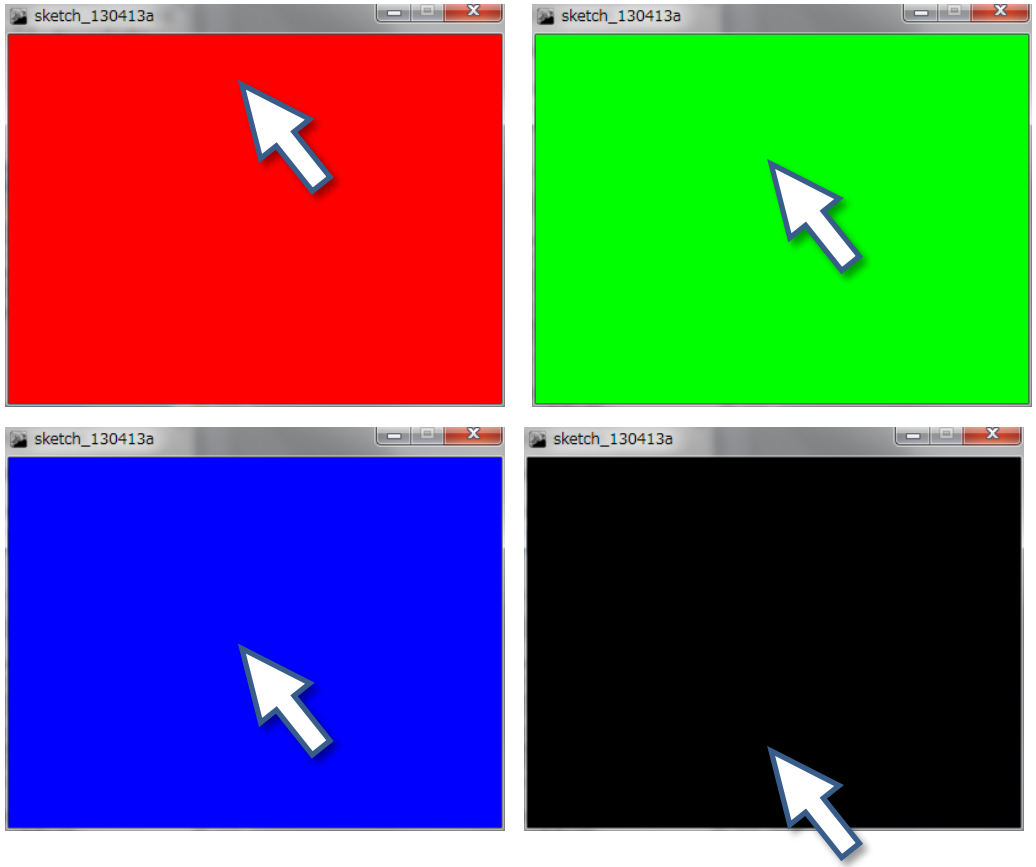
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

```
// kujiをひく
int kuji = (int)random(0, 10);

// 0-3, 4-6, 7-9
if(kuji <= 3)
{
  println("凶");
}
else if(kuji <= 6)
{
  println("吉");
}
else
{
  println("大吉");
}
```

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

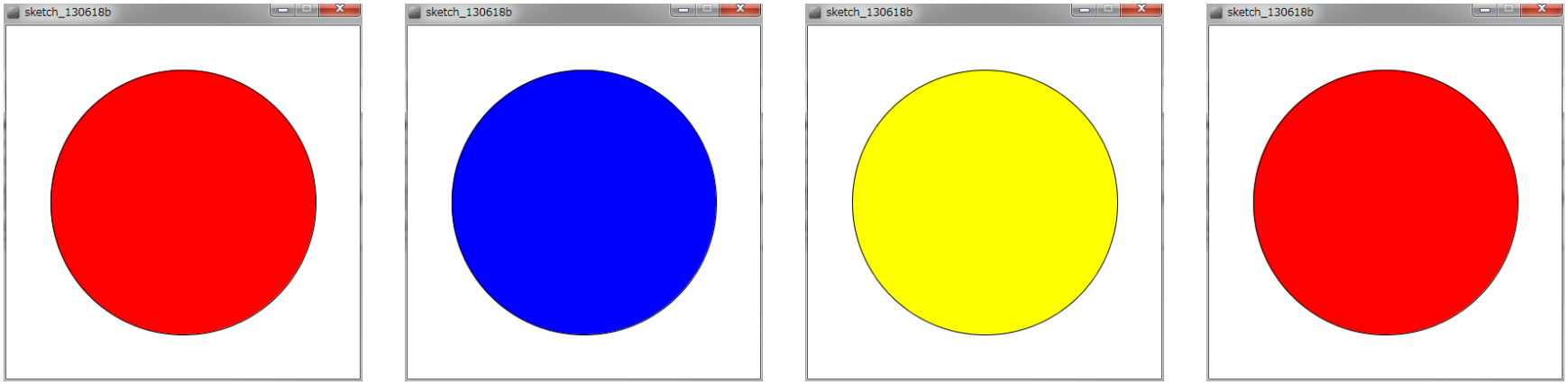
- 画面を上下に4分割し，1番目（一番上）なら赤色，2番目なら緑色，3番目なら青色，4番目（一番下）なら黒色にしてみましょう



円の色を変える

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) クリックする度に400x400のウインドウの中心に設置された円の色（直径300）が赤→青→黄→赤→青と変化するプログラムを作る



クリックする度に変化

小テスト

円の色を変える［解法1］

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
  - 400x400の中心に直径300の円描画は
    - size(400,400); と ellipse(200,200,300,300);
  - マウスがクリックされると mousePressed() に処理がやってくる
  - 赤→青→黄→赤と変化するということは，マウスのクリック回数が
    - 0回，3回，6回，9回の場合は赤色 fill(255, 0, 0);
    - 1回，4回，7回，10回の場合は青色 fill(0, 0, 255);
    - 2回，5回，8回，11回の場合は黄色 fill(255, 255, 0);
  - クリック回数を数える変数（click）を用意し，clickを3で割った余りが0なら赤，1なら青，2なら黄となる

円の色を変える［解法1］

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

```
int click = 0;

// setupで400x400のウインドウを作る

void draw()
{
  background(255, 255, 255);
  if (click % 3 == 0)
  {
    fill(255, 0, 0);
  }
  else if (click % 3 == 1)
  {
    fill(0, 0, 255);
  }
  else if (click % 3 == 2)
  {
    fill(255, 255, 0);
  }
  ellipse(width / 2, height / 2, 300, 300);
}

void mousePressed()
{
  click = click + 1;
}
```

click 回数を数える変数を用意

click を 3 で割った余りが 0 の時

click を 3 で割った余りが 1 の時

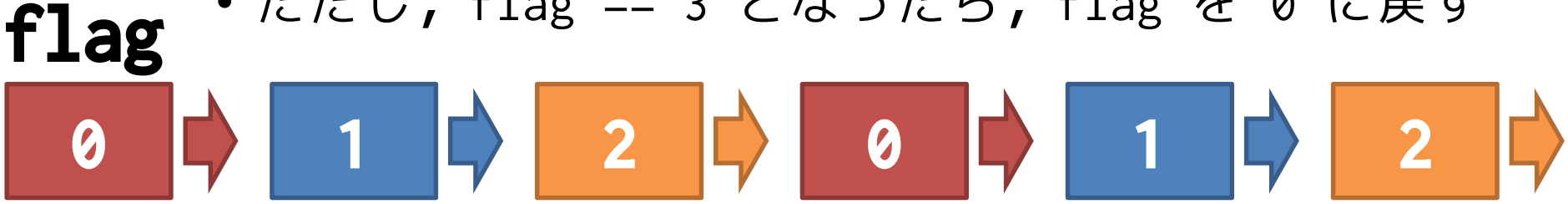
click を 3 で割った余りが 2 の時

クリックの度に click の値を増やす

円の色を変える［解法2］

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
  - 整数の変数 flag を用意し，flag によって色を変化させるようにする
    - flag == 0 の場合は赤色 fill(255, 0, 0);
    - flag == 1 の場合は青色 fill(0, 0, 255);
    - flag == 2 の場合は黄色 fill(255, 255, 0);
  - マウスがクリックされると mousePressed() に処理がやってくるのでその度に flag の値を増やす
    - ただし，flag == 3 となったら，flag を 0 に戻す



円の色

```
int flag = 0;

void draw()
{
  background(255, 255, 255);
  if (flag == 0)
  {
    fill(255, 0, 0);
  }
  else if (flag == 1)
  {
    fill(0, 0, 255);
  }
  else if (flag == 2)
  {
    fill(255, 255, 0);
  }
  ellipse(width / 2, height / 2, 300, 300);
}

void mousePressed()
{
  flag = flag + 1;
  if(flag == 3)
  {
    flag = 0;
  }
}
```

色を変更するための変数を用意

flag が 0 の時は赤色

flag が 1 の時は青色

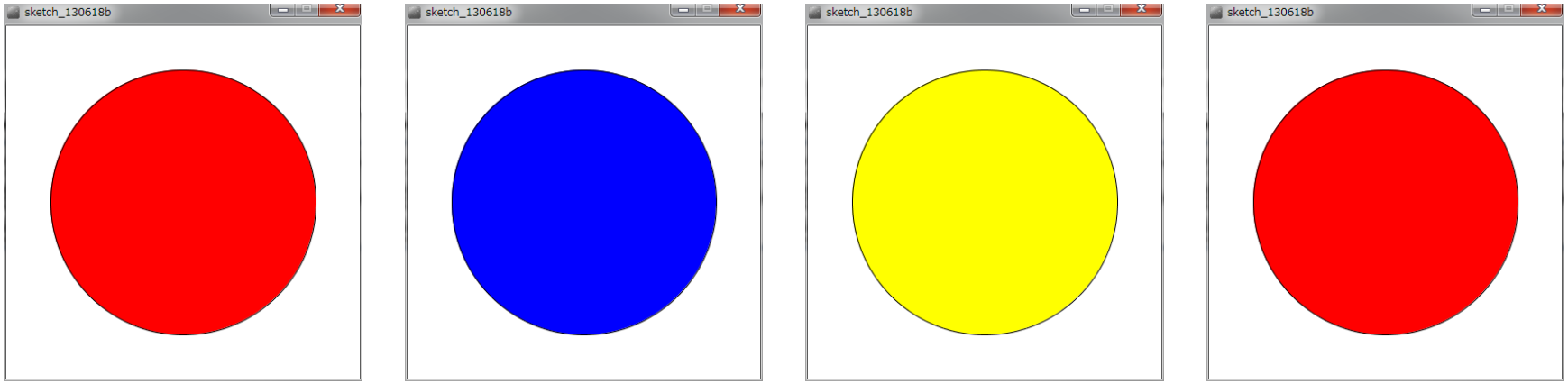
flag が 2 の時は黄色

クリックの度に flag の値を増やし
3になったら0に戻す

円の色を変える

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) 円内でクリックする度に400x400のウインドウの中心に設置された円の色（直径300）が赤→青→黄→赤→青と変化するプログラムを作る



クリックする度に変化

円の色を変える

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
  - 赤→青→黄→赤と変化するということは，マウスのクリック回数が
    - 0回，3回，6回，9回の場合は赤色 fill(255, 0, 0);
    - 1回，4回，7回，10回の場合は青色 fill(0, 0, 255);
    - 2回，5回，8回，11回の場合は黄色 fill(255, 255, 0);
  - クリック回数を数える変数（click）を用意し，clickを3で割った余りが0なら赤，1なら青，2なら黄となる
  - マウスがクリックされると mousePressed() に処理がやってくるが，mouseX，mouseY が円内にあるときだけ click の値を増やせば良い！

円の色を変える

```
int click = 0;

void draw()
{
  background(255, 255, 255);
  if (click % 3 == 0)
  {
    fill(255, 0, 0);
  }
  else if (click % 3 == 1)
  {
    fill(0, 0, 255);
  }
  else if (click % 3 == 2)
  {
    fill(255, 255, 0);
  }
  ellipse(200, 200, 300, 300);
}

void mousePressed()
{
  if(dist(200, 200, mouseX, mouseY) < 150)
  {
    click = click + 1;
  }
}
```

円の内部でクリックされた場合に click の値を増やす

円の色を変える

- width/2
- height/2
  - は中央

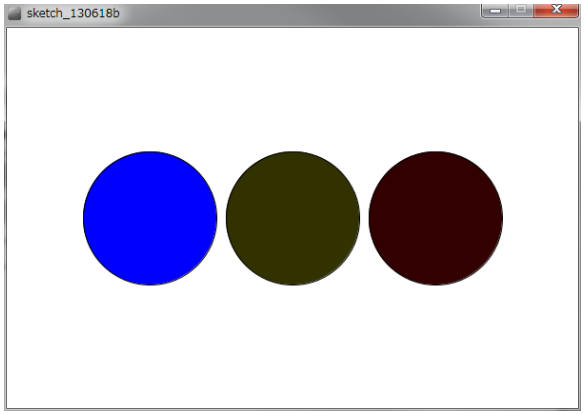
```
int click = 0;

void draw()
{
  background(255, 255, 255);
  if (click % 3 == 0)
  {
    fill(255, 0, 0);
  }
  else if (click % 3 == 1)
  {
    fill(0, 0, 255);
  }
  else if (click % 3 == 2)
  {
    fill(255, 255, 0);
  }
  ellipse(width/2, height/2, 300, 300);
}

void mousePressed()
{
  if(dist(width/2, height/2, mouseX, mouseY) < 150)
  {
    click = click + 1;
  }
}
```

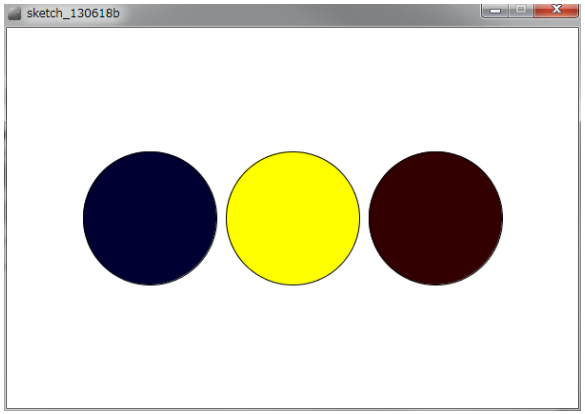
円の内部でクリックされた場合に click の値を増やす

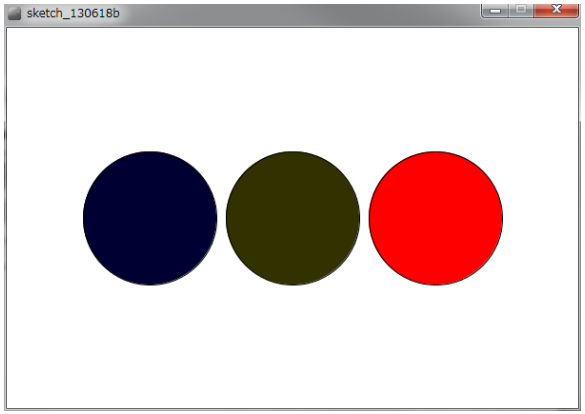
信号を作る課題

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

(Q) 信号機のプログラムを作ってください．マウスクリックによって光る丸が青→黄→赤→青→... とシフトしていくようにしてください





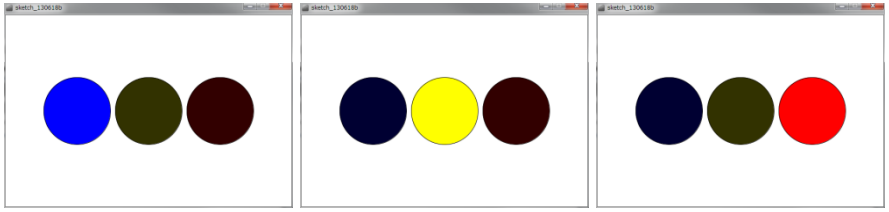


クリックする度に変化

信号を作る課題

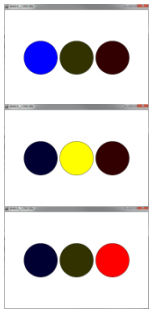
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 考え方
  - 何パターンの描画があるかを整理
    - この場合は3パターン



  - それぞれの描画パターンをどう描くかをプログラム
  - flag で描画パターンを切り替える（クリックの度に数を増やし，3になると0にする）

    - flag == 0 なら
    - flag == 1 なら
    - flag == 2 なら

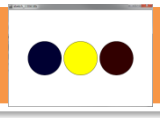


となるように draw 内で条件分岐

```
void setup()
{
  size(600, 400);
}

int flag = 0;
void mousePressed()
{
  flag = flag + 1;
  if (flag == 3)
  {
    flag = 0;
  }
}
```

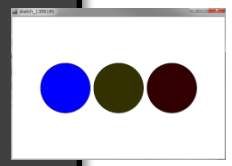
パターンを変更するための変数を用意

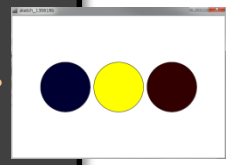
flag が 0 の時は  に

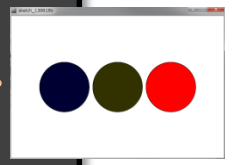
flag が 1 の時は  に

flag が 2 の時は  に

```
void draw()
{
  background(255, 255, 255);
  if (flag == 0)
  {
    fill(0, 0, 255);
    ellipse(150, 200, 140, 140);
    fill(50, 50, 0);
    ellipse(300, 200, 140, 140);
    fill(50, 0, 0);
    ellipse(450, 200, 140, 140);
  }
  else if (flag == 1)
  {
    fill(0, 0, 50);
    ellipse(150, 200, 140, 140);
    fill(255, 255, 0);
    ellipse(300, 200, 140, 140);
    fill(50, 0, 0);
    ellipse(450, 200, 140, 140);
  }
  else if (flag == 2)
  {
    fill(0, 0, 50);
    ellipse(150, 200, 140, 140);
    fill(50, 50, 0);
    ellipse(300, 200, 140, 140);
    fill(255, 0, 0);
    ellipse(450, 200, 140, 140);
  }
}
```



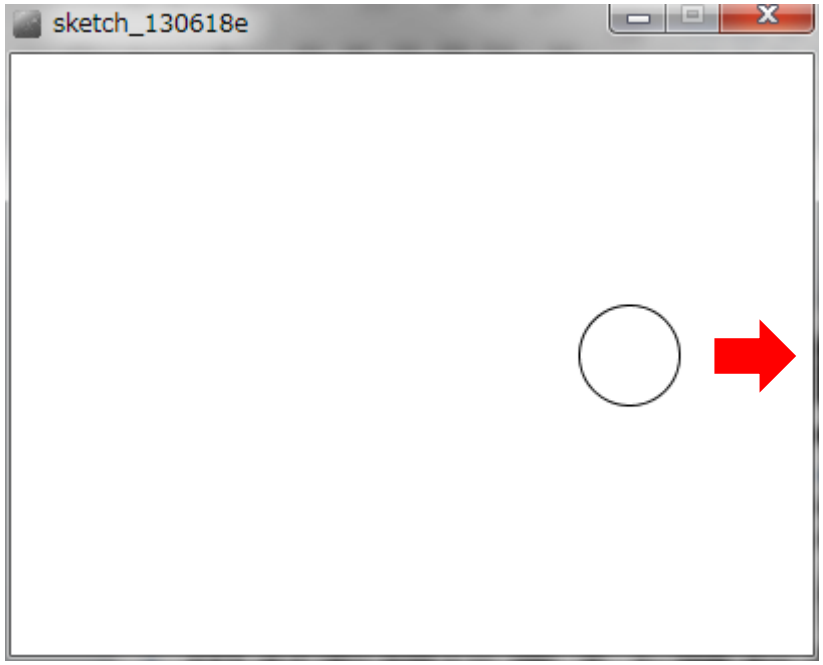


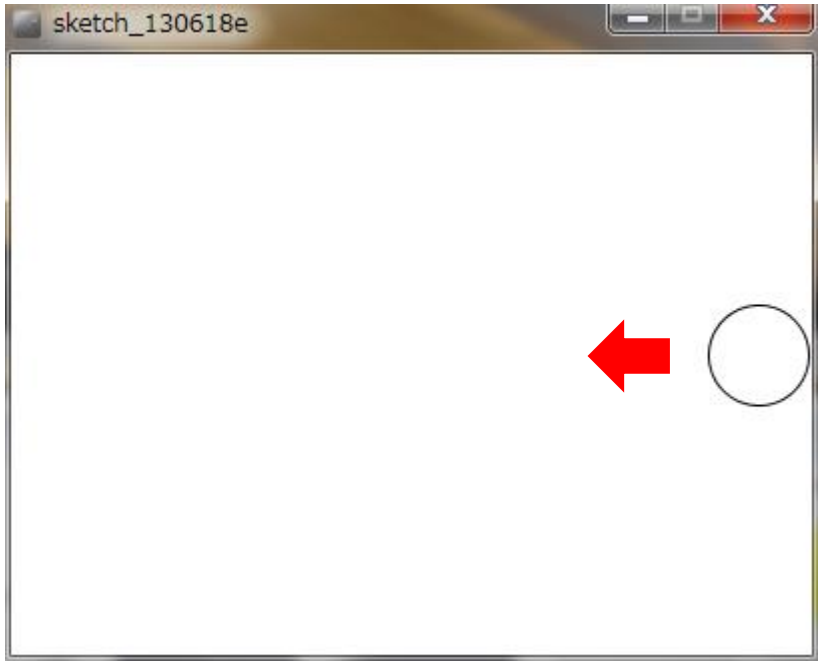


端で跳ね返る円を描く

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) 400x300のウインドウ内で，画面中央から毎フレーム2ピクセルずつ右側に移動する直径が50の円が右端に来ると跳ね返るようにする





端で跳ね返る円を描く

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

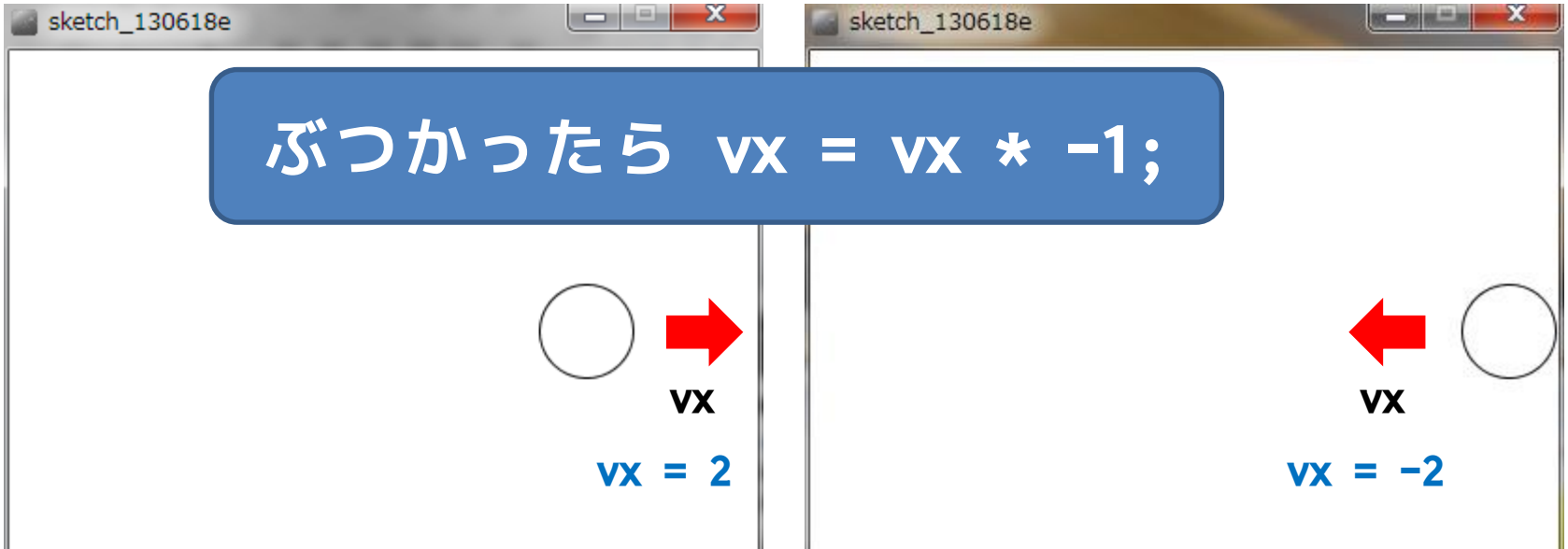
- 考え方
  - 円の座標は (x, 150) なので ellipse(x, 150, 50, 50);
  - draw() の度に x 座標を 2 ずつ増やす
  - 円がウインドウの右端に接するときの条件を考える
    - 「円の中心座標＋円の半径 >= ウインドウの右端」の場合に円がウインドウの右端と接している
      - if(x+25 >= 400){ 右端に接した時の処理をここに書く }
  - 円がウインドウの右端に接した時に速度を反転
    - 右端に到着した時に速度を -2 にしたら良い

よくある失敗．右端で吸着してしまう

端で跳ね返る円を描く

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（続き）
  - draw() の度に x 座標を 2 ずつまたは -2 ずつ変更するというのを，変数 vx だけ変更するにする！
  - 最初は vx を 2 としておき，衝突すると vx = -2 とすれば，常に x = x + vx; で座標を計算可能！
    - 衝突したら vx = -vx; とするだけでもOK!



端で跳ね返る円を描く

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


```
int x = 200;
int vx = 2;
int r = 25; // 半径をrとしよう
void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
  x = x + vx;
  if (x + r > width)
  {
    x = width - r;
    vx = -vx;
  }
  ellipse(x, height / 2, r * 2, r * 2);
}
```

両端で跳ね返る円を描く

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) 400x300のウインドウ内で，画面中央から毎フレーム2ピクセルずつ移動する直径が50の円が右端・左端に来ると跳ね返るようにする

- 考え方
  - 左端で衝突する時の条件を整理
  - 衝突した時の速度を反転させる

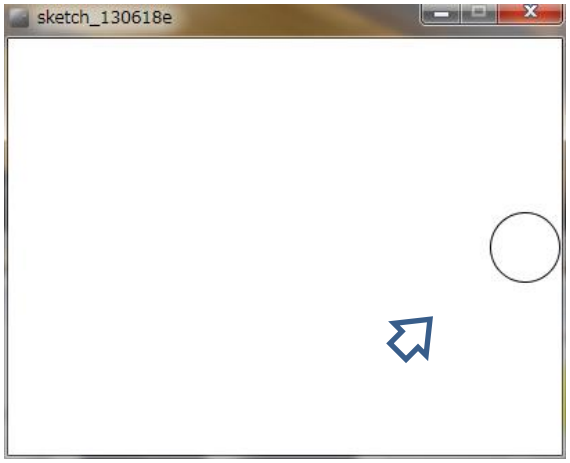
両端で跳ね返る円

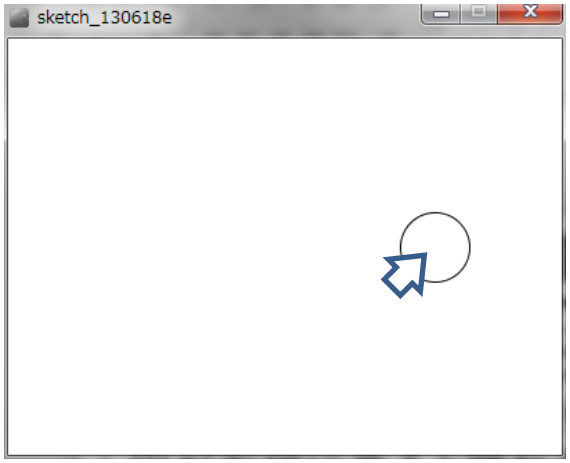
```
int x = 200;
int vx = 2;
int r = 25; // 半径をrとしよう
void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
  x = x + vx;
  if (x + r > width)
  {
    x = width - r;
    vx = -vx;
  }
  else if(x - r < 0)
  {
    x = r;
    vx = -vx;
  }
  ellipse(x, 150, r * 2, r * 2);
}
```

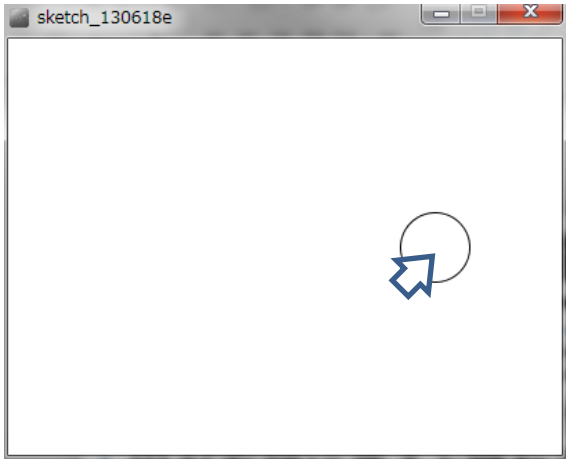
カーソルで停止する

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) 両端を行き来する円の上にマウスカーソルがある場合，円を停止させ，マウスカーソルが円から外れると動くようにするには？







カーソルで停止する

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
  - 基本的に両端への移動は先述の通り
  - カーソルが円の中にある場合には移動しない
    - ➔ カーソルが円の外にある場合に移動する
  - 円の中心と，マウスカーソルの距離を計算し，マウスカーソルが円の外にある場合に，移動のための計算を行う
    - 円の外にある場合に，円の中心の座標となる変数xを更新する！

```
int x = 200;
int vx = 2;
int r = 25;
void setup()
{
  size(400, 300);
}

void draw()
{
  background(255, 255, 255);
  if(dist(x, 150, mouseX, mouseY) > r)
  {
    x = x + vx;
  }
  if (x + r > width)
  {
    x = width - r;
    vx = -vx;
  }
  else if(x - r < 0)
  {
    x = r;
    vx = -vx;
  }
  ellipse(x, 150, r * 2, r * 2);
}
```

マウスカーソルが円の
外にある場合に移動する！
円の座標の変数を変更する！

演習

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

- 円を X 方向のみならず Y 方向にも移動させ，端に来ると跳ね返るようにしましょう
- マウスカーソルの下にある場合は停止するようにしましょう
- 衝突毎に色を変更しましょう！