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(7)
条件分岐（2）

中村、小林、辻野、鈴木

中間試験について

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

- 遅刻しないように！
- 中間試験では本日までの基本課題の、数値を変えただけのものを出します。
  - 基本課題だけできるようになっておけばOKですのでしっかり勉強しておいて！

目標

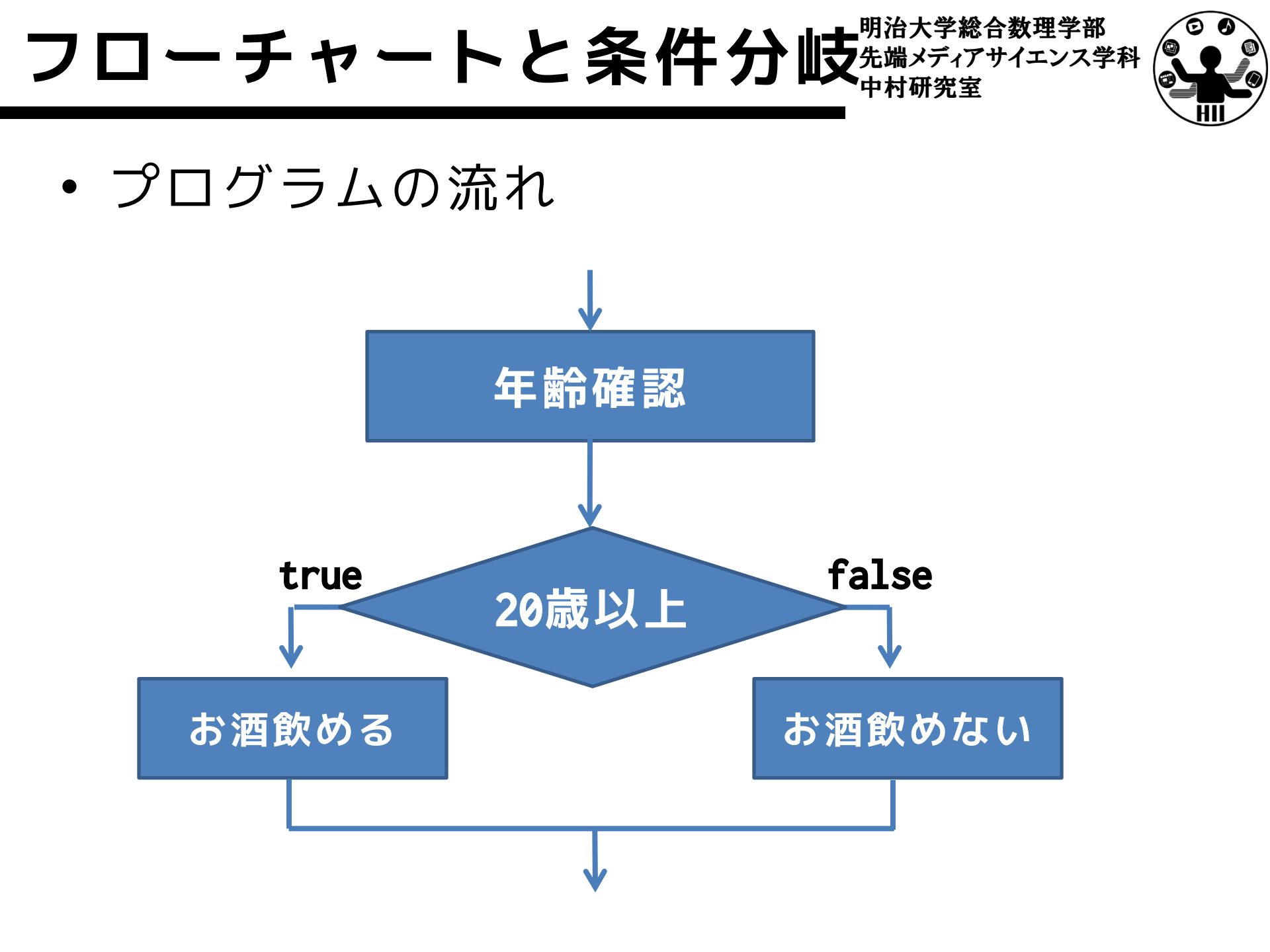
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- Processing で当たり判定に挑戦！
  - 条件分岐を理解する
  - 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
  - シューティングゲームやもぐらたたきに挑戦！

- 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);{ 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」と同意
- 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」と同意
  - 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

(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
学科

```
void setup()
{
  // 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");
  }
}
```

return

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

- return があるとそこで関数を抜け出す！

```
int hogehogeFunc(int nya, int piyo)
{
  // 色々な処理

  if( nya == ?? )
  {
    // 色々な処理
    return 2;
  }

  // 色々な処理
  return 3;
}
```

この if 文の中に入ったら
return 2 して終わり！

↑の if 文の判定で含まれ
なかったらここに来る！

(A) 占い


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

void setup()
{
  // kujiをひく
  int num = (int)random(0, 10);
  println(uranaiKekka(num));
}
```

最大値を求める

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) 乱数で発生させられた3つの数字の最大値を求めよ

```
int num1 = (int)random(100);
int num2 = (int)random(100);
int num3 = (int)random(100);
```

– 出力例

```
最大値は83です
```

最大値を求める

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

  num1 と num2 のどちらが大きいかを比較

  (a) num1 が大きい場合は、num1 と num3 を比較

  (a-1) num1 が大きい場合は num1 が最大値

  (a-2) そうでない場合は num3 が最大値

  (b) そうでない場合は、num2 と num3 を比較

  (b-1) num2 が大きい場合は num2 が最大値

  (b-2) そうでない場合は num3 が最大値

最大値

```
int num1 = (int)random(100);
int num2 = (int)random(100);
int num3 = (int)random(100);

if(num1 > num2)
{
  if(num1 > num3)
  {
    println(num1 + "が最大値");
  }
  else
  {
    println(num3 + "が最大値");
  }
}
else
{
  if(num2 > num3)
  {
    println(num2 + "が最大値");
  }
  else
  {
    println(num3 + "が最大値");
  }
}
```

最大値


```
void setup(){
  int number1 = (int)random(100);
  int number2 = (int)random(100);
  int number3 = (int)random(100);
  println("最大値は" + getMaxNumber(number1, number2, number3));
}

int getMaxNumber(int num1, int num2, int num3){
  if(num1 > num2)
  {
    if(num1 > num3)
    {
      return num1;
    }
    else
    {
      return num3;
    }
  }
  else
  {
    if(num2 > num3)
    {
      return num2;
    }
    else
    {
      return num3;
    }
  }
  return 0; // ここに来ることはない
}
```

最大値を求める

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（2）
  - どの変数が最大値なのかを求める必要はなく、どの値が最大値かを求めれば良い
  - 最大値を保持する変数 max_value を用意する
  - max_value に num1 を代入
  - max_value と num2 を比較して max_value が num2 より小さければmax_value に num2 を代入
  - max_value と num3 を比較して max_value が num3 より小さければmax_value に num3 を代入
  - max_value を最大値として出力する

最大値を求める

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

- こちらの方がプログラム的にはシンプル

```
int num1 = (int)random(100);
int num2 = (int)random(100);
int num3 = (int)random(100);

int max_value = num1;

if(max_value < num2)
{
  max_value = num2;
}
if(max_value < num3)
{
  max_value = num3;
}
println(max_value + "が最大値");
```

最大値を求める関数

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

- 関数で書いておくと使い回せるよ！

```
void setup(){
  int number1 = (int)random(100);
  int number2 = (int)random(100);
  int number3 = (int)random(100);
  println(getMaxValue(number1, number2, number3) + "が最大値");
}

int getMaxValue(int num1, int num2, int num3){
  int max_value = num1;
  if(max_value < num2)
  {
    max_value = num2;
  }
  if(max_value < num3)
  {
    max_value = num3;
  }
  return max_value;
}
```

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

- 乱数で発生させられた3つの数字の最小値を求めよ

- ヒント
  - 最大値を求めるプログラムと逆のことをやればOK！

小テスト

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

- 乱数で発生させられた4つの数字の最大値を求めよ
  - 4つの数字の最大値を求める関数とか作ってみよう！
- ヒント
  - num4 を追加して条件の判定をしていくだけ！

～かつ～はどうするのか？

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

- 論理積演算子と論理和演算子
  - ～ かつ ～ のとき　　&&
  - ～ または ～ のとき　　||

```
if ((x > 200) && (x < 400))
{
  xが200より大きく、400より小さい時
}
if ((n < -10) || (n > 10))
{
  nが-10より小さいか、10より大きいとき
}
```

~~if(200 < x < 400)~~

～かつ～はどうするのか？

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


```
if ((x > 200) && (x < 400)){
    x が200より大きく、400より小さい時はここに来る
}
```

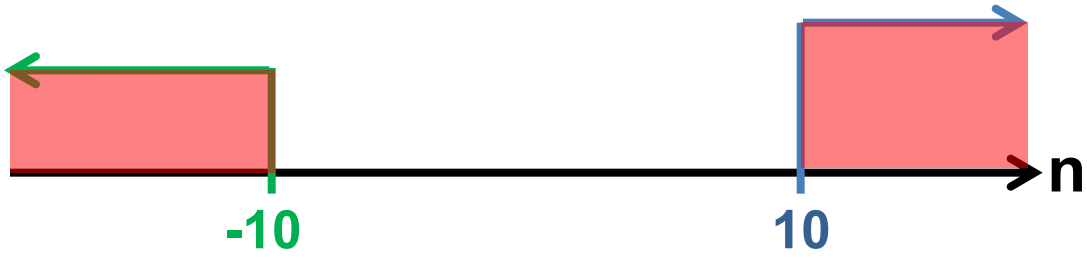
詳しく説明すると・・・

- x = 100 のときは、「x>200がfalse」で「x<400がtrue」となるため、「false && true」となり、「false」となる
- x = 300 のときは、「x>200がtrue」で「x<400がtrue」となるため、「true && true」となり、「true」となる
- x = 500 のときは、「x>200がtrue」で「x<400がfalse」となるため、「true && false」となり、「false」となる

～かつ～はどうするのか？

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


```
if ((n < -10) || (n > 10)){
   nが-10より小さいか、10より大きければここに来る
}
```



詳しく説明すると・・・

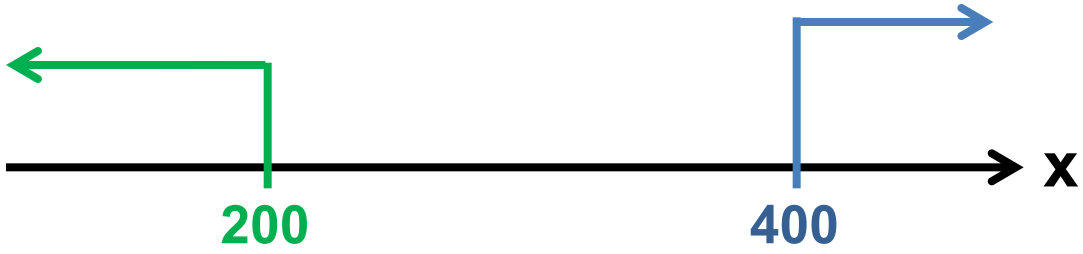
- n = -20 のときは、「n<-10がtrue」で「n>10がfalse」となるため、「true || false」となり、「true」となる
- n = 0 のときは、「n<-10がfalse」で「n>10がfalse」となるため、「false || false」となり、「false」となる
- n = 20 のときは、「n<-10がfalse」で「n>10がtrue」となるため、「false || true」となり、「true」となる

色々とつまずくポイント

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

全部 true にならない例

```
if((x < 200) && (x > 400)){
    // だめ
}
```



全部 true になってしまう例

```
if((n > -10) || (n < 10)){
    // だめ
}
```

(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

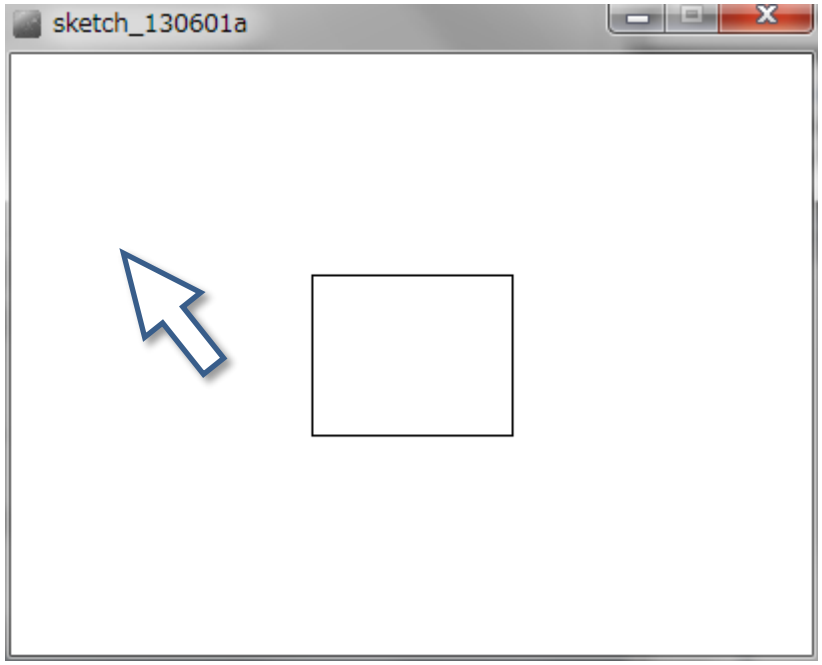
明治大学総合数理学部
先端メディアサイエンス学科

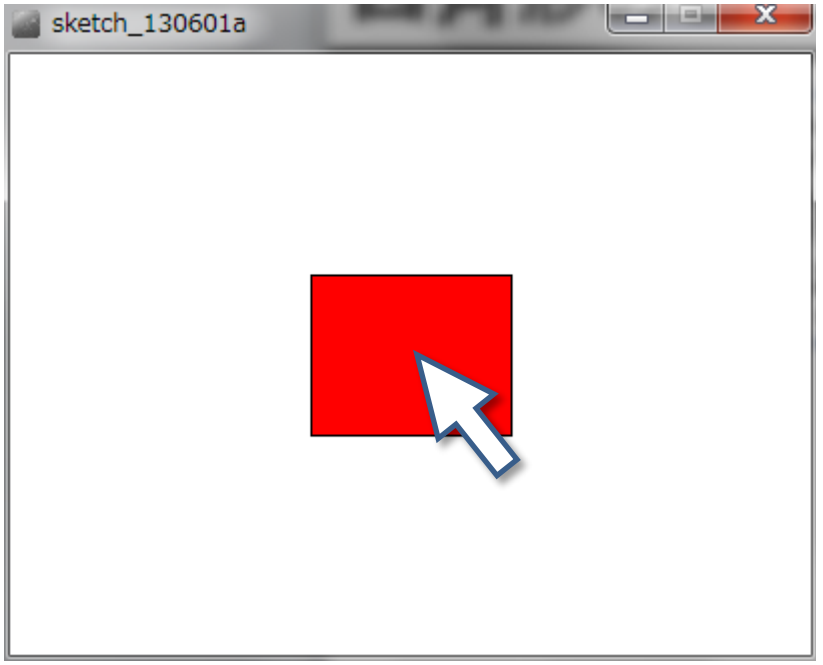
```
// kujiをひく
int kuji = (int)random(0, 10);

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

ボタン（四角形）の判定

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）400x300のウインドウの中央に表示された横100、縦80のボタンの上にカーソルがあるとボタンを赤色に、そうでなければ白色にする



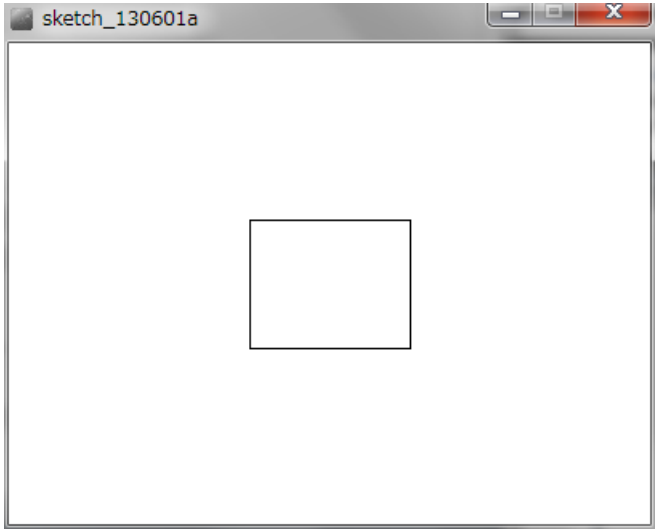


小テスト

ボタン（四角形）の判定

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

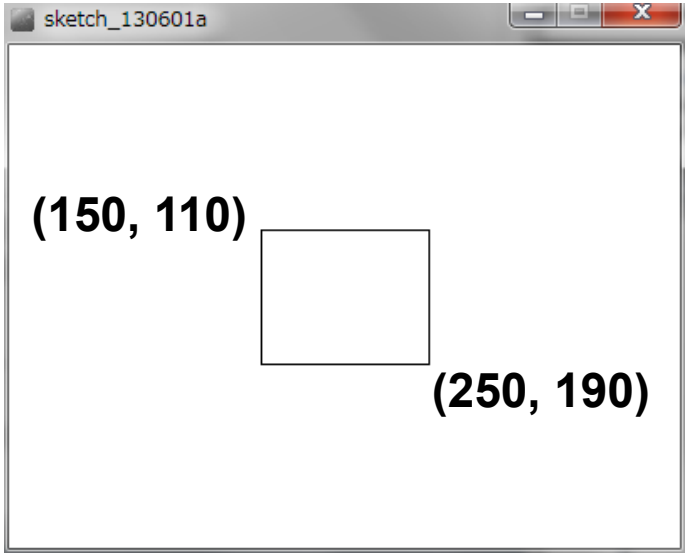
- 考え方
  - 画面の中央は（200，150）
  - ボタンの左上と右下の座標は？？
  - ボタンは rect(左上x，左上y，横幅，縦幅)；で描画
  - マウスカーソルの座標は（mouseX，mouseY）



ボタン（四角形）の判定

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

- 中央(200, 150)で、横幅100, 縦幅80なので
  - 左上の座標は(200-100/2, 150-80/2) = (150, 110)
  - 右下の座標は(200+100/2, 150+80/2) = (250, 190)
- 150 ≦ mouseX ≦250 かつ 110 ≦ mouseY ≦ 190なら赤色、そうでなければ白色で塗りつぶす
  - つまり、mouseX>=150 かつ mouseX<=250 かつ mouseY>=110 かつ mouseY<=190 の時！
- 「かつ」は、「&&」で表現する



ボタン（四角形）の判定

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

（注意） 150 <= mouseX <= 250 はダメ！

(150 <= mouseX) && (mouseX <= 250) に分解

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

void draw()
{
  background(255, 255, 255);
  if((mouseX >= 150) && (mouseX <= 250) && (mouseY >= 110) && (mouseY <= 190))
  {
    fill(255, 0, 0);
  }
  else
  {
    fill(255, 255, 255);
  }
  rect(150, 110, 100, 80);
}
```

多段階の条件分岐

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


```
void draw()
{
  background(255, 255, 255);
  if((mouseX >= 150) && (mouseX <= 250))
  {
    if((mouseY >= 110) && (mouseY <= 190))
    {
      fill(255, 0, 0);
    }
    else
    {
      fill(255, 255, 255);
    }
  }
  else
  {
    fill(255, 255, 255);
  }
  rect(150, 110, 100, 80);
}
```

さらに110≦mouseY≦190
ならここに入ってくる

150≦mouseX≦250なら
ここに入ってくる

さらに110≦mouseY≦190
でない場合はここに入ってくる

ボタン判定の関数を作る

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

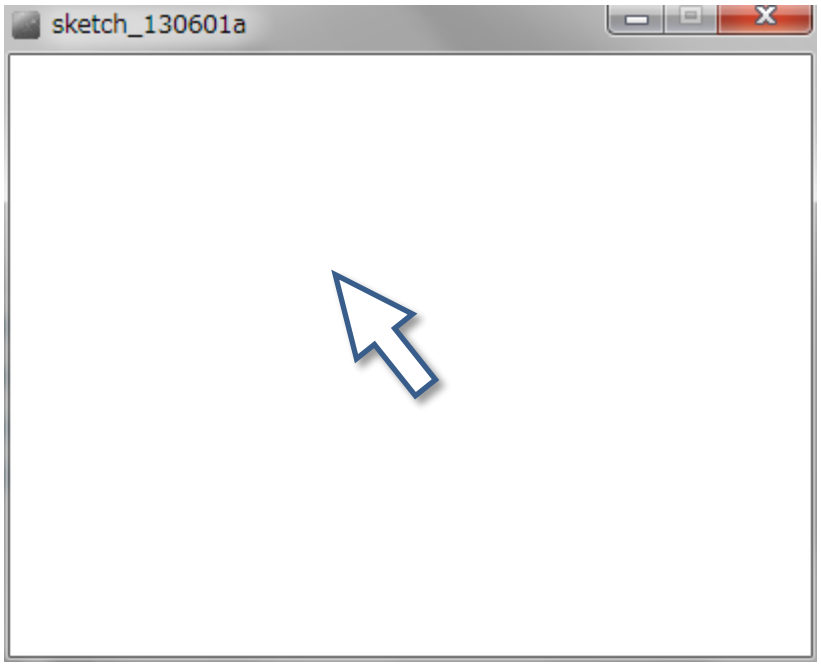
内部だったらtrue、外部ならfalseを返そう

```
boolean isInside(int px, int py, int x1, int y1, int x2, int y2){
{
  if((x1 <= px) && (px <= x2) && (y1 <= py) && (py <= y2)){
    return true;
  }
  return false;
}

void setup(){
  background(255, 255, 255);
  if(isInside(mouseX, mouseY, 150, 110, 250, 190))
  {
    fill(255, 0, 0);
  }
  else
  {
    fill(255, 255, 255);
  }
  rect(150, 110, 100, 80);
}
```

宝探し（四角形を探す）

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) ランダムな場所に、横幅30x縦幅20の大きさで配置された四角形を配置し、探せるようにせよ（カーソルがその上にあるときだけ表示する）



宝探し（四角形を探す）

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
  - 四角形の左上の座標を実数の変数に
    - `float leftTopX;  // 他の変数名でもOK`
    - `float leftTopY;  // hidariueX, hidariueY でもOK`
  - 四角形の左上の座標をランダムに決める
    - `random(最小値, 最大値);` で、最小値から最大値までの値を求めることができる
      - `leftTopX = random(50, 350); // 50～350のランダムな値`

      などのように・・・

宝探し（四角形を探す）

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）
  - mouseX がどういった条件を満たせば表示する？
    - ➔ leftTopX ≦ mouseX ≦ leftTopX+30
  - mouseY がどういった条件を満たせば表示する？
    - ➔leftTopY ≦ mouseY ≦ leftTopY+20
  - 2つの条件を満たした時に長方形を表示する！
    - 長方形は rect(左上x, 左上y, 横幅, 縦幅); で描画

```
float leftTopX;
float leftTopY;

void setup()
{
  size(400, 300);
  leftTopX = random(0, 390); // 0～390までの任意の値
  leftTopY = random(0, 290); // 0～290までの任意の値
}

void draw()
{
  background(255, 255, 255);
  if((mouseX >= leftTopX) && (mouseX <= leftTopX + 30))
  {
    if((mouseY >= leftTopY) && (mouseY <= leftTopY + 20))
    {
      rect(leftTopX, leftTopY, 30, 20);
    }
  }
}
```

400までにしちゃうとはみ出て見つからない

宝探し（四角形を探す）

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


```
float leftTopX;
float leftTopY;

void setup()
{
  size(400, 300);
  leftTopX = random(width - 10);
  leftTopY = random(height - 10);
}

void draw()
{
  background(255, 255, 255);
  if((mouseX >= leftTopX) && (mouseX <= leftTopX + 30) &&
     (mouseY >= leftTopY) && (mouseY <= leftTopY + 20))
  {
    rect(leftTopX, leftTopY, 30, 20);
  }
}
```

もちろん全部&&でつないでもOK

2行で書いてもOK!

宝探しで関数を使おう！

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

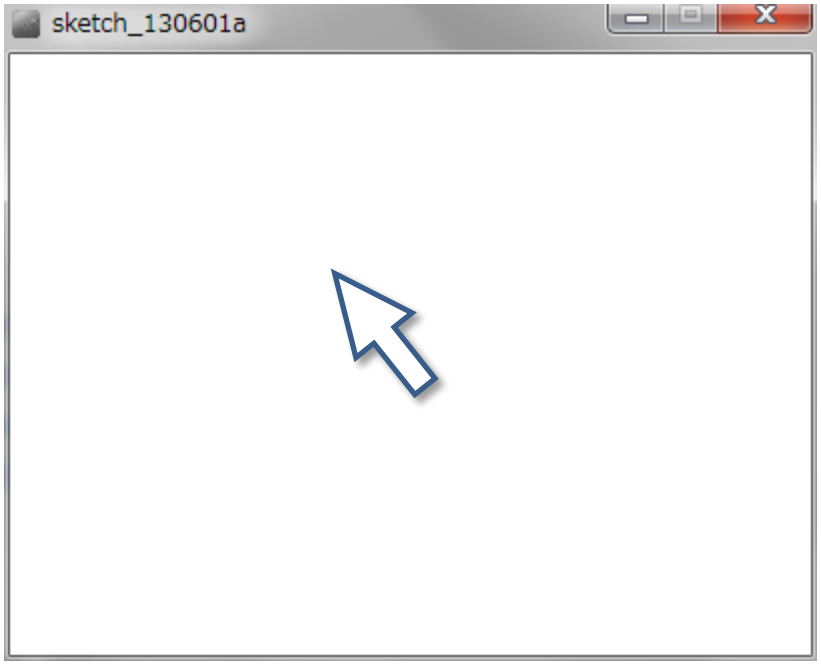
```
float leftTopX, leftTopY;

void setup() {
  size(400, 300);
  leftTopX = random(width - 10);
  leftTopY = random(height - 10);
}

boolean isInside(int x, int y, int ltx, int lty, int rbx, int rby) {
  if(ltx <= x && x <= rbx && lty <= y && y <= rby){
    return true;
  }
  return false;
}

void draw() {
  background(255, 255, 255);
  if(isInside(mouseX, mouseY, leftTopX, leftTopY, leftTopX+30, leftTopY+20))
  {
    rect(leftTopX, leftTopY, 30, 20);
  }
}
```

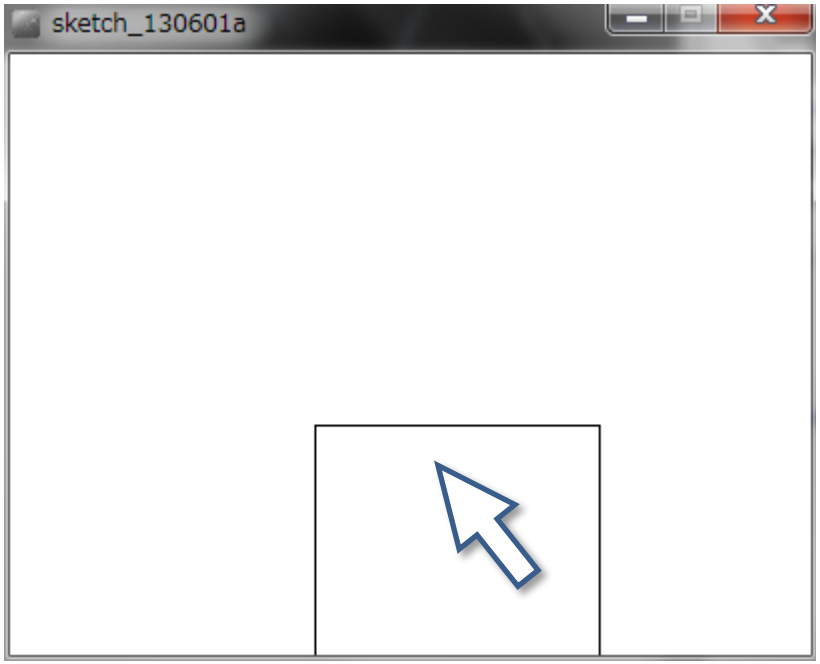
自分が作った関数を使おう！

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

- ランダムな場所に、ランダムな大きさで配置された四角形を探す（カーソルがその上にあるときだけ表示する）プログラムを作成しよう！
  - 四角形の左上の座標、縦幅横幅を実数の変数に（leftTopX, leftTopY, rectWidth, rectHeightなど）

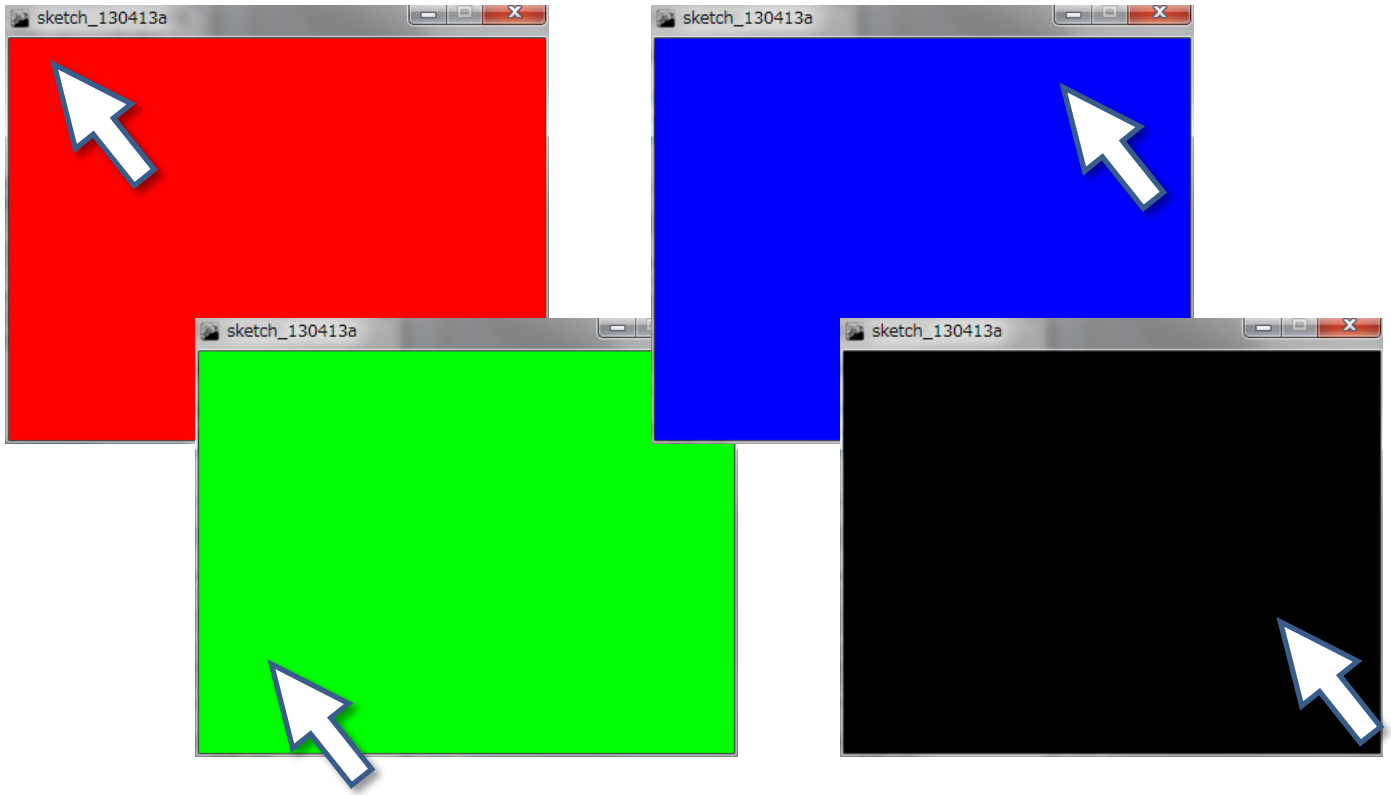




色々な条件に挑戦

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) 400x300のウインドウでマウスが画面の左上で赤背景、右上で青背景、左下で緑背景、右下で黒背景にするには？



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

- 条件（「<」にするか「<=」にするかは人次第）
  - 左上は mouseX < 200 かつ mouseY < 150 ➔ 赤色
  - 右上は mouseX >= 200 かつ mouseY < 150 ➔ 青色
  - 左下は mouseX < 200 かつ mouseY >= 150 ➔ 緑色
  - 右下は mouseX >= 200 かつ mouseY >= 150 ➔ 黒色

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


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

補足

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

- if の処理内容が1つだけのときは { } を省略可能

```
if((mouseX < 200) && (mouseY < 150))
{
  background(255, 0, 0);
}
```

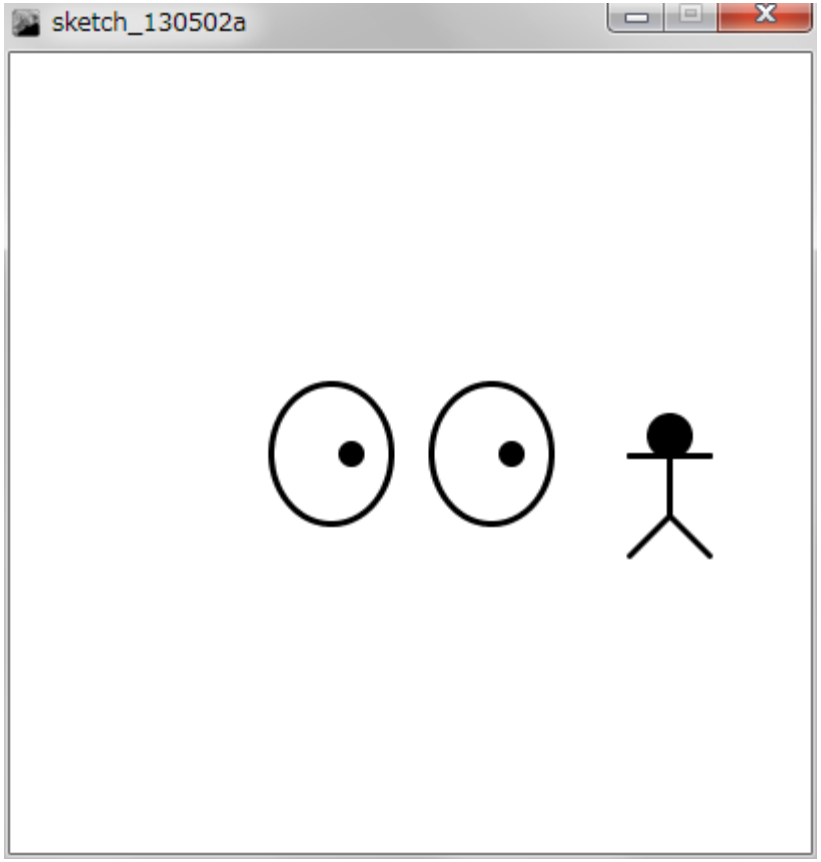
```
if((mouseX < 200) && (mouseY < 150))
  background(255, 0, 0);
```

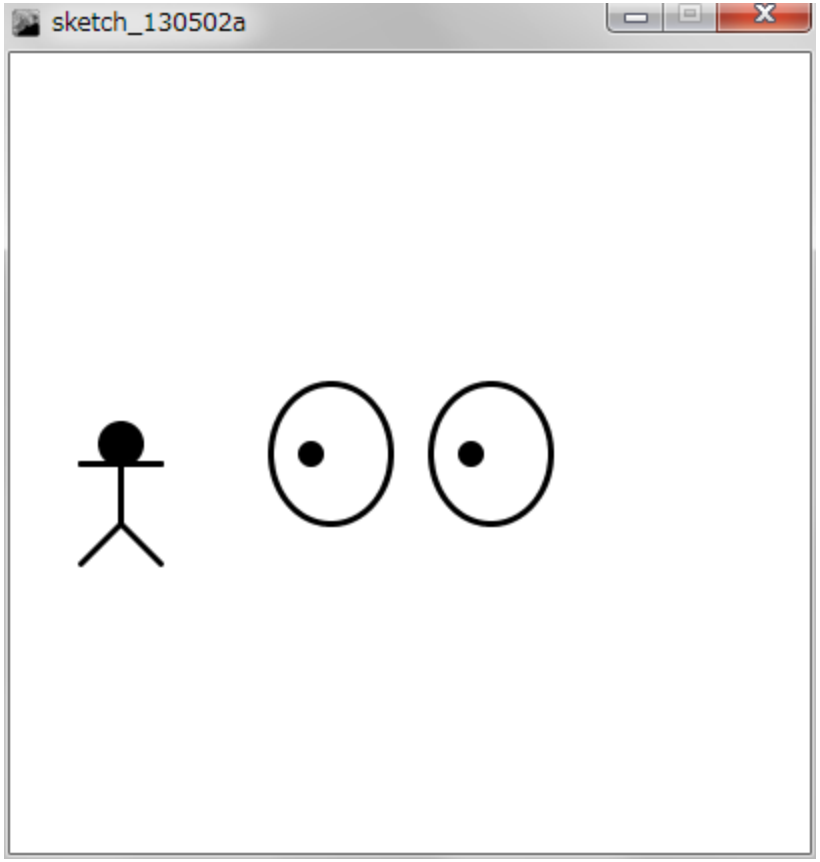
ただ、慣れるまでは省略しない！

関数でアニメーション

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) マウスの位置に応じて黒目を動かすプログラムを作ってみよう！（目玉で1つの関数）





関数でアニメーション

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
  - 目玉の場所（x，y）と、黒目が向く方向を決めるための座標（mx，my）を引数にする
  - mx，my はマウス座標を入力
  - 白目を（x，y）を中心とし適当な楕円で描く
  - x と mx の位置関係で黒目の座標を決め黒目を描く
    - mx < x -10 なら黒目を左へ（10は適当な値）
    - mx > x + 10 なら黒目を右へ
    - そうでなければ黒目を真ん中へ
  - drawEye を2つdraw()の中に書く！
  - drarwHuman は前に作ったのを使う

関数でアニ

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

void drawEye(int x, int y, int mx, int my)
{
  strokeWeight(3);
  fill(255, 255, 255);
  circle(x, y, 60, 70);
  fill(0, 0, 0);
  if(x > mx + 10)
  {
    circle(x - 10, y, 10);
  }
  else if(x < mx - 10)
  {
    circle(x + 10, y, 10);
  }
  else
  {
    circle(x, y, 10);
  }
}
void draw()
{
  background(255);
  drawHuman(mouseX, mouseY);
  drawEye(160, 200, mouseX, mouseY);
  drawEye(240, 200, mouseX, mouseY);
}
```

引数は4つ

描画だけなので
返り値はなし

x と mx の位置で
条件分岐

値で沢山分岐する

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

- キーボードからの入力は「void keyPressed()」で取得、入力キーは変数の「key」を調べるだけ！
  - 沢山 if - else if - else if - else if - else if…と繋げても良いが、見通しがやや悪くなる
  - switchを使おう！

```
switch(判断する変数)
{
  case 条件A:
    println(“条件Aです！”);
    break;
  case 条件B:
    println(“条件Bです！”);
    break;
  default:
    println(“それ以外です！”);
    break;
}
```

switch - case - break

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


```
switch(分岐させる変数){
  case 値1:
    値1の時の処理
    break; // 値1の時の処理ここまで
  case 値2:
    値2の時の処理
    break; // 値2の時の処理ここまで
  default:
    // 値1でも2でもない場合の処理
    break;
}
```

値で沢山分岐する

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


```
void keyPressed()
{
  if(key == 'a')
  {
    println("Aが押されました");
  }
  else if(key == 'b')
  {
    println("Bが押されました");
  }
  else if(key == 'c')
  {
    println("Cが押されました");
  }
  else
  {
    println("それ以外のキー");
  }
}
```

```
void keyPressed()
{
  switch(key)
  {
    case 'a':
      println("Aが押されました");
      break;
    case 'b':
      println("Bが押されました");
      break;
    default:
      println("それ以外のキー");
      break;
  }
}
```

上下左右

key ではなく
keyCode という
変数を利用

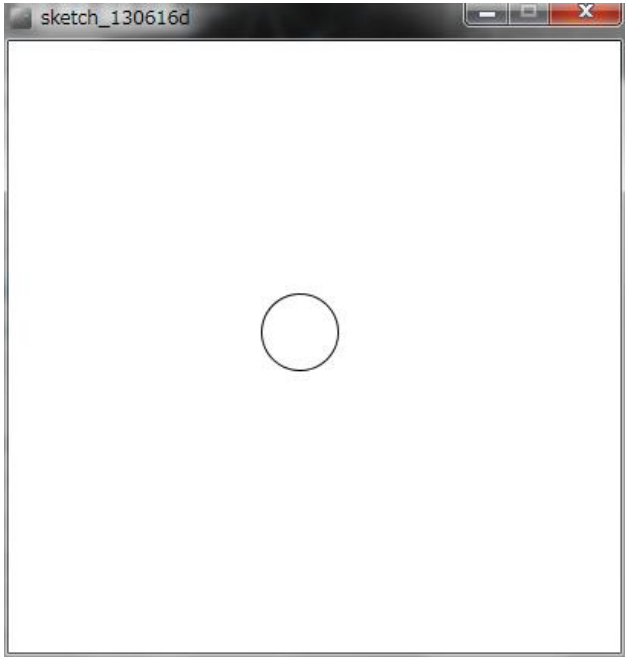
```
void keyPressed()
{
  switch(keyCode)
  {
    case UP:
      println("upが押されました");
      break;
    case DOWN:
      println("downが押されました");
      break;
    case LEFT:
      println("leftが押されました");
      break;
    case RIGHT:
      println("rightが押されました");
      break;
    default:
      println("それ以外のキー");
      break;
  }
}
```

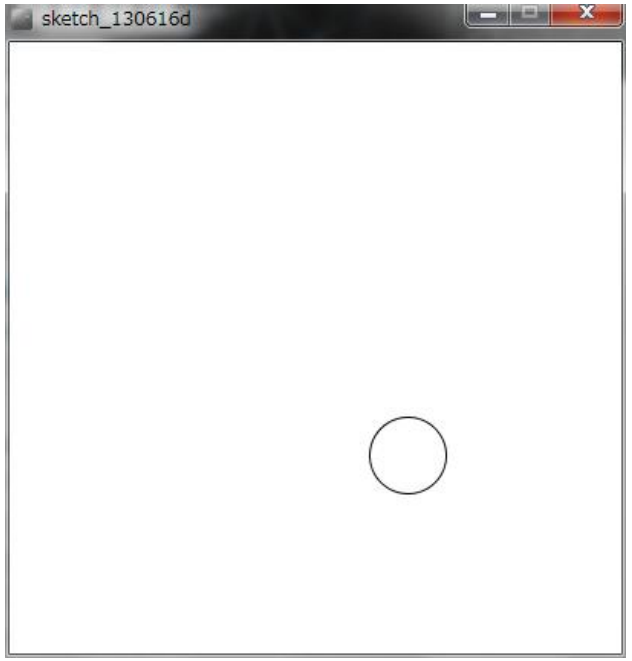
up/down/left/rightで移動

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) 上下左右キーで円を動かす

- keyPressed() で入力を取得
- keyCode で上下左右ボタンを取得
- 座標を up/down/right/left で変更





up/down/left/rightで移動

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(400, 400);
}

int x = 200;
int y = 200;
void draw()
{
  background(255, 255, 255);
  circle(x, y, 50);
}
```

```
void keyPressed()
{
  switch(keyCode)
  {
    case UP:
      y = y - 10;
      break;
    case DOWN:
      y = y + 10;
      break;
    case LEFT:
      x = x - 10;
      break;
    case RIGHT:
      x = x + 10;
      break;
  }
}
```

up/down/left/rightで移動

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(400, 400);
}

int x = 200;
int y = 200;

void draw()
{
  background(255, 255, 255);
  circle(x, y, 50);
}
```

```
void keyPressed()
{
  if(keyCode == UP)
  {
    y = y - 10;
  }
  else if(keyCode == DOWN)
  {
    y = y + 10;
  }
  else if(keyCode == LEFT)
  {
    x = x - 10;
  }
  else if(keyCode == RIGHT)
  {
    x = x + 10;
  }
}
```

マウスクリックで占う

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) マウスクリックするたびに標準出力に占い結果を表示するプログラムを作る

- 考え方
  - マウスクリックした際に 0〜9 の乱数を発生
  - 前にも登場したけど乱数を発生は random(...);

```
num = (int) random(10);
```

  - 乱数の値を変数に格納
  - 変数の値に応じて占い結果を表示する！

マウス

```
void setup()
{
  size(100, 100);
}

void draw()
{
}

void mousePressed()
{
  num = (int)random(0,10);
  switch(num)
  {
    case 0:
      println("DAIKICHi!! (^o^)");
      break;
    case 1:
      println("DAIKYO!! (X_X)");
      break;
    case 2:
      println("KICHI  (-_-)");
      break;
      // ～以降 3 ～ 9 まで繰り返し
  }
}
```

予習問題

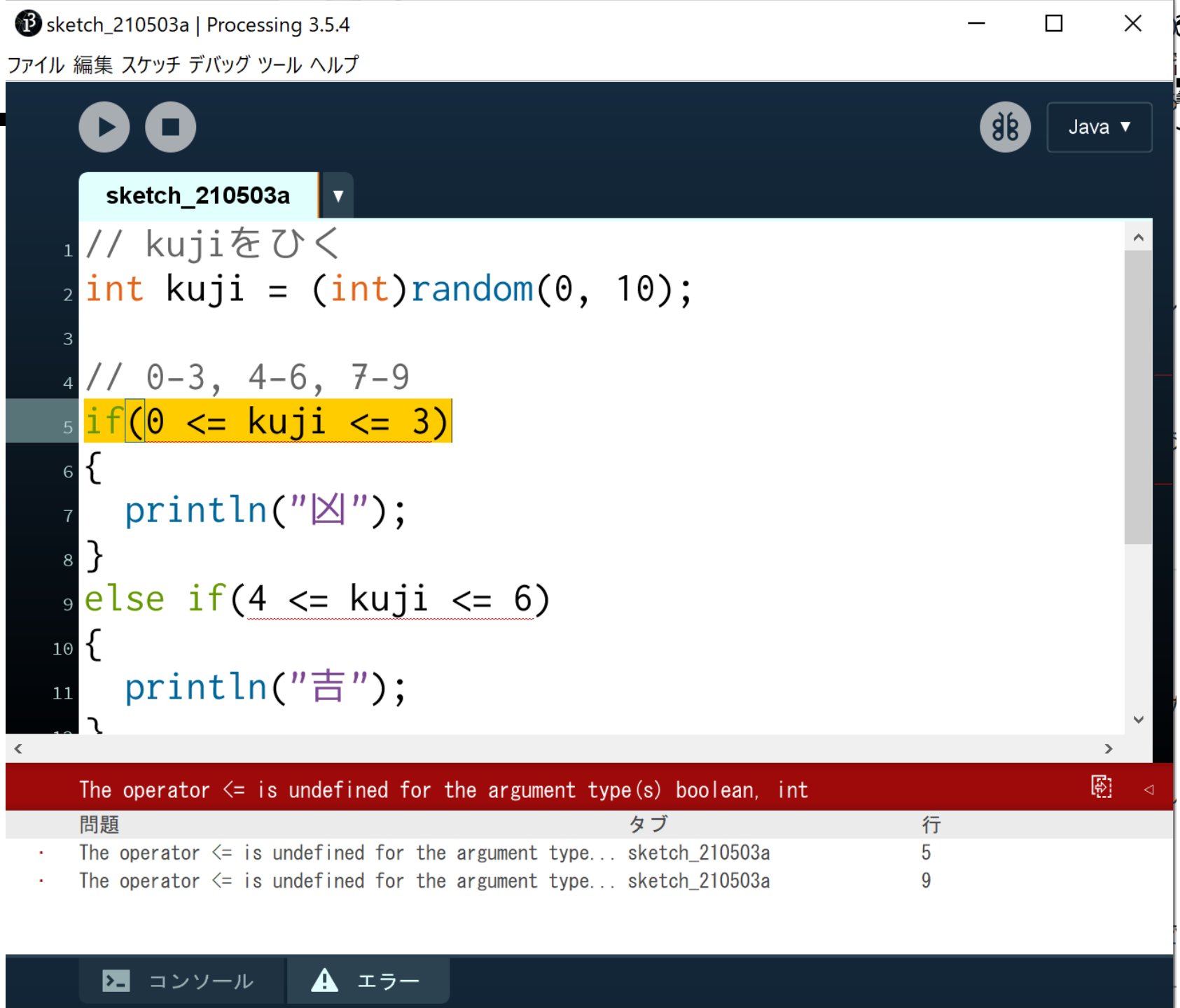
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 画面に表示されていない左から右に動く四角形を探すプログラムに挑戦
  - 惜しい場合はヒントを表示しましょう！
- 画面のクリック場所によって占いの結果を変更するプログラムを作ってみましょう
- 左から右に動く白色の四角形をマウスでクリックすると赤色で塗りつぶすプログラムに挑戦

今回のエラーのコーナー

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

- これからは画面だけをスクショで示します


sketch_210503a | Processing 3.5.4
ファイル 編集 スケッチ デバッグ ツール ヘルプ
Java
sketch_210503a
// kujiをひく
int kuji = (int)random(0, 10);
// 0-3, 4-6, 7-9
if(0 <= kuji <= 3)
{
println("凶");
}
else if(4 <= kuji <= 6)
{
println("吉");
The operator <= is undefined for the argument type(s) boolean, int
問題
タブ
行
The operator <= is undefined for the argument type... sketch_210503a 5
The operator <= is undefined for the argument type... sketch_210503a 9
コンソール
エラー

sketch_210503a | Processing 3.5.4

ファイル 編集 スケッチ デバッグ ツール ヘルプ

Java ▼

sketch_210503a

```
1  // kujiをひく
2  int kuji = (int)random(0, 10);
3
4  // 0-3, 4-6, 7-9
5  if(0 <= kuji <= 3)
6  {
7    println("凶");
8  }
9  else if(4 <= kuji <= 6)
10 {
11   println("吉");
12 }
```

<= が定義されていない？

A <= B <= C はダメ！
～以上～以下の場合は、
一度に書くのではなく
各条件を&&でつなぐ！

The operator <= is undefined for the argument

	問題	タブ	行
・	The operator <= is undefined for the argument type...	sketch_210503a	5
・	The operator <= is undefined for the argument type...	sketch_210503a	9

コンソール　エラー

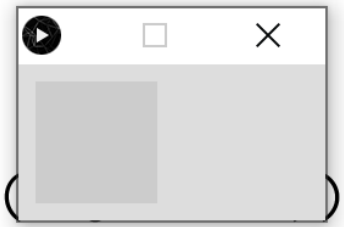
sketch_210503a | Processing 3.5.4

ファイル 編集 スケッチ デバッグ ツール ヘルプ

エラーが出てないけど
おかしい！

sketch_210503a

```
1  // kujiをひく
2  int kuji = (int)random(0, 10);
3
4  // 0-3, 4-6, 7-9
5  if((kuji >= 0) & (kuji <= 3))
6  {
7    println("凶");
8  }
9  else if((kuji >= 4) & (
10 {
11   println("吉");
12 }
```



問題　タブ　行

コンソール　エラー

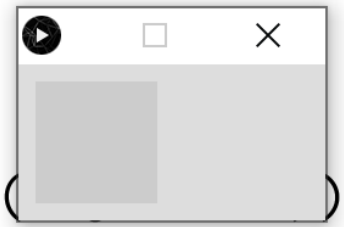
sketch_210503a | Processing 3.5.4

ファイル 編集 スケッチ デバッグ ツール ヘルプ

Java ▼

sketch_210503a ▼

```
1  // kujiをひく
2  int kuji = (int)random(0, 10);
3
4  // 0-3, 4-6, 7-9
5  if((kuji >= 0) & (kuji <= 3))
6  {
7    println("凶");
8  }
9  else if((kuji >= 4) & (
10 {
11   println("吉");
```



問題

コンソール　エラー

「&」ではなく「&&」であることに注意！

sketch_210511a | Processing 3.5.4

ファイル 編集 スケッチ デバッグ ツール ヘルプ

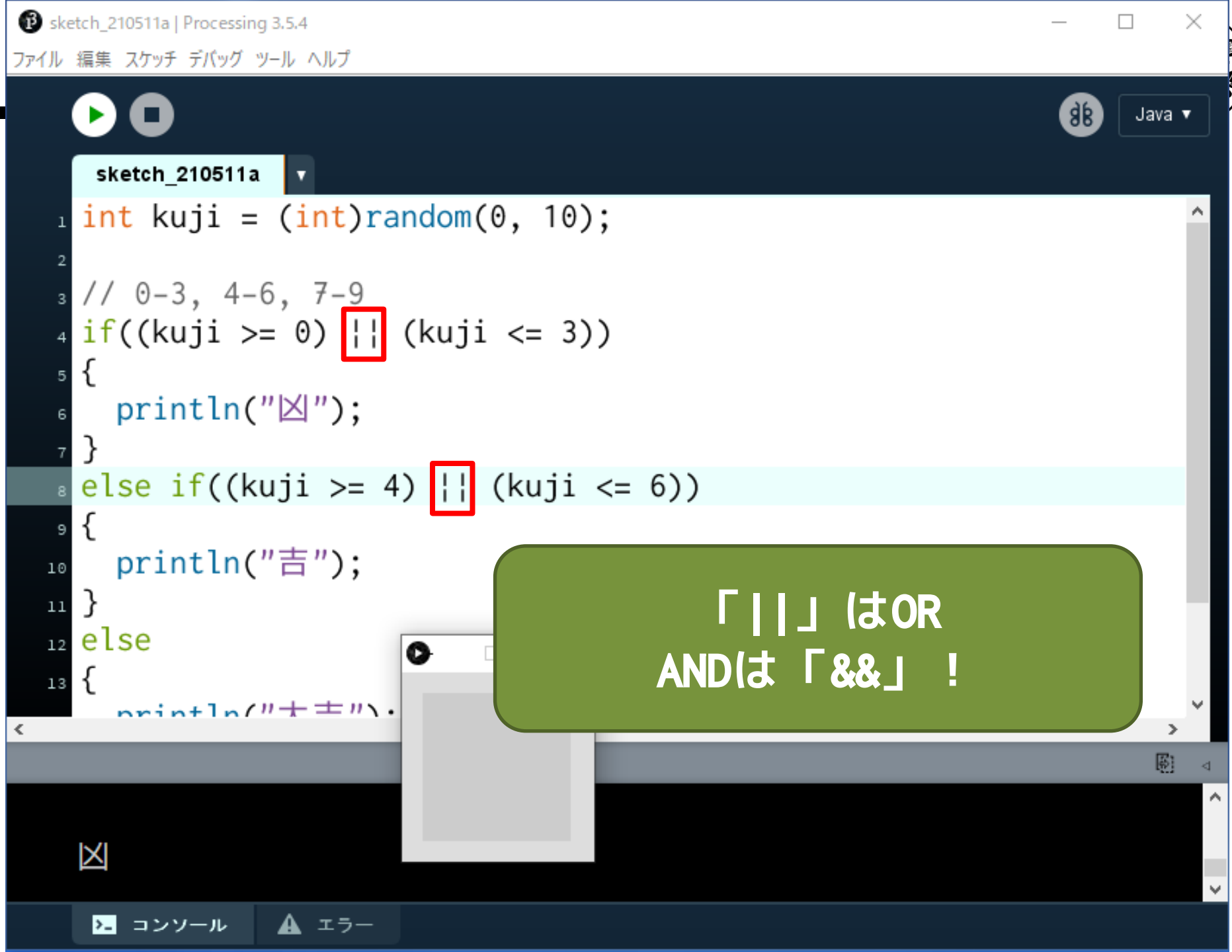
エラーが出てないけど
全部「凶」になる！

sketch_210511a

```
1  int kuji = (int)random(0, 10);
2
3  // 0-3, 4-6, 7-9
4  if((kuji >= 0) || (kuji <= 3))
5  {
6    println("凶");
7  }
8  else if((kuji >= 4) || (kuji <= 6))
9  {
10   println("吉");
11 }
12 else
13 {
```

凶

コンソール エラー


sketch_210511a | Processing 3.5.4
ファイル 編集 スケッチ デバッグ ツール ヘルプ
Java
sketch_210511a
int kuji = (int)random(0, 10);
// 0-3, 4-6, 7-9
if((kuji >= 0) || (kuji <= 3))
{
println("凶");
}
else if((kuji >= 4) || (kuji <= 6))
{
println("吉");
}
else
{
「||」はOR
ANDは「&&」 !
凶
コンソール
エラー

sketch_210511a | Processing 3.5.4

ファイル 編集 スケッチ デバッグ ツール ヘルプ

Java ▾

sketch_210511a ▾

```
4
5  void draw()
6  {
7  }
8
9  void keyPressed()
10 {
11   if(key == a)
12   {
13     println("Aが押されました");
14   }
15   else if(key == b)
16   {
     println("Bが押されました");
```

a cannot be resolved to a variable

	問題	タブ	行
・	変数 "a" は存在しません	sketch_210511a	11
・	変数 "b" は存在しません	sketch_210511a	15
・	変数 "c" は存在しません	sketch_210511a	19

コンソール エラー

sketch_210511a | Processing 3.5.4

ファイル 編集 スケッチ デバッグ ツール ヘルプ

Java ▾

sketch_210511a

```
4
5  void draw()
6  {
7  }
8
9  void keyPressed()
10 {
11   if(key == a)
12   {
13     println("Aが押されました");
14   }
15   else if(key == b)
16   {
     println("Bが押されました");
```

キーが押されたかどうかの
判定は ’a’ と ’’ で囲う

a cannot be resolved to a variable

	問題	タブ	行
・	変数 "a" は存在しません	sketch_210511a	11
・	変数 "b" は存在しません	sketch_210511a	15
・	変数 "c" は存在しません	sketch_210511a	19

コンソール エラー