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2
関数（メソッド）

中村，小松，菊池

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

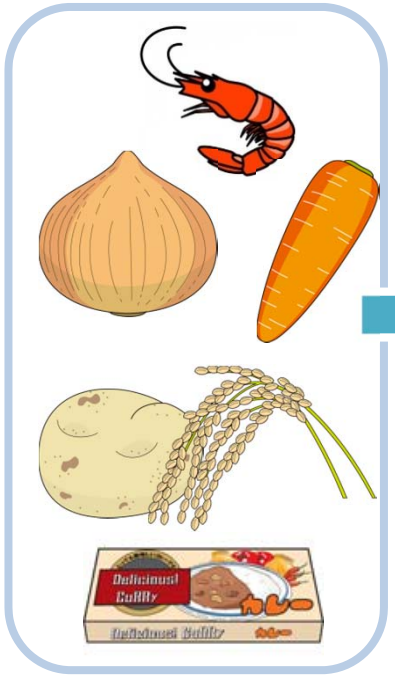
- 関数（メソッド）の理解を深める

- 静的メソッドって何やねん？
- インスタンスメソッドって何やねん？
- ってか，そもそもよくわからん．

関数（メソッド）とは

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

- 何かの処理をしてくれる「処理機械」みたいな物



なんか
処理機

「なんか処理機」の中は
良くわからないブラックボックス
だけど使える！
具材入れたらカレーができる！

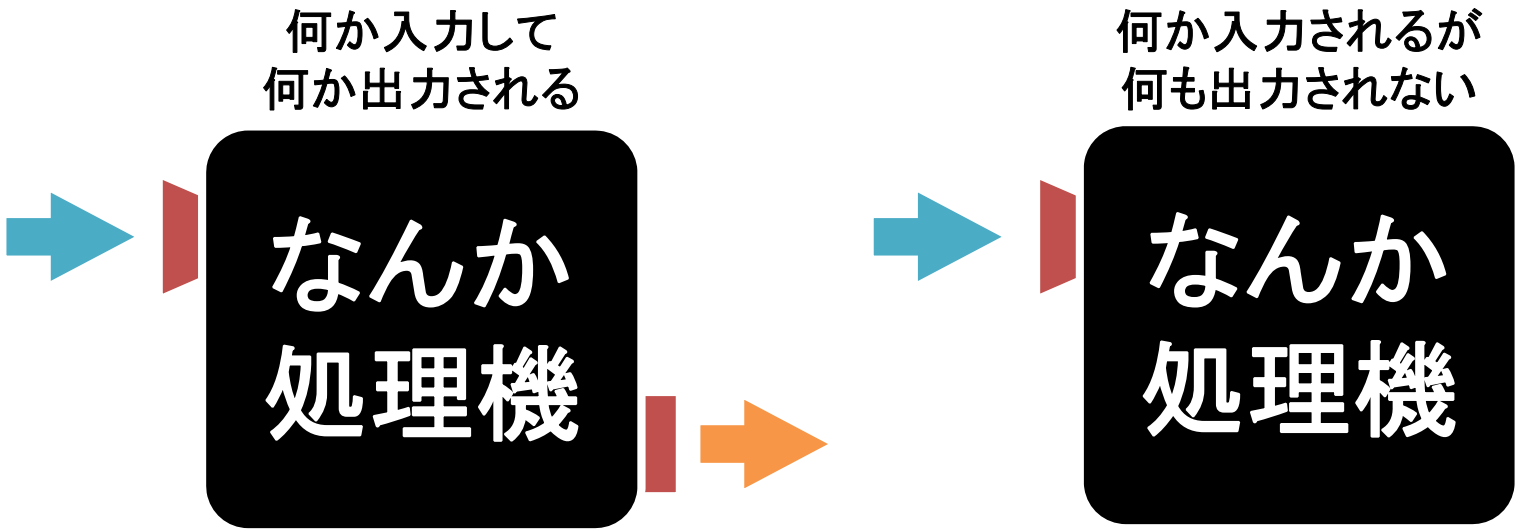
例：関数（メソッド）

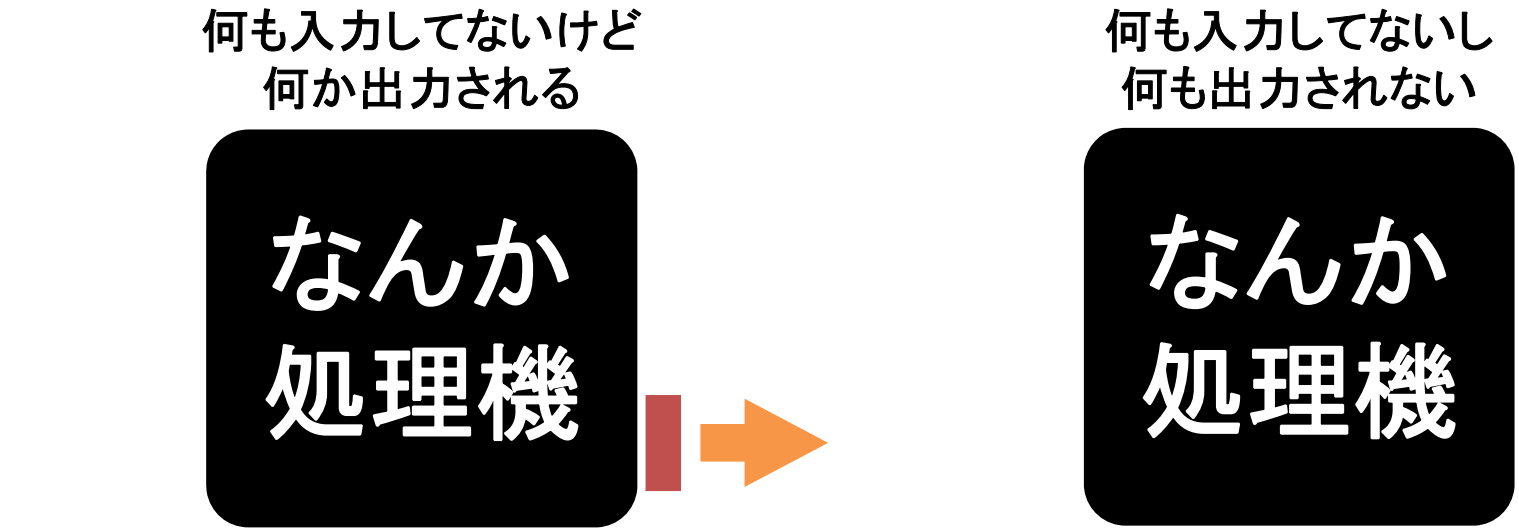
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 電子レンジ
  - 冷えたお弁当を入れてボタンを押すと，温まったお弁当が手に入る
- ポット
  - ボタンを押すとなんかお湯が出てくる
- 冷蔵庫
  - 食材を冷蔵庫に入れる（保存する）
- ストップウォッチ
  - ボタンを押すとラップタイプが記録されているらしい

メソッドは4種類

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





わかりにくいけれど

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

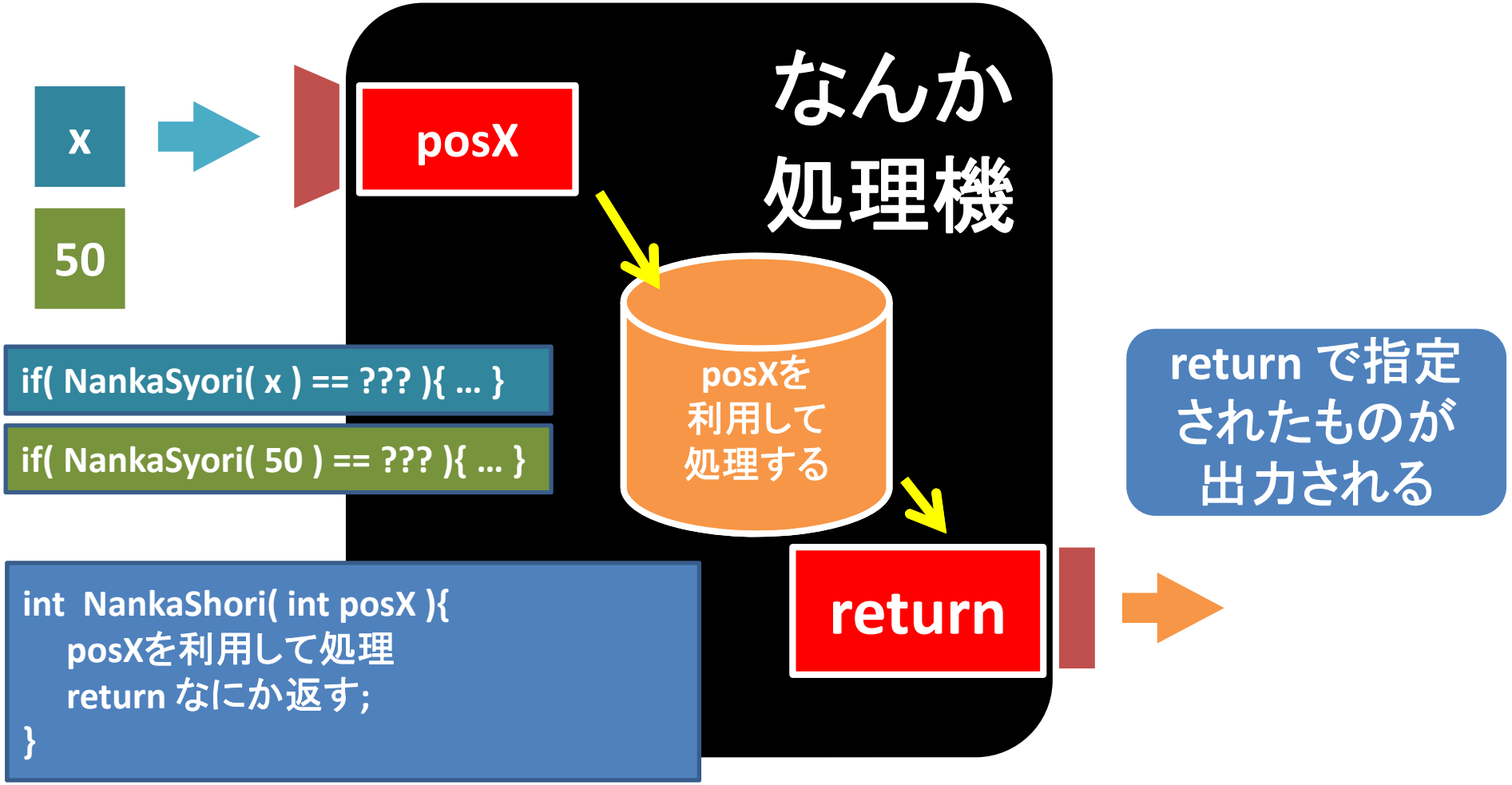
- メソッドは何らかの出力をするのでは？
  - 何かを画面に表示する
  - 何か音を鳴らす
  - 何かをファイルに出力する
  - クラス内変数の値を変更する
  - グローバル変数の値を何かに変更する

出力してるじゃん！！

- 上記はメソッドの明示的な出力ではなく副産物みたいなもの．メソッドとしての明示的な出力はある場合とない場合がある．

メソッドの内部処理

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

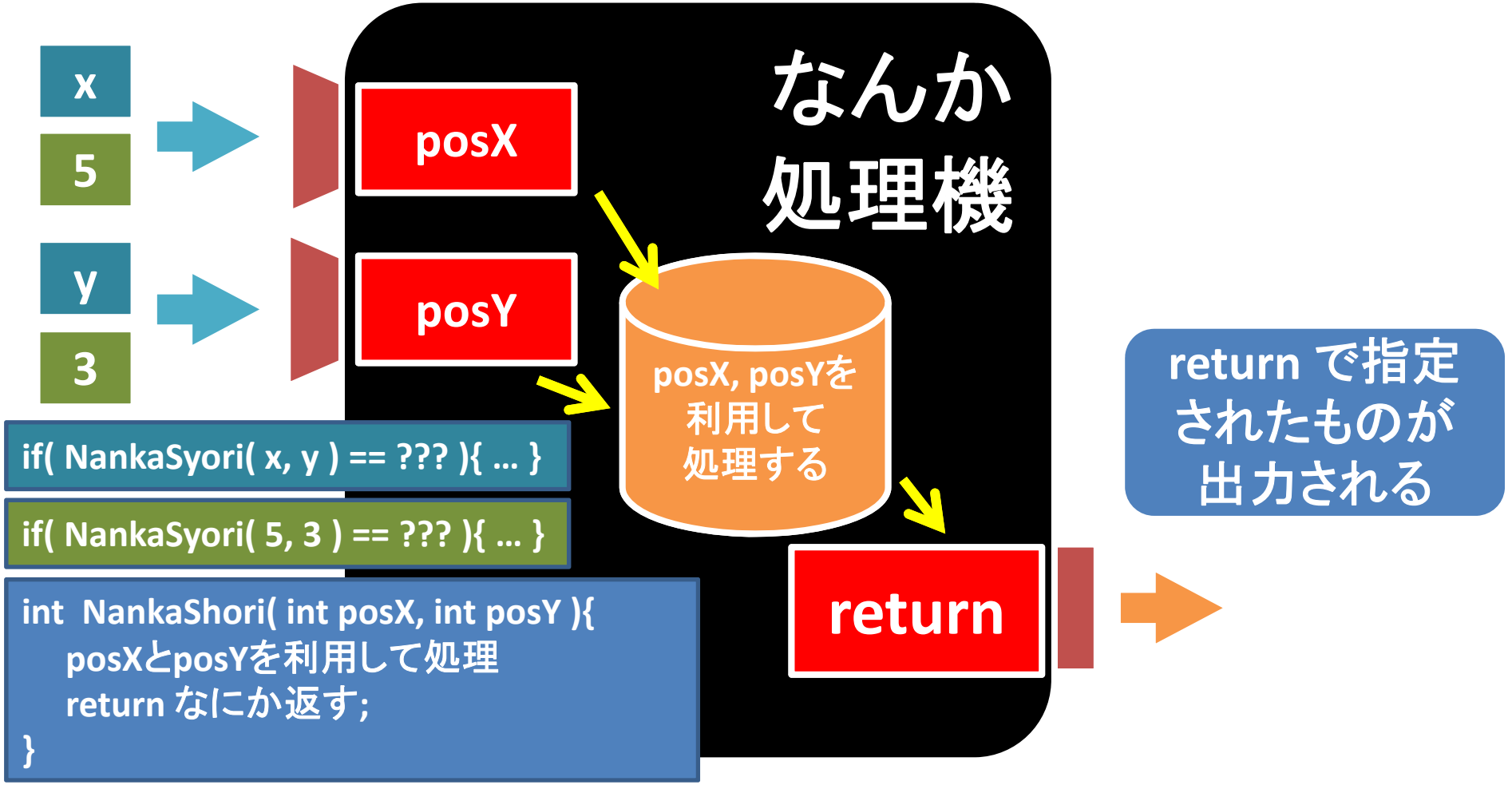


- 引数として取得した値は，引数で指定された変数名を利用して処理．returnで何か返される

メソッドの内部処理

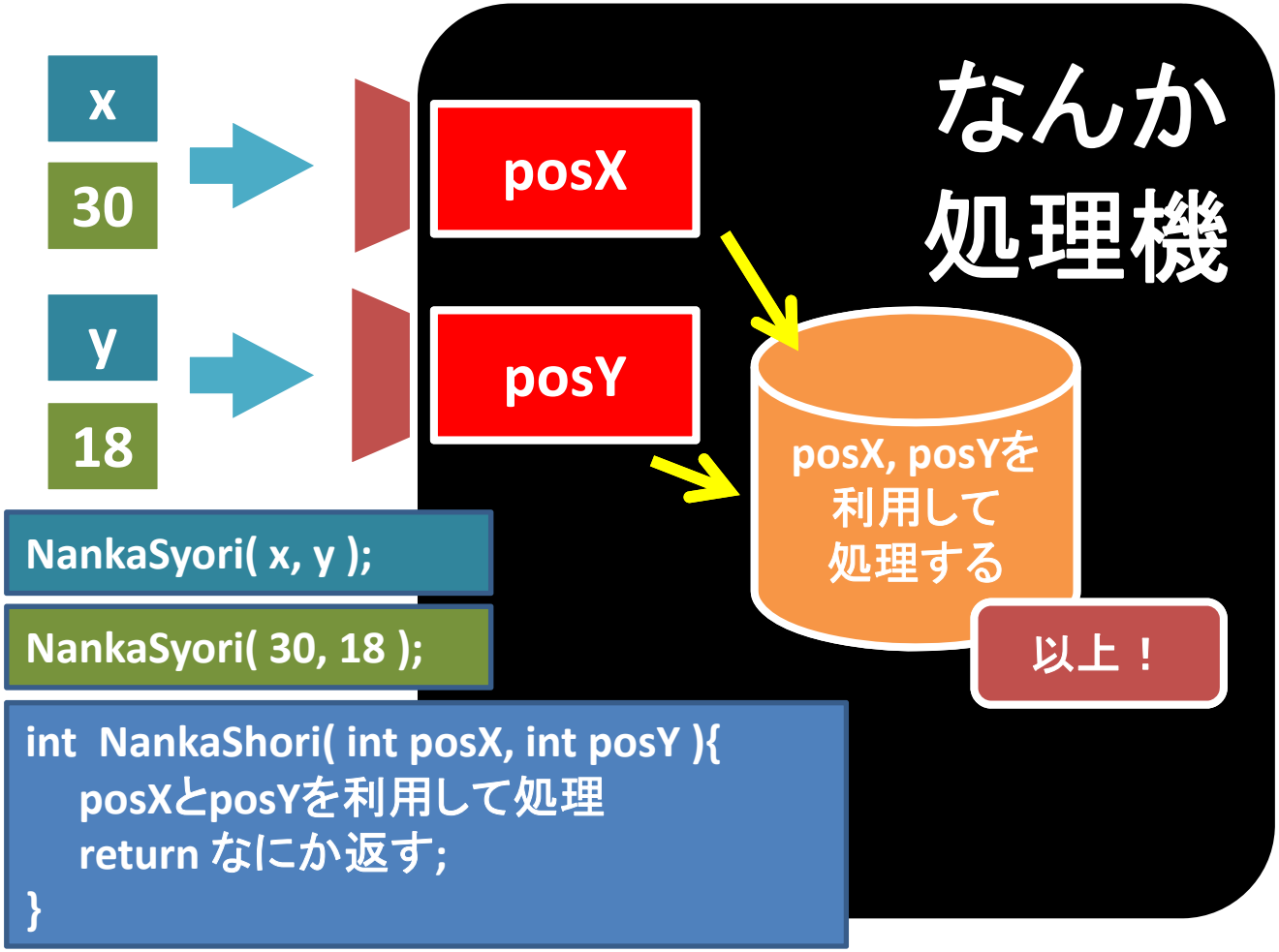
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- 引数は増えるけれど返り値（returnされる値）は増えない



メソッドの内部処理

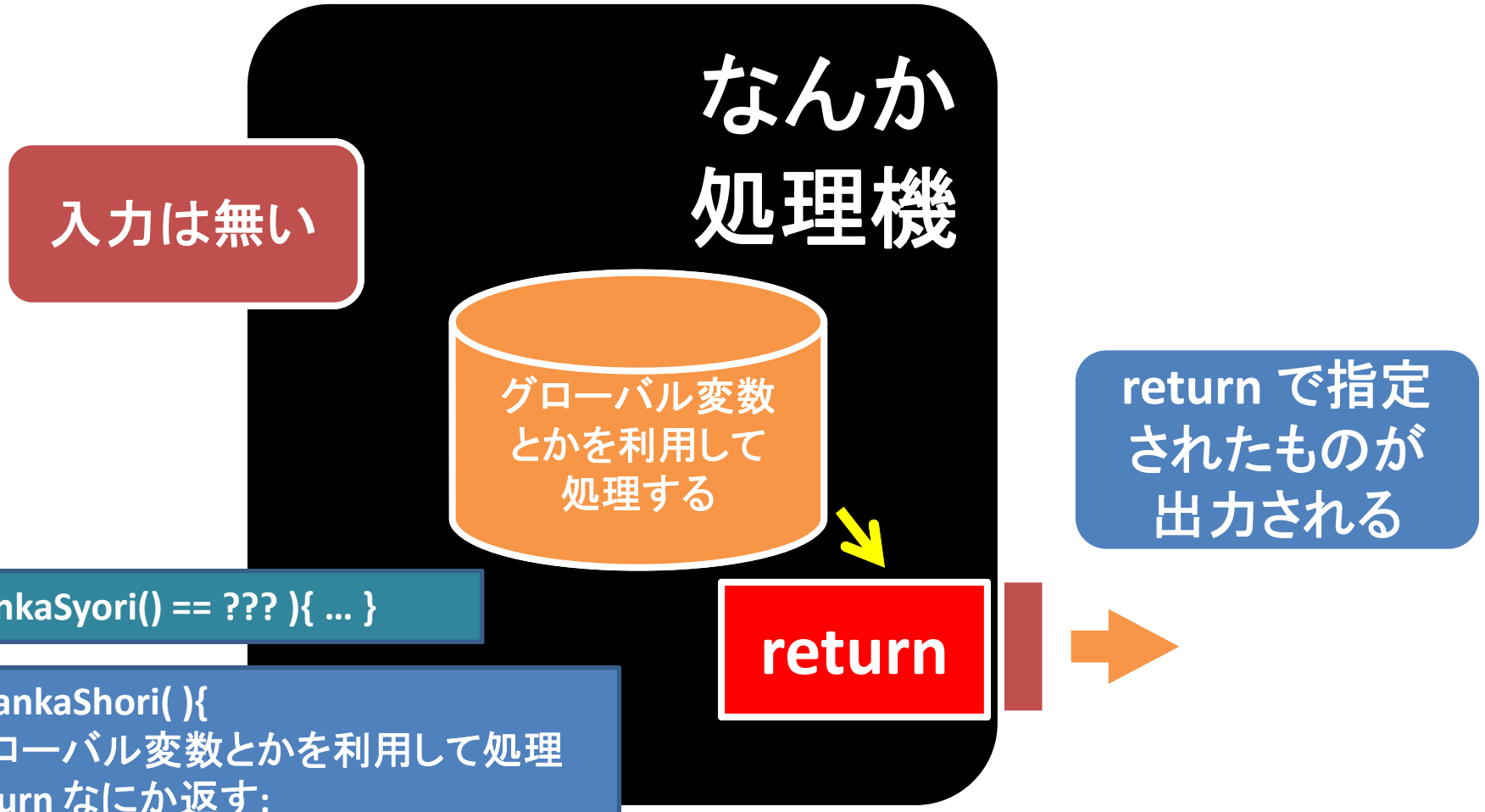
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室



- 引数は増えるけれど返り値（returnされる値）は増えない

メソッドの内部処理

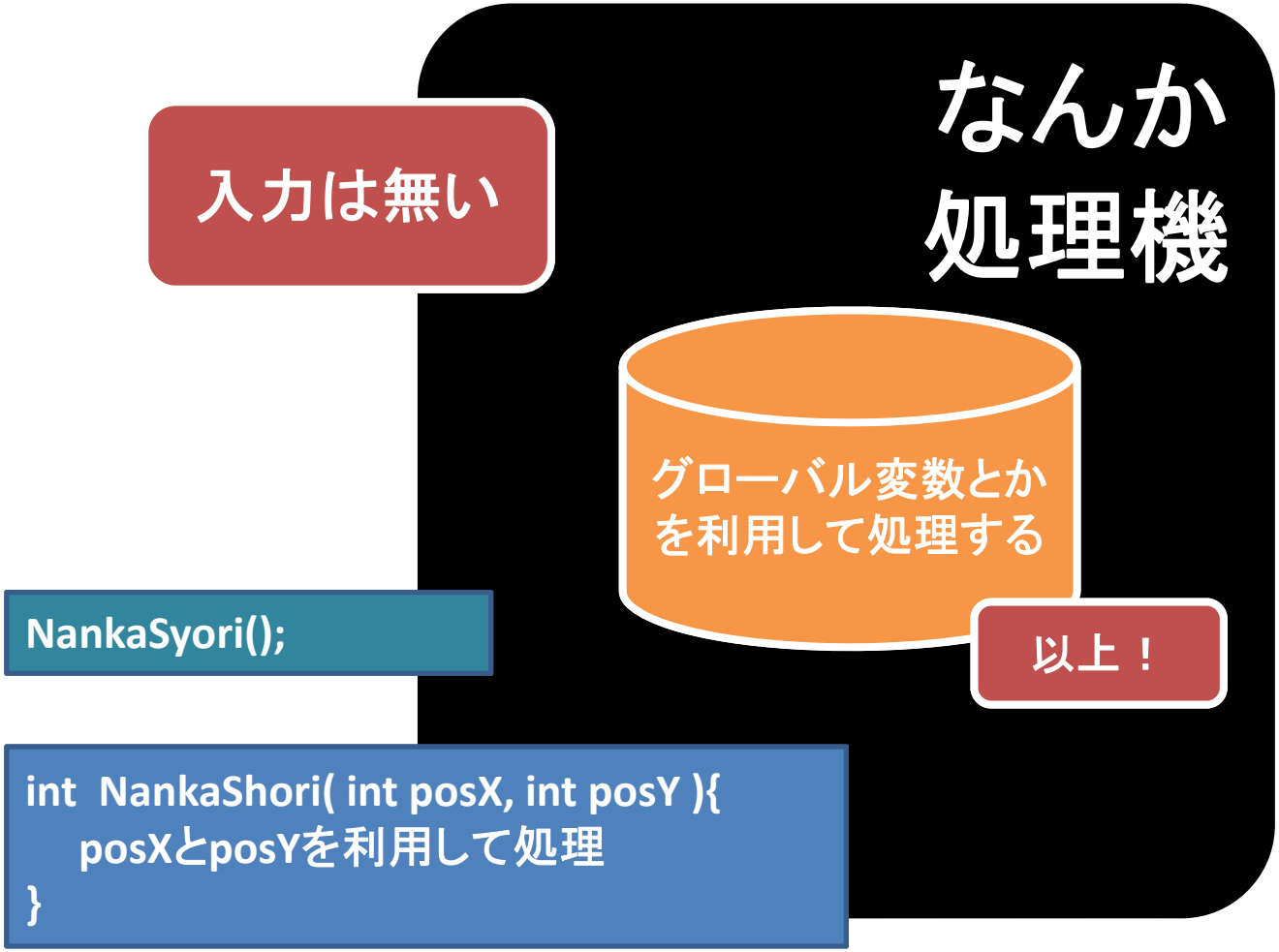
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室



- 返り値だけのケース（引数がない場合）

メソッドの内部処理

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



- 入力も出力もないメソッド

もう一度整理

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

何か入力して
何か出力される

何か入力されるが
何も出力されない

何も入力してないけど
何か出力される

何も入力してないし
何も出力されない

引数〇，返り値〇

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) ある入力された数字の約数の数を求める関数をどう作るか？ また，その関数を使って数字と約数の数のペアを出力しよう

```
17989: 2
17990: 16
17991: 6
17992: 16
17993: 4
17994: 8
17995: 8
17996: 12
17997: 8
17998: 4
17999: 4
18000: 60
```

約数の数を求める関数

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

- 考え方
  - 引数は整数型の num にする
  - 約数を数える整数型の変数 count を用意
  - 整数型の変数 i（1からnumまで1ずつ増やす）を用意し，num が i で割り切れたら count を1追加する
  - 最後に count の値を返す（return count;）
  - println で数と，返って来た値を表示する

約数の数を求める関数

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


```
int getNumberOfDivisor( int num ){
   int i=1;
   int count=0;
   while( i<=num ){
      if( num%i == 0 ){
         count++;
      }
      i++;
   }
   return  count;
}

void setup(){
   for( int i=1; i<100000; i++ ){
      println( i+": "+ getNumberOfDivisor(i) );
   }
}
```

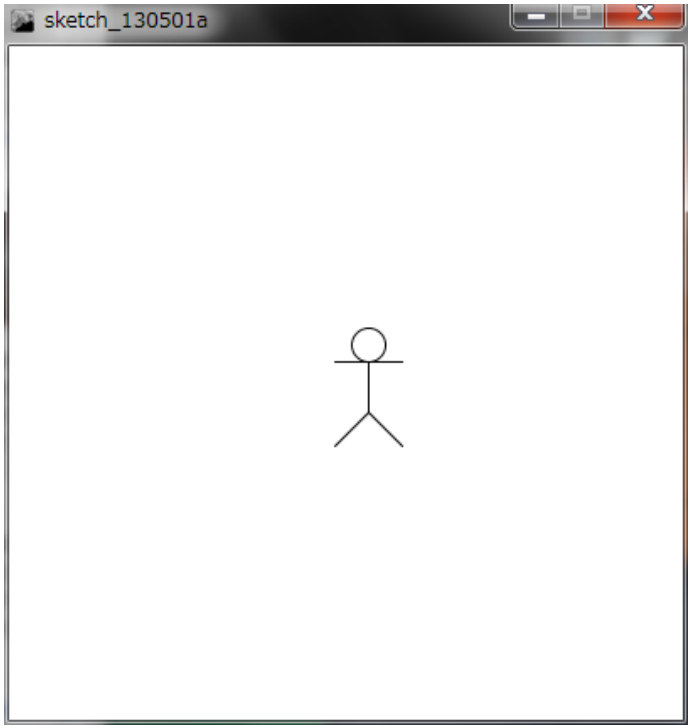
返り値（この値が，呼び出し元に返る）

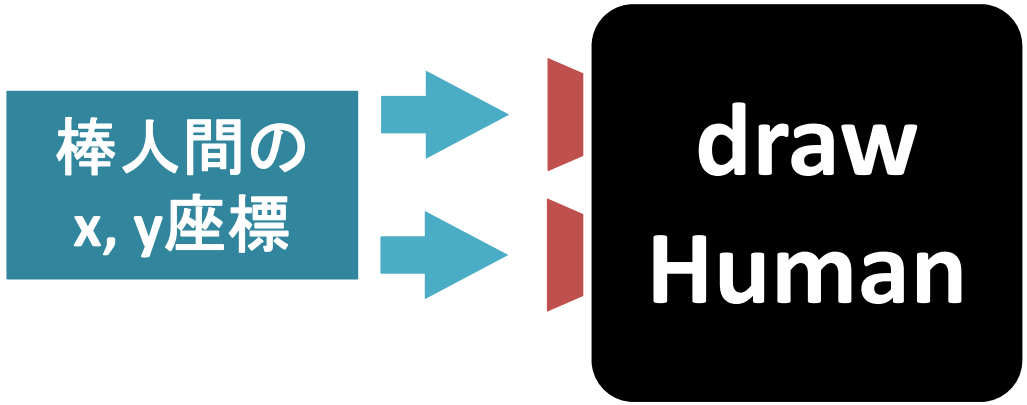
ここに戻る

引数○，返り値×

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) x, y 座標を指定すると棒人間を描いてくれる関数を作成せよ！

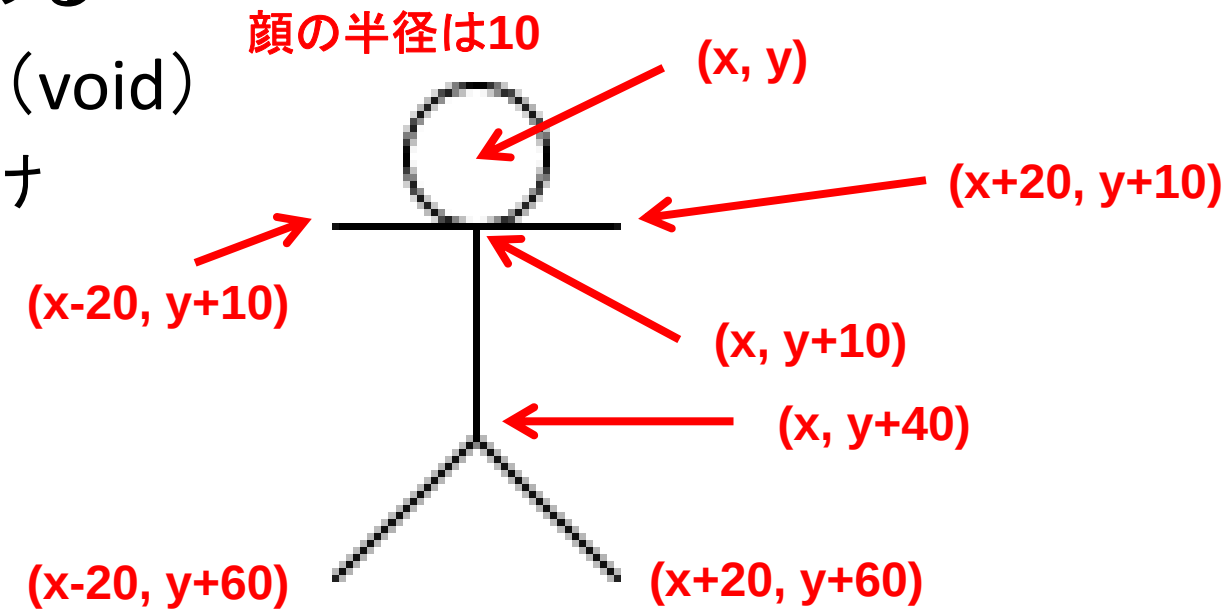




棒人間を描く

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
  - 棒人間は，顔の中心の座標(x, y)を与えると，勝手に体と手と足を描くものにする
  - 棒人間の中心の座標を(x, y)としたときのそれぞれの座標を決める
  - 返り値はなし(void)
    - 描画するだけ



棒人間を描く

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

- マウスカーソルの場所に棒人間を描く

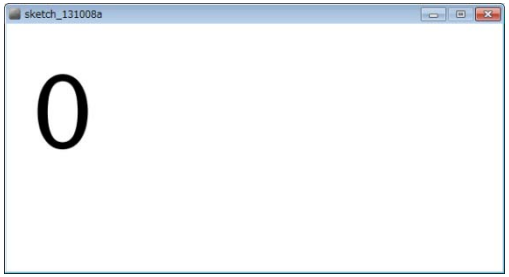
```
void setup(){
    size( 400, 400 );
}
void drawHuman( int x, int y ){
    ellipse( x, y, 20, 20 );
    line( x, y+10, x, y+40 );
    line( x-20, y+10, x+20, y+10 );
    line( x, y+40, x-20, y+60 );
    line( x, y+40, x+20, y+60 );
}
void draw(){
    background( 255 );
    drawHuman( mouseX, mouseY );
}
```

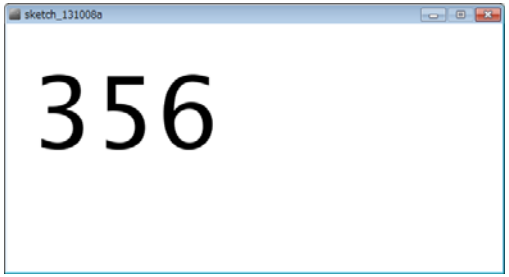
引数×，返り値○

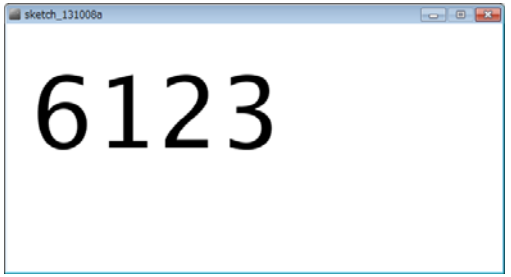
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

(Q) マウスをクリックしてから現在までの経過時間を表示するプログラムを作成し，再度クリックすると経過時間を表示するプログラムを作成せよ

- 考え方
  - 現在の時間を，0時0分0.0秒から換算して，何ミリ秒目かを変換する関数（メソッド）を作成する
    - 時間は hour(), 分は minute(), 秒は second(), ミリ秒は millis() で取得可能
    - メソッドは int getNow(){ … } という形にしよう！



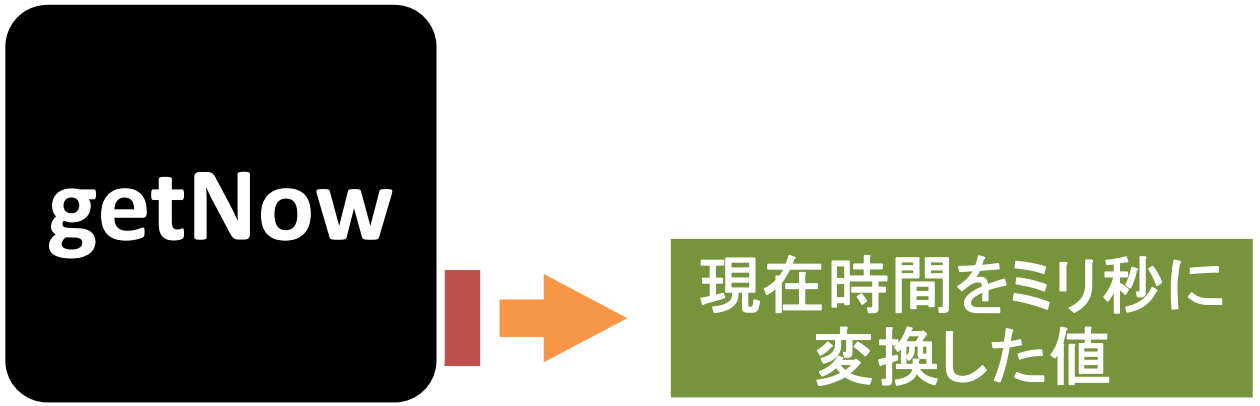




何ミリ秒目かを求める関数

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

- 考え方
  - boolean型（true/falseのみ）のstartingという変数を用意
    - starting==falseなら動かない，starting==trueなら動く
  - int型のstartTimeを用意し，マウスクリックされるとgetNow()を startTime に代入
  - int型のendTimeを用意し，マウスクリックされるとgetNow() を endTime に代入
  - starting==trueの間は，drawで getNow()-startTime の値を表示する！




```
int getNow()
{
   return ((hour()*60+minute())*60+second())*1000+millis();
}

boolean start = false;
int startTime = 0;
int endTime = 0;
void setup() {
   size( 600, 300 );
   textSize( 120 );
}
void draw() {
   background(255);
   fill(0);
   if ( start == true ) {
      text( (getNow()-startTime), 30, 150 );
   } else {
      text( (endTime-startTime), 30, 150 );
   }
}
```

```
void mousePressed() {
   if ( start == false ) {
      startTime = getNow();
      start = true;
   }
   else if ( start == true ) {
      endTime = getNow();
      start = false;
   }
}
```

getNow() 必要？

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

- ミリ秒は
  ((hour()*60+minute())*60+second())*1000+millis();
  で計算できるので getNow() はいらないのでは？
- 毎回書くのは面倒だけど，コピペしたらいいし

本当にそれで良いですか？

どっちが見やすい？

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


```
void draw() {
   background(255);
   fill(0);
   if ( start == true ) {
      text( (((hour()*60+minute())*60+second())*1000+millis()-startTime), 30, 150 );
   } else {
      text( (((hour()*60+minute())*60+second())*1000+millis()-startTime), 30, 150 );
   }
}
```

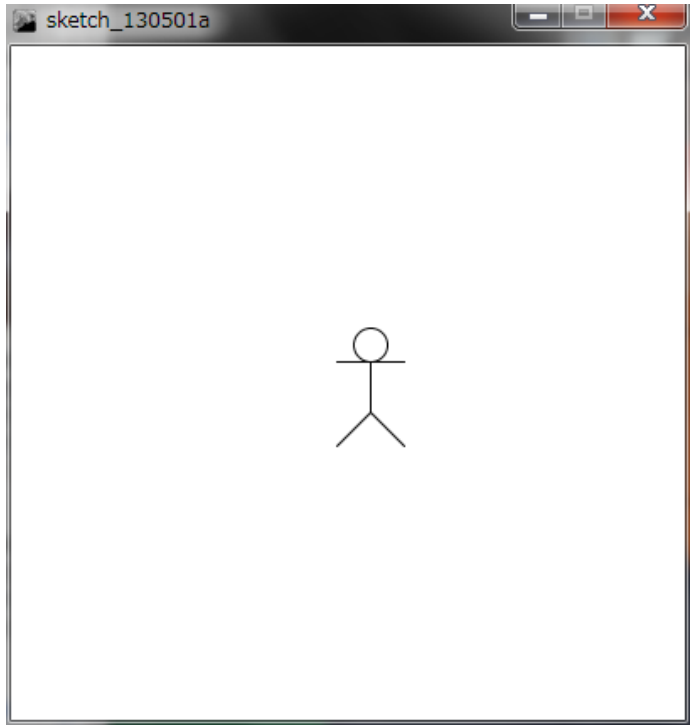
```
void draw() {
   background(255);
   fill(0);
   if ( start == true ) {
      text( (getNow()-startTime), 30, 150 );
   } else {
      text( (getNow()-startTime), 30, 150 );
   }
}
```

こちらなら
メソッドにした
方が良い！

引数×，返り値×

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) マウスカーソルの位置に棒人間を描いてくれる関数を作成せよ！



引数×，返り値×

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

- mouseX, mouseY はグローバル変数だし・・・

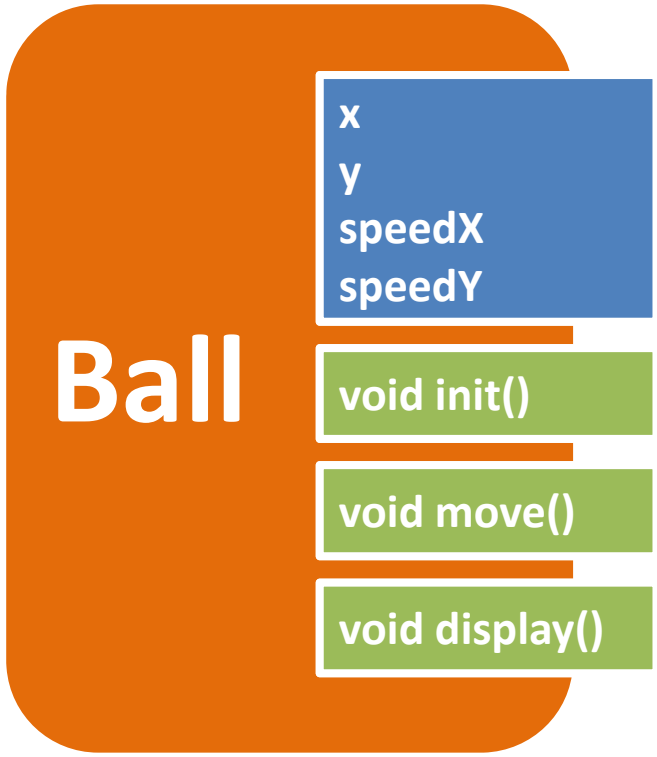
```
void setup(){
    size( 400, 400 );
}
void drawHuman2(){
    ellipse( mouseX, mouseY, 20, 20 );
    line( mouseX, mouseY+10, mouseX, mouseY+40 );
    line( mouseX-20, mouseY+10, mouseX+20, mouseY+10 );
    line( mouseX, mouseY+40, mouseX-20, mouseY+60 );
    line( mouseX, mouseY+40, mouseX+20, mouseY+60 );
}
void draw(){
    background( 255 );
    drawHuman2();
}
```

グローバル変数を使えば何もなくても良いが，改良しにくい

さて

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

- クラスのインスタンスメソッドには引数が無いものや返り値が無いものが多くなかった？



Ballの例の場合は，
全て引数・返り値無し

Ballクラス

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

```
class Ball{
   int x;
   int y;
   int speedX;
   int speedY;
   void init(){
      x = (int)random(width);
      y = (int)random(height);
      speedX = (int)random(5);
      speedY = (int)random(5);
   }
   void display(){
      ellipse( x, y, 30, 30 );
   }
```

```
   void move(){
      x = x + speedX;
      y = y + speedY;
      if ( x+15 > width ) {
         x = width - 15;
         speedX = -speedX;
      }
      if( x - 15 < 0 ){
         x = 15;
         speedX = -speedX;
      }
      if( y + 15 > height ){
         y = height - 15;
         speedY = -speedY;
      }
      if( y - 15 < 0 ){
          y = 15;
          speedY = -speedY;
      }
   }
}
```

インスタンスメソッドは・・・

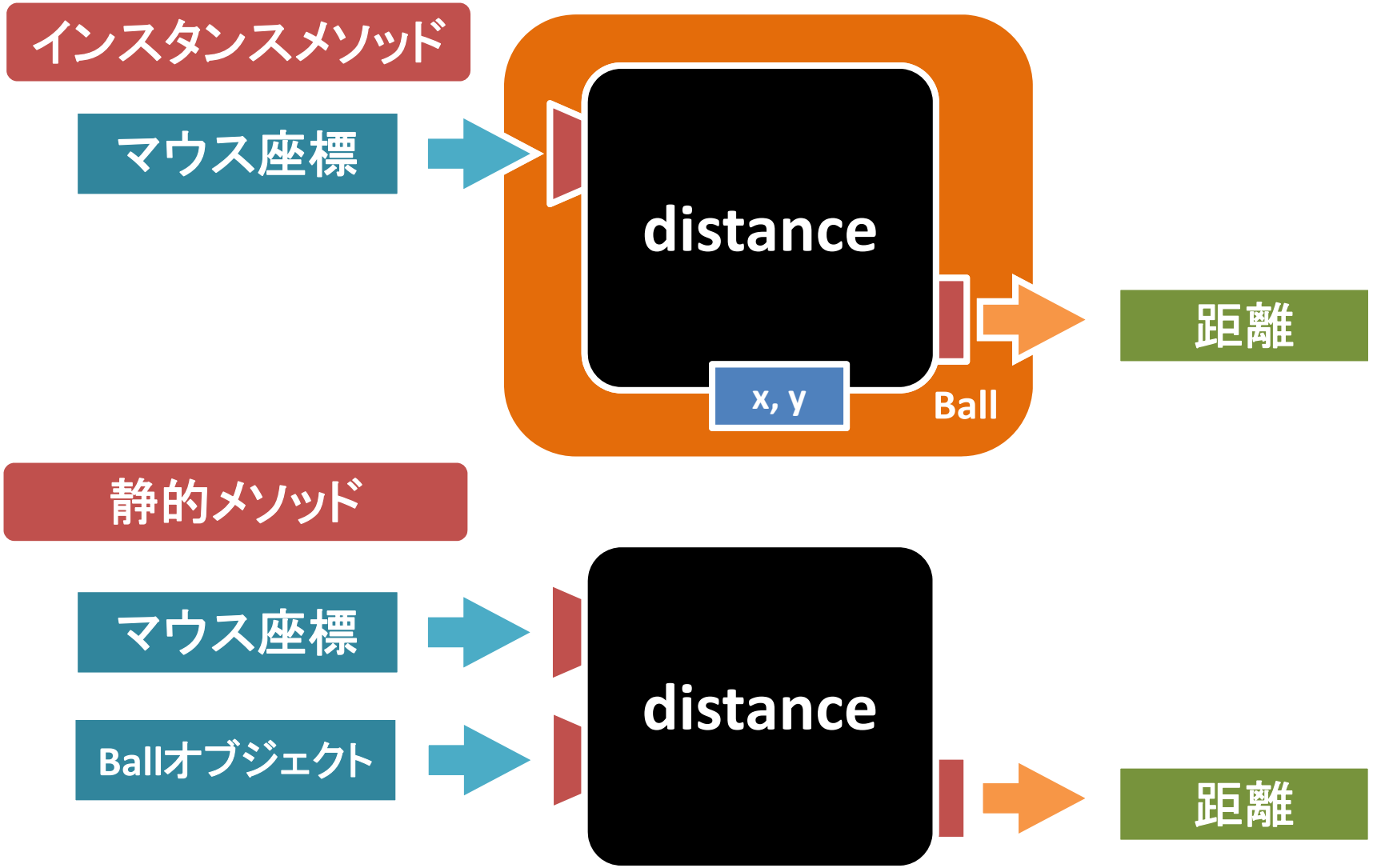
明治大学総合数理学部
先端メディアサイエンス学科
中村研究室

- クラスの内部の変数を自由に利用できる！
  - インスタンスメソッドはクラスの挙動自体を書き，内部の変数を外から隠しつつ利用するものも多く，引数がないことは珍しくない
    - move()「お任せで移動して！」
    - init()「お任せで初期位置と速度を決めて！」
  - 結果についても，内部変数の値を変更することに利用されることが多く，返り値が無いこともある
    - move()「移動した後の座標に更新しとくよ！」
    - init()「初期座標と移動速度適当に設定しとくね！」

距離を計算する

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

- マウスカーソルからBallまでの距離を計算する



距離を計算する

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

インスタンスメソッド

```
class Ball{
   int x;
   int y;
   int speedX;
   int speedY;
   int distance( int mx, int my ){
      int dd = (mx-x)*(mx-x)+(my-y)*(my-y);
      return sqrt(dd);
   }
```

```
if( komatsu.distance( mouseX, mouseY ) < 10 ){
    println( "Hit! komatsu-san" );
}
```

静的メソッド

```
int distance( int mx, int my, Ball obj ){
   int dd = (mx-obj.x)*(mx-obj.x)+(my-obj.y)*(my-obj.y);
   return sqrt(dd);
}
```

```
if( distance( mouseX, mouseY, komatsu ) < 10 ){
    println( "Hit! komatsu-san" );
}
```

距離を計算する

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

インスタンスメソッド

```
class Ball{
    int x;
    int y;
    int speedX;
    int speedY;
    int distance( int mx, int my ){
        int dd = (mx-x)*(mx-x)+(my-y)*(my-y);
        retur
    }
```

```
if( komatsu.distance( mouseX, mouseY ) < 10 ){
    println( "Hit! komatsu-san" );
}
```

ぶっちゃけ，どっちでもいい
使いやすい方を使うべし！

静的メソッド

```
int distance( int mx, int my, Ball obj ){
    int dd = (mx-obj.x)*(mx-obj.x)+(my-obj.y)*(my-obj.y);
    return sqrt(dd);
}
```

```
if( distance( mouseX, mouseY, komatsu ) < 10 ){
    println( "Hit! komatsu-san" );
}
```

ということで

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

- とりあえずメソッドについてまとめ終わり
- またそのうち補足するかもしれません