

Java

for文(ループ処理) 応用

13 時間目

for文でも、if文同様に
ネスト（入れ子）を作ることが可能です。

ネスト（＝入れ子）の書き方

```
for(初期値; 条件式; 増減式) {  
    if(条件) {  
        if文の処理内容;  
    }  
    for文の処理内容;  
}
```

for文と剰余のネスト

例) ある数(i)が、初期値1から10以下の間で、
ある数(i)が 偶数の場合は、ある数(i)を表示して改行(¥n)する。

Javaファイル

```
for(int i=1; i<=10; i++) {  
    if(i % 2 == 0) {  
        System.out.println( i + "¥n");  
    }  
}
```

2

4

6

8

10

if文の条件に該当する(i)を表示

for文と剰余のネスト 解説

外側のfor文 1回目

① 外側のfor文の1を走らせる。1は10以下 (=true) なので、内側 (=ネスト) にあるif文に進む。

if文

② 内側のif文で、 $i(=1)$ は2で割り切れない (=false) ので、外側のfor文に戻る。

外側のfor文 2回目

③ 外側のfor文の2を走らせる。2は10以下 (=true) なので、内側 (=ネスト) にあるif文に進む。

if文

④ 内側のif文で、 $i(=2)$ は2で割り切れる (=true) ので、`System.out.println` で、2を表示し、外側のfor文に戻る。

外側のfor文 3回目

⑤ 外側のfor文の3を走らせる。3は10以下 (=true) なので、内側 (=ネスト) にあるif文に進む。

if文

⑥ 内側のif文で、 $i(=3)$ は2で割り切れない (=false) ので、外側のfor文に戻る。

⑦ 同様に処理を続け、外側のfor文で11を走らせ、falseになった時点でfor文が終了する。

ネスト (=入れ子) の書き方

```
for(初期値; 条件式; 増減式){  
    for(初期値; 条件式; 増減式){  
        処理を実行する  
    }  
    処理を実行する  
}
```

例) 変数bが 初期値1から4未満の間(1, 2, 3の計3回)、変数bを表示する。
変数aが 初期値0から3未満の間、上記のfor文と改行(¥n)を繰り返す。

Javaファイル

```
for(int a=0; a<3; a++){  
    for(int b=1; b<4; b++){  
        System.out.println(b);  
    }  
    System.out.println("¥n");  
}
```

```
1  
2  
3  
  
1  
2  
3  
  
1  
2  
3
```

外側のfor文 a 0回目

- ① 外側のfor文の0を走らせる。0は3以下 (=true) なので、内側 (=ネスト) にあるif文に進む。

処理

内側のfor文
b 1~4回目

- ② 内側のfor文で、1を走らせる。1は4未満 (=true)なので、System.out.printlnで【 1 】が表示される。
- ③ 内側のfor文で、2を走らせる。2は4未満 (=true)なので、System.out.printlnで【 2 】が表示される。
- ④ 内側のfor文で、3を走らせる。3は4未満 (=true)なので、System.out.printlnで【 3 】が表示される。
- ⑤ 内側のfor文で、4を走らせる。4は4未満じゃない (=false)ので、外側のfor文に戻る。

外側のfor文 a 1回目

- ① 外側のfor文の1を走らせる。0は3以下 (=true) なので、内側 (=ネスト) にあるif文に進む。

処理

内側のfor文
b 1~4回目

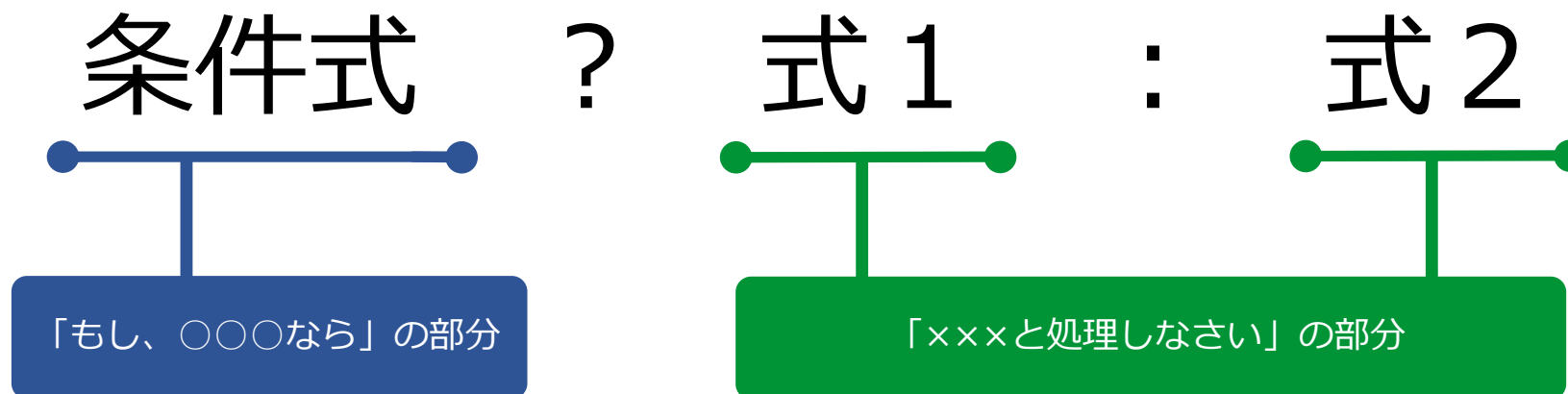
- ⑥ 内側のfor文で、1を走らせる。1は4未満 (=true)なので、System.out.printlnで【 1 】が表示される。
- ⑦ 内側のfor文で、2を走らせる。2は4未満 (=true)なので、System.out.printlnで【 2 】が表示される。

⑧ 同様に処理を続け、外側のfor文で4を走らせ、falseになった時点でfor文が終了する。

条件演算子(三項演算子)

解説

条件演算子は、条件によって処理内容を分けることができる演算子です。三項演算子とも呼ばれます。



「**条件式**が正しい(true)の場合、**式 1**を実行する。
間違い(false)の場合、**式 2**を実行する。」

という命令をする演算子

例) ある数(a)が0以上だった場合プラス、0未満だった場合マイナス、と表示する。

Javaファイル

```
int a =10;  
  
String b = a >= 0 ? "プラス" : "マイナス";  
  
System.out.println(b);
```

プラス

例) ある数(a)が偶数か奇数かを判別して表示する。

Javaファイル

```
int a = 2;  
  
String b = a % 2 == 0 ? "偶数" : "奇数";  
  
System.out.println(b);
```

偶数

例) ある数(i)が偶数の場合「☆」、奇数の場合「★」を表示する。
これをある数(i)が0以上5以下の間、繰り返す。

Javaファイル

```
for(int i=0; i<=5; i++) {  
    String b = i % 2 == 0 ? "☆" : "★";  
    System.out.println(b);  
}
```

☆ ★ ☆ ★ ☆ ★