

# ji-5. Javaプログラミングにおける繰り返し処理と数値計算の基礎

(Java プログラミング入門)

URL: <https://www.kkaneko.jp/pro/ji/index.html>



金子邦彦



# 内容



- 例題 1 . 自然数の和
- 例題 2 . 最大公約数の計算
- 例題 3 . ベクトルの長さ  
while 文
- 例題 4 . 九九の表  
for 文と繰り返しの入れ子
- 例題 5 . ド・モアブルの公式  
計算誤差の累積

- **繰り返し**（while 文, for 文）を使って，**繰り返し計算**を行えるようになること
- **ループカウンタ**として，**整数の変数**を使うこと
- 見やすいプログラムを書くために，**字下げ**を行う

# 繰り返しとは



- **繰り返し**とは、ある条件が満たされるまで、同じことを繰り返すこと。
- **繰り返し**を行うための文として**while文**, **for 文**などがある。

# 繰り返しの例



- ユークリッドの互助法
  - $m$  と  $n$  の最大公約数を求めるために、「割った余りを求めること」を、余りが 0 になるまで**繰り返す**。
- 九九の表
  - 九九の表を求めるために、**掛け算**を 8 1 回**繰り返す**

- プログラミングを行えるオンラインのサービス

<https://www.onlinegdb.com>

- ウェブブラウザを使う

- たくさんの言語を扱うことができる

Python3, Java, C/C++, C#, JavaScript,  
R, アセンブリ言語, SQL など

- オンラインなので、「秘密にしたいプログラム」  
を扱うには十分な注意が必要

# Online GDB で Java を動かす手順



① ウェブブラウザを起動する

② 次の URL を開く

<https://www.onlinegdb.com>

A screenshot of a web browser's address bar. It features a magnifying glass icon on the left, followed by the text "https://www.onlinegdb.com". The address bar is set against a light gray background.

### ③ 「Language」 のところで, 「Java」 を選ぶ

SPONSOR Slack — Bring your team together with Slack, the collaboration hub for work.

Run Debug Stop Share Save { } Beautify

Language -- select --

source code

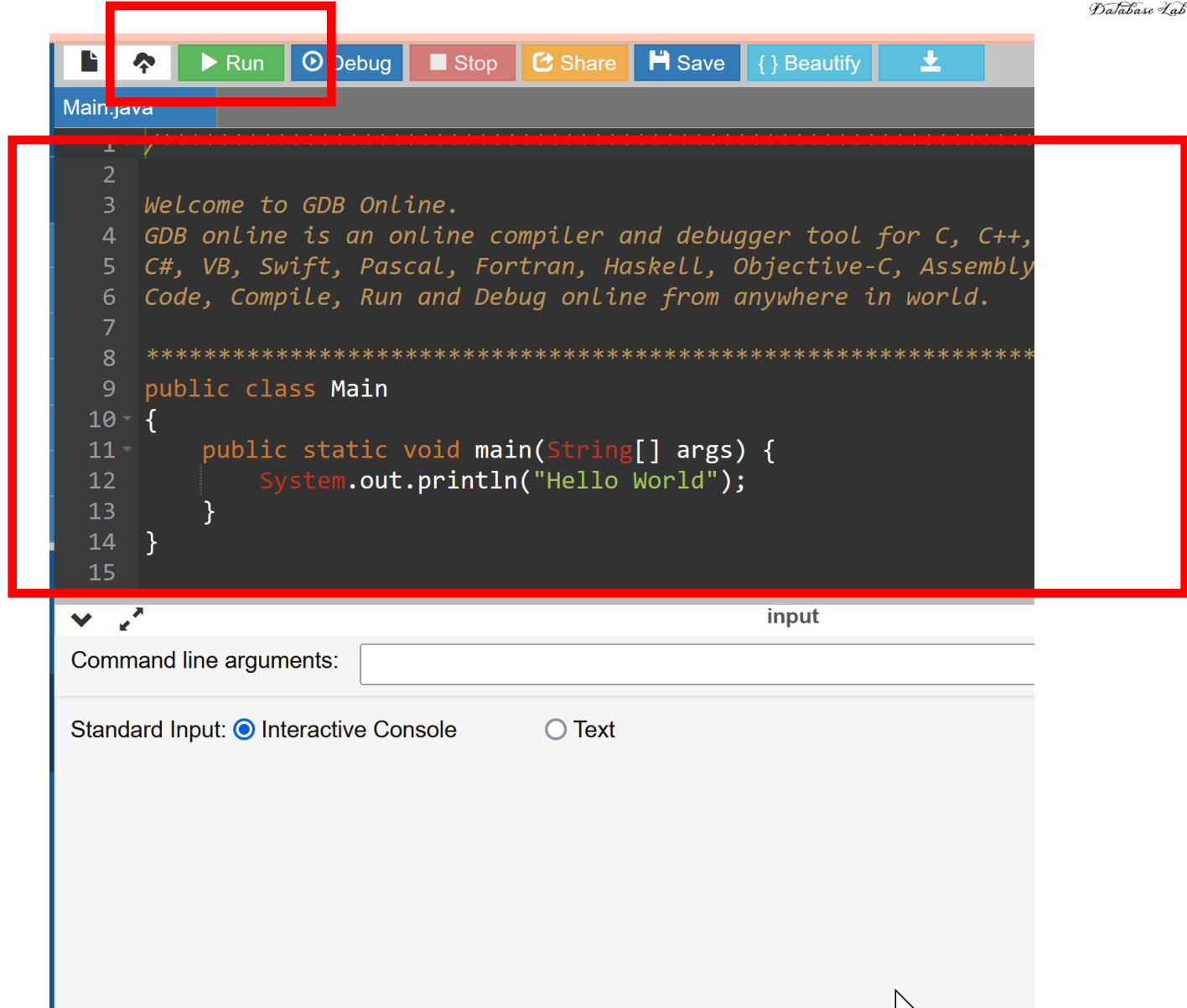
```
1 /*****
2
3 Welcome to GDB Online.
4 GDB online is an online compiler and debugger tool for C, C++, Python
5 C#, VB, Perl, Swift, Prolog, Javascript, Pascal, HTML, CSS, JS
6 Code, Compile, Run and Debug online from anywhere in world.
7
8 *****/
9 #include <stdio.h>
10
11 int main()
12 {
13     printf("Hello World");
14
15     return 0;
16 }
17
```

-- select --

- C
- C++
- C++ 14
- C++ 17
- Java**
- Python 3
- PHP
- C#
- VB
- HTML,JS,CSS
- Ruby
- Perl
- Pascal
- R
- Fortran
- Haskell
- Assembly(GCC)
- Objective C
- SQLite



## 実行ボタン



Main.java

```
1  ✓
2
3  Welcome to GDB Online.
4  GDB online is an online compiler and debugger tool for C, C++,
5  C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly
6  Code, Compile, Run and Debug online from anywhere in world.
7
8  *****
9  public class Main
10 {
11     public static void main(String[] args) {
12         System.out.println("Hello World");
13     }
14 }
15
```

input

Command line arguments:

Standard Input: ☒ Interactive Console ☐ Text

## エディタ画面

プログラムを  
書き換えること  
ができる

# 例題 1 . 自然数の和



- 整数データ（**N**とする）を読み込んで，**1 から N までの和を求めるプログラムを作る**
- ここでは，練習のため，自然数の和の公式は使わずに，**while文を用いる**

例) 100 → 5050

```
import java.util.Scanner;
public class Main
{
    public static void main(String[] args) {
        int N, sum, i;
        Scanner s = new Scanner(System.in);
        System.out.println("Please Enter N =");
        N = s.nextInt();
        sum = 0;
        i = 1;
        while(i <= N) {
            sum = sum + i;
            i = i + 1;
        }
        System.out.printf("goukei = %d¥n", sum);
    }
}
```

**条件式**

**繰り返し実行される  
部分**

## 実行結果の例

```
Please Enter N =  
100  
goukei = 5050
```

```
Please Enter N =  
200  
goukei = 20100
```

# 繰り返し処理の中身



- 繰り返しの前

$i = 1$  と  $sum = 0$  を実行

- 繰り返しの各ステップでなされること

1. 「sum」に「i」を足しこむ

→ 「sum」には、その時点での「1からi」までの和が入る

2. 「i」の値を1増やす

- 繰り返しの終了条件

$i \leq N$  が成り立たなくなったら終了

→ つまり  $i > N$  になったら終了

# 自然数の和



## N = 7 とすると

sum の値

i の値

はじめは  $\text{sum} = 0$

$i = 1$

繰り返し 1 回目	$i \leq 7$ が成立する	$\text{sum} = 0 + 1$	$i = 2$
繰り返し 2 回目	$i \leq 7$ が成立する	$\text{sum} = 1 + 2$	$i = 3$
繰り返し 3 回目	$i \leq 7$ が成立する	$\text{sum} = 3 + 3$	$i = 4$
繰り返し 4 回目	$i \leq 7$ が成立する	$\text{sum} = 6 + 4$	$i = 5$
繰り返し 5 回目	$i \leq 7$ が成立する	$\text{sum} = 10 + 5$	$i = 6$
繰り返し 6 回目	$i \leq 7$ が成立する	$\text{sum} = 15 + 6$	$i = 7$
繰り返し 7 回目	$i \leq 7$ が成立する	$\text{sum} = 21 + 7$	$i = 8$
繰り返し 8 回目	$i \leq 7$ が成立しない		

# while 文



- 何かの処理の**繰り返し**
- **繰り返し**のたびに while 文で書かれた条件式の真偽が判定され, 真である限り, while のあとに続く文が**実行され続ける**.

```
while(i <= N) {  
    sum = sum + i;  
    i = i + 1;  
}
```

## 例題 2 . 最大公約数の計算



- 2つの整数データを読み込んで、**最大公約数を求めるプログラムを作る.**
  - ユークリッドの互助法を用いること
  - ユークリッドの互助法を行うために while 文を使い、**余りを求めることを繰り返す.**

例) 20, 12 のとき : 4



# ユークリッドの互助法



- **最大公約数を求めるための手続き**
- $m, n$  の最大公約数は,
  - $m \geq n$  とすると,
  - 「 $m$  を  $n$  で割った余り」 = 0 なら, 最大公約数は  $n$
  - 「 $m$  を  $n$  で割った余り」 > 0 なら,  $m$  と  $n$  の最大公約数は, 「 $m$  を  $n$  で割った余り」と  $n$  の最大公約数に等しい
- (なお,  $n > \text{「}m \text{ を } n \text{ で割った余り」}$  が成り立つ)

```
import java.util.Scanner;
public class Main
{
    public static void main(String[] args) {
        int m, n, r;
        Scanner s = new Scanner(System.in);
        System.out.println("Please Enter m =");
        m = s.nextInt();
        System.out.println("Please Enter n =");
        n = s.nextInt();
        r = m % n;
        while(r > 0) {
            m = n;
            n = r;
            r = m % n;
        }
        System.out.printf("GCD = %d¥n", n);
    }
}
```

## 条件式

条件が成り立つ限り,  
実行されつづける部分

# 最大公約数の計算



## 実行結果の例

```
Please Enter m =  
12  
Please Enter n =  
8  
GCD = 4
```

```
Please Enter m =  
80  
Please Enter n =  
35  
GCD = 5
```

# 繰り返し処理の中身



- **繰り返しの前**

$r = m \% n$  を実行 ( $m$  を  $n$  で割った余りが  $r$  に入る)

- **繰り返しの各ステップ**でなされること

$m = n;$

$n = r;$

$r = m \% n;$

を実行 ( $m, n, r$  の値は小さくなっていく)

- **繰り返しの終了条件**

$r$  が 0 になったら終了

# 最大公約数の計算



$m = 80$ ,  $n = 35$  とすると,  
最初の「 $r = m \% n$ 」で,  $r = 10$  になる

	m の値	n の値	r の値
繰り返し 1 回目	$r > 0$ が成立する $m = 35$ $n = 10$ $r = 5$		

80, 35 の最大公約数は  
35, 10 の最大公約数に等しい

繰り返し 2 回目	$r > 0$ が成立する $m = 10$ $n = 5$ $r = 0$		
--------------	--	--	--

35, 10 の最大公約数は  
10, 5 の最大公約数に等しい

繰り返し 3 回目	$r > 0$ が成立しない		
--------------	----------------	--	--

## 例題 3 . 総和と平均



- データ  $x_1, x_2, \dots, x_k$  を 1 つずつ読み込んで, **合計と平均を求めるプログラム**を作成す
- **負の数が入力されたら終了する**

整数のデータ 1, 2, 3 に対して

1

2

3

-1

入力

```
import java.util.Scanner;
public class Main
{
    public static void main(String[] args) {
        int sum, i, x;
        sum = 0;
        i = 0;
        Scanner s = new Scanner(System.in);
        System.out.printf("Please Enter x[%d] =", i);
        x = s.nextInt();
        while(0 <= x) {
            sum = sum + x;
            i = i + 1;
            System.out.printf("Please Enter x[%d] =", i);
            x = s.nextInt();
        }
        System.out.printf("sum = %d¥n", sum);
        System.out.printf("%d / %d = %d", sum, i, sum / i);
    }
}
```

## 条件式

条件が成り立つ限り、  
実行されつづける部分

## 実行結果の例

```
Please Enter x[0] =1
Please Enter x[1] =2
Please Enter x[2] =3
Please Enter x[3] =-1
sum = 6
6 / 3 = 2
```

```
Please Enter x[0] =3
Please Enter x[1] =5
Please Enter x[2] =4
Please Enter x[3] =2
Please Enter x[4] =-1
sum = 14
14 / 4 = 3
```



# 繰り返し処理の中身



- **繰り返しの前**

**sum = 0 と i = 0 を実行**

**X0 を読み込む**

- **繰り返しの各ステップでなされること**

1. 「sum」に「Xi」を足しこむ

→ 「sum」には、その時点での「X0から Xi」までの和が入る

2. 「i」の値を1増やす

3. Xiを読み込む

- **繰り返しの終了条件**

**$X_i < 0$  ならば終了**

## 例題 4 . 九九の表



- 九九の表を表示するプログラムを作成する
  - 九九の表を表示するために、繰り返しの入れ子を使う

```
public class Main
{
    public static void main(String[] args) {
        int i, j;
        for(i = 1; i <= 9; i++) {
            System.out.printf("%3d:", i);
            for(j = 1; j <= 9; j++) {
                System.out.printf("%3d", i * j);
            }
            System.out.println("");
        }
    }
}
```

繰り返し  
実行される部分

# 実行結果画面（例）



1:	1	2	3	4	5	6	7	8	9
2:	2	4	6	8	10	12	14	16	18
3:	3	6	9	12	15	18	21	24	27
4:	4	8	12	16	20	24	28	32	36
5:	5	10	15	20	25	30	35	40	45
6:	6	12	18	24	30	36	42	48	54
7:	7	14	21	28	35	42	49	56	63
8:	8	16	24	32	40	48	56	64	72
9:	9	18	27	36	45	54	63	72	81

# for 文



```
for(i = 1; i <= 9; i++) {  
    System.out.printf("%3d:", i);  
    for(j = 1; j <= 9; j++) {  
        System.out.printf("%3d", i * j);  
    }  
    System.out.println("");  
}
```

## 例題 5 . ド・モアブルの定理



- $\theta$ を読み込んで、次の値を計算するプログラムを作る

$$(\cos \theta + i \sin \theta)^n$$

$$\cos n\theta + i \sin n\theta$$

- なお、 $i$ は虚数単位
- ここでは  $(\sin\theta + i \cos\theta)^n$  を求めるために、while文を用いた繰り返し計算を行ってみる

# 複素数の積



$$z_1 = x_1 + i y_1$$

$$z_2 = x_2 + i y_2 \text{ のとき}$$

$$\begin{aligned} z_1 z_2 &= (x_1 + i y_1) (x_2 + i y_2) \\ &= x_1 x_2 + x_1 i y_2 + i y_1 x_2 + i y_1 i y_2 \\ &= \underbrace{x_1 x_2 - y_1 y_2}_{\text{実数部}} + i \underbrace{(x_1 y_2 + y_1 x_2)}_{\text{虚数部}} \end{aligned}$$

```
import java.lang.Math;
import java.util.Scanner;
public class Main
{
    public static void main(String[] args) {
        int n, j;
        double x1, y1, x2, y2, theta;
        Scanner s = new Scanner(System.in);
        System.out.println("Please Enter n =");
        n = s.nextInt();
        System.out.println("Please Enter theta =");
        theta = s.nextFloat();
        x1 = Math.cos(theta);
        y1 = Math.sin(theta);
        x2 = x1;
        y2 = y1;
        j = 1;
        while(j <= n) {
            System.out.printf("(cos theta + i sin theta)%d = %8.3f + i%8.3f\n", j, x1, y1);
            System.out.printf("cos %d theta + i sin %d theta = %8.3f + i %8.3f\n", j, j,
                Math.cos(j * theta), Math.sin(j * theta));
            x1 = x1 * x2 - y1 * y2;
            y1 = x1 * y2 + x2 * y1;
            j = j + 1;
        }
    }
}
```

条件式

繰り返し実行される  
部分



# 実行結果



```
Please Enter n =
3
Please Enter theta =
0.1
(cos theta + i sin theta)1 =      0.995 + i      0.100
cos 1 theta + i sin 1 theta =      0.995 + i      0.100
(cos theta + i sin theta)2 =      0.980 + i      0.197
cos 2 theta + i sin 2 theta =      0.980 + i      0.199
(cos theta + i sin theta)3 =      0.955 + i      0.292
cos 3 theta + i sin 3 theta =      0.955 + i      0.296
(cos theta + i sin theta)4 =      0.922 + i      0.382
cos 4 theta + i sin 4 theta =      0.921 + i      0.389
(cos theta + i sin theta)5 =      0.879 + i      0.468
cos 5 theta + i sin 5 theta =      0.878 + i      0.479
(cos theta + i sin theta)6 =      0.828 + i      0.548
cos 6 theta + i sin 6 theta =      0.825 + i      0.565
(cos theta + i sin theta)7 =      0.769 + i      0.622
cos 7 theta + i sin 7 theta =      0.765 + i      0.644
(cos theta + i sin theta)8 =      0.703 + i      0.689
cos 8 theta + i sin 8 theta =      0.697 + i      0.717
```

# 繰り返し処理の中身



- 繰り返しの前

```
x1 = Math.cos(theta);  
y1 = Math.sin(theta);  
x2 = x1;  
y2 = y1;
```

を実行

- 繰り返しの各ステップでなされること

```
x1 = x1 * x2 - y1 * y2;  
y1 = x1 * y2 + x2 * y1;
```

を実行 (x1 が**実数部**, y1 が**虚数部**)

$$(\cos\theta + i \sin\theta)^n = \cos n\theta + i \sin n\theta$$

$$\begin{aligned}(\cos\theta + i \sin\theta)^2 &= \cos^2\theta - \sin^2\theta + 2i \cos\theta \sin\theta \\ &= \cos 2\theta + i \sin 2\theta\end{aligned}$$

$$\begin{aligned}(\cos\theta + i \sin\theta)^3 &= (\cos\theta + i \sin\theta)^2 (\cos\theta + i \sin\theta) \\ &= (\cos 2\theta + i \sin 2\theta) (\cos\theta + i \sin\theta) \\ &= \cos 2\theta \cos\theta - \sin 2\theta \sin\theta \\ &\quad + i (\cos 2\theta \sin\theta + \sin 2\theta \cos\theta) \\ &= \cos (2\theta + \theta) + i \sin (2\theta + \theta) \\ &= \cos 3\theta + i \sin 3\theta\end{aligned}$$

(以下同様に考える. 数学的帰納法で証明できる)

# 計算結果から分かること



- 本来なら「 $(\cos\theta + i \sin\theta)^n = \cos n\theta + i \sin n\theta$ 」が成り立つはず
- しかし、コンピュータでの計算は、近似計算
- **計算を繰り返す**（つまり、計算結果を使った計算）  
**たびに、誤差が積み重なる**

# 演習. $m$ から $n$ までの和



- 2つの整数データ ( $M$ ,  $N$ とする) を読み込んで,  
 $M$ から $N$ までの和を求めるプログラムを作りなさい