

# プログラミングⅡ

黒瀬 浩

[kurose@neptune.kanazawa-it.ac.jp](mailto:kurose@neptune.kanazawa-it.ac.jp)

OH: 講義の前後, Eメール問合せ, 月3限21-405

居室 67・121

# VSCodeのCode Runnerで実行時に入力を行う方法

テキストエディタVSCodeを使用していない人は関係ありません  
機能拡張Code Runnerを入れると左上の実行ボタンやCtrl-Alt-Nで  
そのファイル拡張子に応じた処理をしてくれる

Javaの場合はコンパイルと実行

```
cd 保管場所 && javac クラス名.java && java クラス名
```

&&は前のコマンドでエラーがなければ次を行う

通常は "出力" に表示結果が出るが、キーボード入力できない

設定 ⇨ 上部の検索窓に `code-runner terminal` を入力

Code-runner:Run In Terminal のチェックボックスをチェックする

実行すると "出力" ではなく "ターミナル" で動作し

キーボード入力を受け付けるようになる

一度実行してから、ターミナルで↑を押すとコンパイル実行コマンドが  
呼び出せるので、実行ボタンを押すより楽

エディタでソースを修正したらファイルを保存(ctrl-s)すること 2

# 配列(dimension, array) 教科書3-4参照

同じ型の値を連続的に管理する変数

型[] 変数名;

変数名 = new 型[要素数];

型[] 変数名 = new 型[要素数]; 上の2行をまとめた形

参照、代入するときは 変数名[添字] を使う

添字は0からの連続

初期化する場合

型[] 変数名 = {値1, 値2, ...} 要素数は値の並びによる

2次元配列

型[][] 変数名 = new 型[要素数][要素数];

要素数を得る 変数名.length

サンプルプログラムの man(String[] args) の意味は既にわかるはず

# 1次元配列の例

pythonのリストと異なり固定的

内容の型は全て同じ

生成後、要素を追加したり削除したりできない

// 1から10を2倍した数列作成

```
public class Ex4 {  
    public static void main(String[] args){  
        int[] a = new int[10];           // 整数で10要素の配列生成  
        for(int i=0; i<a.length; i++){   // 配列添字は0から始まる  
            a[i]=(i+1)*2;                 // 1からなので1足している  
        }  
        for(int i=0; i<a.length; i++){  
            System.out.printf("%3d ", a[i]);    // ここではしていない  
        }  
        System.out.println("");             // ここで出力  
    }  
}
```

実行結果

2    4    6    8    10    12    14    16    18    20

for文の中で変数iを宣言しているので  
for文を抜けると変数iはなくなる  
配列aの前後でiを宣言すればfor文の  
中で宣言する必要はない

# コマンド引数から値を読む

実行時 java プログラム名 引数1 引数2 ... ↵

```
public static void main(String[] args){  
    System.out.println( args.length );    // 引数の数  
    for(int i=0; i<args.length; i++){  
        int j = Integer.parseInt( args[i] );  
        System.out.println( j );  
    }  
}
```

コマンド引数は main()の引数で指定した変数に入る(上記ではargs)  
何個指定されたかは .length でわかる

文字列で入るので、整数化したければ Integer.parseInt() を使う

pythonと異なり、引数にプログラム名は入らない(pythonでは0要素目はプログラム名)

# レポート5 素数判定

コマンド引数から正数を得る(99を入力することにする)  
正数1から与えられた正数までが、素数ならば各3桁で表示  
出力例

2    3    5    ...

1から正数までの繰り返しは、for文を使うこと  
約数の数を返すメソッドdivisors() を実装すること  
1からその数まで割って余りが0ならカウントアップする  
divisors()の結果が2ならば数値を表示する

参考

書式指定の表示(pythonの文字列の%演算と同様) System.out.printf()

# レポート6 2次元配列を使った掛け算九九の表

掛け算九九の表を表示する

1	2	3	4	5	6	7	8	9
2	4	6	8	10	12	14	16	18

....

2次元の配列を使うこと

値の計算 2重のforループを使い、一旦配列に値を格納すること

値の表示 2重のforループを使い、各3桁で数値を表示すること

チャレンジ

表のタイトル, 見出しをつける

1	2	3
---	---	---

+---+---+---+ ...

1	1	2	4
---	---	---	---

参考

改行しない表示 `System.out.print()`

書式指定の表示(pythonの文字列の%演算と同様) `System.out.printf()`

# レポート5, レポート6の提出

レポートタイトル, 提出日, クラス名列, 氏名を上部に記載  
A4で1枚(両面可) レポート5,6は別の紙にすること  
javaソースコードと実行画面のハードコピーを貼ること

期限 第14回開始時

全てのレポートの最終提出期限は第14回開始時



# クイズ（次回予定）の学習

javaの型と定数の例を書け

正数

実数

真理値（真偽値）

文字

文字列

配列の生成の文を書け

正数 10要素

文字列 5要素

正数 3要素×4要素

配列の要素数（長さ）の取得方法は

1から10までfor文を用いて総和を求めるプログラムをかけ

キーボードから入力するのに必要な2文を書け

正数2つをもらい合計を返すメソッドを書け

文字列と正数をもらい文字列を正数回繰り返した文字列を返すメソッドをかけ

# クイズ（次回予定）の学習

10要素の1次元配列に数値が入っている（値は不明）

負の値が出るか（なければ要素の終わりまで）を加算する処理を  
for文, while文, do while文の3種類でかけ

メソッドのオーバーロードができない条件を書け

ソースのリストからファイル名を特定するにはどこをみたらよいか

ソースファイルからクラスファイルを生成する方法を書け

クラスファイルから実行する方法を書け

キーボードから入力した文字列を実数にする方法を書け

コマンド引数で入力した文字列を正数にする方法を書け

メソッドが返す型はどこをみたらわかるか

配列に格納する値の型はどこをみたらわかるか