

プログラミング演習2 課題 補足資料

第2週：データ型

練習課題2-1

ラッパークラス

- データ構造とアルゴリズムの第1週講義資料 pp. 46-47、教科書 p. 35以降を見て
 - プリミティブ型(数値型)
 - ラッパークラス
 - 数値型とラッパークラスの変換

について、確認すること

練習課題2-2

オブジェクトの生成とコピー

- 第1週講義資料 pp. 51-53を見ながら、動作を確認してみる

練習課題2-3,2-4,2-5,2-6

PositionクラスとRobotクラス

- 練習課題2-3

- ちゃんと入力できたか、動作を確認できればよい

- 練習課題2-4

- Robot robitaMkII = robita ; の行で、
何が行われるかを教科書pp. 46-49, Fig. 3.3(1)を見ながら確認

- 練習課題2-5

- new Robot(this.position, this.name + “クローン”) ; で、
何が行われるかを教科書pp. 49-50, Fig. 3.3(2)を見ながら確認

- 練習課題2-6

- 教科書p.50, Fig. 3.3(3)を見ながら考えること

練習課題2-7, 必須課題2-8

Point2D/Line2Dクラスにcloneメソッドを追加

- 基本的な考え方は、練習課題2-6と同じ
- 第1週講義資料 pp. 55-59を見ながら考えること
 - 深いコピーを実現するには？
- プリミティブ型の変数と、クラス型の変数の動作の違いに注意
 - コピーにnewが必要なのは、クラス型の変数だけ

```
int a = 0;  
int b = 1;  
System.out.println(a + " , " + b);
```

// プリミティブ型のコピー

```
b = a;  
System.out.println(a + " , " + b);
```

// aの値を変えても、bは変わらない

```
a = 2;  
System.out.println(a + " , " + b);
```

int型の場合は
特に意識しなくても
以下の出力が得られる

```
0 , 1  
0 , 0  
2 , 0
```

練習課題2-7, 必須課題2-8

動作確認プログラム

- 以下のようなプログラムで動作確認をすること

```
Point2D p1 = new Point2D(10,10);  
Point2D p2 = p1.clone(); // p2をクローン
```

```
p2.setX(20); // p2のx座標を変更  
p1.print();  
p2.print();
```

```
Line2D l1 = new Line2D(p1,p2);  
Line2D l2 = l1.clone(); // l1をクローン
```

```
// l2の終点を(25,30)に変更  
l2.getP2().setX(25);  
l2.getP2().setY(30);  
l1.print();  
l2.print();
```

以下の出力が
得られればOK

```
(10.0, 10.0)  
(20.0, 10.0)  
(10.0, 10.0) - (20.0, 10.0)  
(10.0, 10.0) - (25.0, 30.0)
```

必須課題2-9

Line2Dクラスにequalsメソッドを追加

- 第1週講義資料 p. 57を見ながら考えること
- 以下のようなプログラムで、動作を確認する

```
Point2D p1 = new Point2D(10, 20) ;  
Point2D p2 = new Point2D(30, 40) ;  
Point2D p3 = new Point2D(50, 60) ;
```

```
Line2D l1 = new Line2D(p1, p2) ;  
Line2D l2 = new Line2D(p1, p2) ;  
Line2D l3 = new Line2D(p1, p3) ;
```

```
System.out.println("l1, l2 = " + l1.equals(l2)) ;  
System.out.println("l2, l1 = " + l2.equals(l1)) ;  
System.out.println("l1, l3 = " + l1.equals(l3)) ;
```

以下の出力が
得られればOK

```
l1, l2 = true  
l2, l1 = true  
l1, l3 = false
```