

2012年10月30日

ライフスタイルデザイン研究センター 金井秀明

1. 「長方形」クラス (Rectangle) のサブクラスとして、以下の機能を持つ PlacedRectangle を宣言してください。

(a) フィールド

- 長方形の中心座標(x, y)を示すdouble型のフィールドx, yを持つ。

(b) コンストラクタ

- 引数なし。ただし、初期座標を(x,y)=(0.0,0.0)とする。widthとheightの初期値は0.0とする。
- 中心座標(x, y)を指定する。
- 中心座標(x, y)および幅(width)と高さ(height)を指定する。

```
class Rectangle {
    protected double width;
    protected double height;

    Rectangle () {
        setSize(0.0, 0.0);
    }

    Rectangle(double w, double h) {
        setSize(w, h);
    }

    void setSize(double w, double h) {
        this.width = w;
        this.height = h;
    }
}
```

図. 長方形 (Rectangle) クラスの宣言

2. 以下のコードに関する項目について、正解のものを答えよ。もし正解が複数あるときには、それらをすべて書け。

- (a) フィールドaをクラスAの外からアクセスできない。
 (b) メソッドdをクラスAからアクセスできる。
 (c) フィールドcをクラスAAAからアクセスできない。
 (d) フィールドbをクラスAAからアクセスできる。

```
class A {
    public static int a = 0;
    private int b;
    protected int c;
    public A() {
        ...
    }
    public void d (int n ) {
        b = n;
    }
}

class AA extends A { }
class AAA extends AA { }
```

図. 問題2のコード

3. 次のようなdouble値の2次元座標を表すクラスPoint2Dを作成せよ.

(a)フィールド

- 座標(x, y)を示すprivateのメンバのdouble型のフィールド

(b)コンストラクタ (2つ)

- 引数なし // 初期座標を(x, y) =(0.0, 0.0)とする.
- 引数あり // x, y座標を指定する.

(c)メソッド (5つ)

- setX // x座標値を設定する.
- setY // y座標値を設定する.
- getX // x座標値を得る. 戻り値はdouble型.
- getY // y座標値を得る. 戻り値はdouble型
- GetDistance
原点(0.0, 0.0)から任意の点 (x, y)の距離を計算する.
ここで, 原点(0.0,0.0)から任意の点 (x, y)の距離は,
 $\text{Math.sqrt}(x*x+y*y)$ で求まる.

4. 抽象クラス車 (V e h i c l e) はナンバープレート番号を示すフィールド (car_license) と抽象メソッド給油する (feed) を持つ. 車クラスのサブクラスとして, トラック (track)と乗用車 (car) を作成せよ. またトラックと乗用車のオブジェクトを作成せよ.

5. 次のようにクラスA,B,Cが定義されています。

```
class A { }  
class B extends A { }  
class C
```

次の変数宣言と代入で誤りがあるものを答えよ。

- (1) A a = new A();
- (2) A a = new B();
- (3) A a = new C();
- (4) B b = new C();
- (5) B b = new B();
- (6) B b = new C();

6. 次の文章で誤っているものを選びなさい。

- (a) インタフェースの定義の中では、メソッドが行う処理の内容は定義しない。
- (b) 1つのクラスが実装できるインタフェースは1つまでである。
- (c) インタフェースを実装するクラスは、インタフェースに含まれるメソッドを定義してもしなくてもよい。

7. インターフェースの定義として、正しいものを選び、もし正解が複数あるときには、それらをすべて書け。

(a) `public interface Inter { void method(); }`

(b) `public interface Inter { void method() {}; }`

(c) `public interface Inter { Inter(); }`

(d) `public interface Inter { Inter(){}; }`