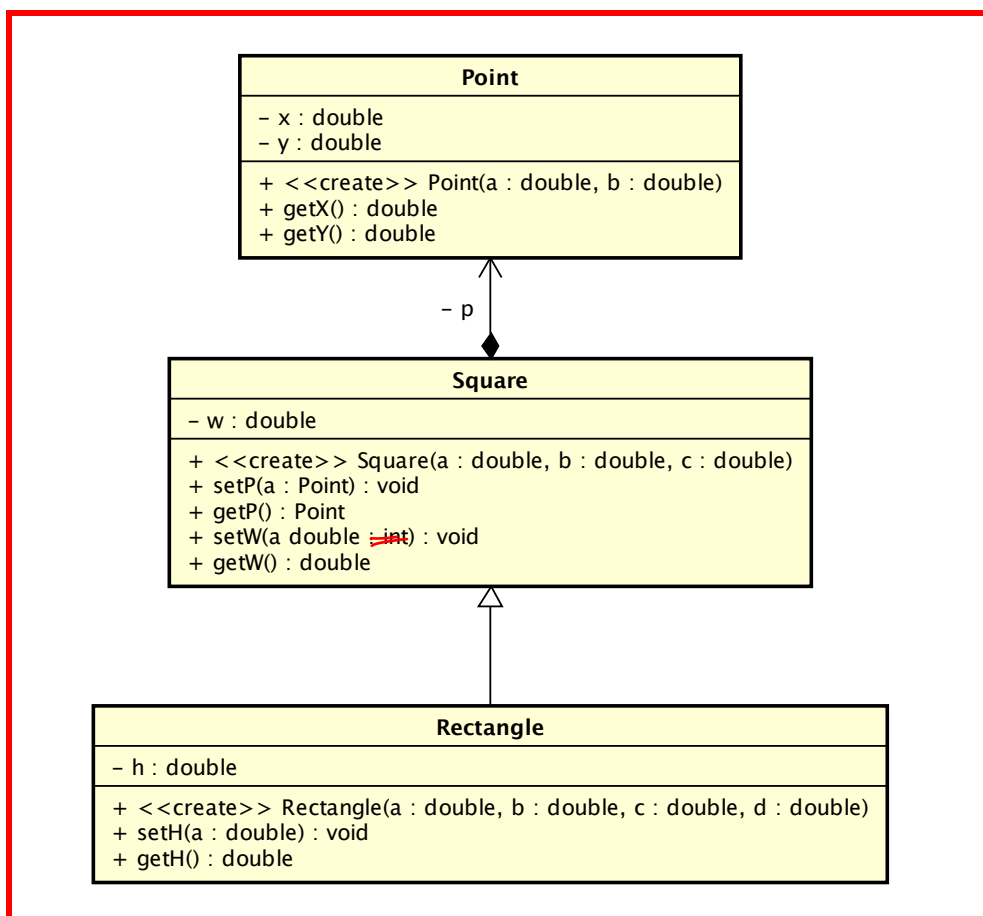


## 1 点クラスと長方形クラス

付録1のJavaプログラムをよく読んで答えなさい。

### 1.1 3つのクラスと関連をクラス図に示しなさい。(30点)

注意：コンストラクタは、型を省略した操作として書くものとします。  
(ステレオタイプを明示しても良いが、わからない人は省略して良い。)



オブジェクト指向プログラミング H30 年度 後期中間試験 (2018.12.17 重村 哲至)

IE5 \_\_\_\_ 番 氏名 \_\_\_\_ 模範解答

1.2 プログラムの正しい行に○, 誤っている行に × を書きなさい.  
(2 点 × 15 問 = 30 点)

```
Point p = new Point(10,20);  
double x = p.x; // (1) ____ X  
double y = p.getY(); // (2) ____ O  
p.y = 30; // (3) ____ X  
  
Square s = new Square(10, 20, 5);  
double a = s.p.x; // (4) ____ X  
double b = s.p.getY(); // (5) ____ X  
Point q = s.p; // (6) ____ X  
Point r = s.getP(); // (7) ____ O  
double c = s.getP().x; // (8) ____ X  
double d = s.getP().getY(); // (9) ____ O  
  
s.setP(p); // (10) ____ O  
s.getP().x = 40; // (11) ____ X  
double e = s.getP().x; (8) // (12) ____ X  
double f = s.getP().getY(); (9) // (13) ____ O  
double g = s.getH(); // (14) ____ X  
s.setH(40); // (15) ____ X
```

1.3 プログラムを完成しなさい。 (20 点)

下のプログラムの実行結果は次のようになります。同じ実行結果になるように下のプログラムの空欄（下線部）を補いなさい。

実行結果

```
$ java Ex1
(x,y)-(h,w)
(10.0,20.0)-( 4.0, 3.0)
```

```
class Ex1 {
    static public void main(String[] args) {
        Rectangle r = new Rectangle(10, 20, 3, 4);

        double x = r. getP().getX();
        double y = r. getP().getY();
        double h = r. getH();
        double w = r. getW();

        System.out.printf("(x,y)-(h,w)\n");
        System.out.printf("(%.1f,%.1f)-("%.1f,%.1f)\n",x,y,h,w);
    }
}
```

## 2 グレースケール画像クラス

付録2は授業で説明したグレースケール画像クラスを少し簡単化したものです。プログラム中 (A), (B), (C) に適切な記述を答えなさい。

(A) (10 点)

```
// グレースケールに変換して自身にコピー
for (int y=0; y<img.getHeight(); y++) {
    for (int x=0; x<img.getWidth(); x++) {
        int rgb = img.getRGB(x, y);    // 画素の ARGB を取り出す
        int gray = rgbToGray(rgb);    // グレースケールに変換
        setRGB(x, y, grayToRgb(gray)); // 変換結果を画素に書き込む
    }
}
```

(B) (5 点)

```
return getRGB(x, y) & 0xff;
        または
return rgbToGray(getRGB(x, y));
```

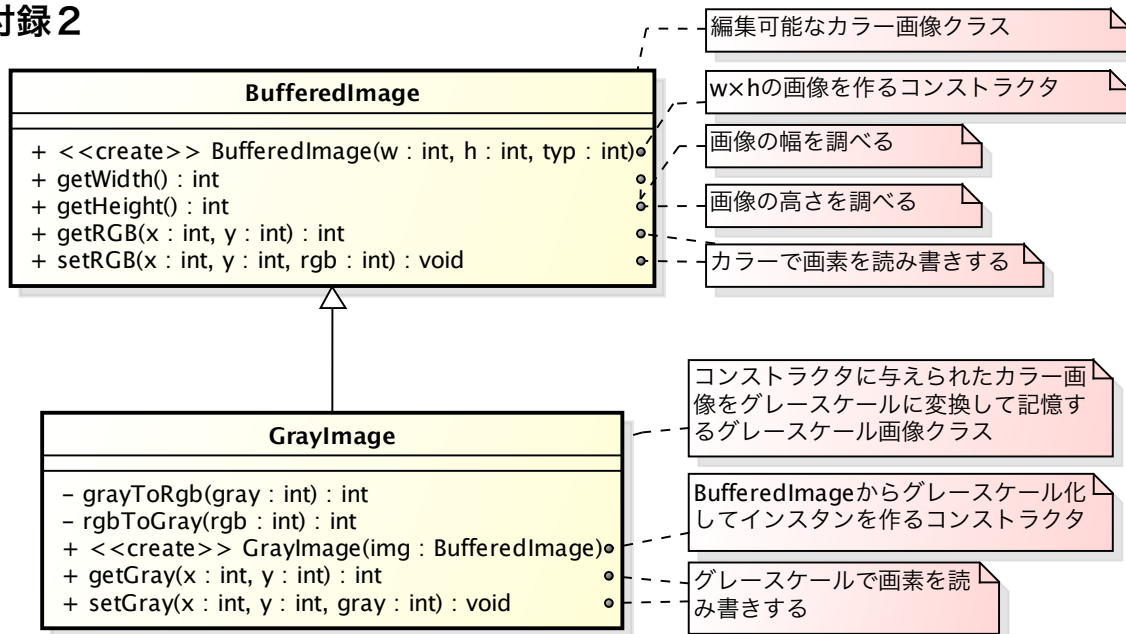
(C) (5 点)

```
setRGB(x, y, grayToRgb(gray));
```

## 付録 1

```
1 // 座標 (x,y) を表すクラス
2 class Point {
3     private double x;
4     private double y;
5     public Point(double a, double b) {
6         x = a;
7         y = b;
8     }
9     public double getX() { return x; } // 座標の変更はできない
10    public double getY() { return y; }
11 }
12
13 // 正方形を表すクラス
14 class Square {
15     private Point p; // 正方形の左上隅の座標
16     private double w; // 正方形の幅
17     public Square(double a, double b, double c) {
18         p = new Point(a, b);
19         w = c;
20     }
21     public void setP(Point a) { p = a; }
22     public Point getP() { return p; }
23     public void setW(double a) { w = a; }
24     public double getW() { return w; }
25 }
26
27 // 長方形を表すクラス
28 class Rectangle extends Square {
29     private double h; // 長方形の高さ
30     public Rectangle(double a, double b, double c, double d) {
31         super(a, b, c); // 親クラスのコンストラクタを呼ぶ
32         h = d;
33     }
34     public void setH(double a) { h = a; }
35     public double getH() { return h; }
36 }
```

## 付録2



```

1 import java.awt.image.BufferedImage;
2 public class GrayImage extends BufferedImage {
3     // 画素をグレースケールから ARGB に変換
4     private int grayToRgb(int gray) {
5         gray = gray & 0xff;
6         return (0xff<<24) | (gray<<16) | (gray<<8) | gray;
7     }
8     // 画素を ARGB からグレースケール (RGB 成分の平均値) に変換
9     private int rgbToGray(int rgb) {
10         int r = (rgb>>16) & 0xff;
11         int g = (rgb>>8) & 0xff;
12         int b = rgb & 0xff;
13         return (r+g+b)/3;
14     }
15     public GrayImage(BufferedImage img) {
16         super(img.getWidth(),img.getHeight(),TYPE_3BYTE_BGR);
17         /* (A) */
18     }
19     // 指定画素のグレースケール (0~255) を返す
20     public int getGray(int x, int y) {
21         /* (B) */
22     }
23     // 指定画素にグレースケールを書き込む
24     public void setGray(int x, int y, int gray) {
25         /* (C) */
26     }
27 }

```