

注意 問題は4問あります。問題中の Java プログラムの行頭の数、説明の都合上付けた行番号であり、プログラムの一部ではありません。解答はすべて別紙の解答用紙に記入しなさい。この定期試験の採点結果およびこの科目の総合成績は8月6日(火) 12:30までに、科目のホームページ <http://www602.math.ryukoku.ac.jp/GKiso/> の「課題の提出と確認」で確認できるようにする予定です。

問題 I 次は、この科目のクラスライブラリを利用した Java プログラムである。このプログラムを起動すると、ウィンドウの中に現れたトランプのデッキ (52枚の1揃えのカードの山) がシャッフルされた後、1枚ずつデッキからカードを引き、そのカードを表向きにして、赤のカード (ハートあるいはダイヤ) の山と黒のカード (スペードあるいはクラブ) の山の2つに分類していく。

```

1 import jp.ac.ryukoku.math.graphics.*;
2
3 class G01 {
4     public static void main(String[] args) {
5         GameFrame f = new GameFrame();
6         Deck d = new Deck();
7         Pile r = new Pile();
8         Pile b = new Pile();
9         f.add(d);
10        f.add(b, 510, 400);
11        f.add(r, 210, 400);
12        d.shuffle();
13        while (d.count() > 0) {
14            Card c = d.pickUp();
15            c.flip();
16            if (c.suit == Suit.HEARTS || c.suit == Suit.DIAMONDS) {
17                c.moveTo(r);
18            } else {
19                c.moveTo(b);
20            }
21        }
22    }
23 }

```

このプログラムに現れている下表中の語が、次のいずれであるかを、解答用紙の解答欄に1～8の数字で示しなさい。(20点)

- | | | | |
|----------------|-----------|-------------|----------------|
| 1. Java のキーワード | 2. クラス名 | 3. インタフェース名 | 4. インスタンス変数名 |
| 5. インスタンスメソッド名 | 6. クラス変数名 | 7. クラスメソッド名 | 8. ローカル(局所)変数名 |

行番号	語
4	public
4	main
6	Deck
9	add
14	d

行番号	語
14	pickUp
16	suit
16	Suit
16	HEARTS
19	c

問題Ⅱ 次の Java プログラムをコンパイルして、java G02 を実行したとする。このプログラムの 47 ~ 51 行目の文が実行される際に出力される文字列を、それぞれ解答用紙の解答欄に書きなさい。(20 点)

G02.java

```
1 class P {
2     String s = "A";
3
4     P() {
5         this("B");
6     }
7
8     P(String s) {
9         this.s = s;
10    }
11
12    String get() {
13        return "C";
14    }
15 }
16
17 class Q extends P {
18     Q () {
19     }
20
21     Q (String s) {
22         super(s);
23     }
24
25     String get() {
26         return s;
27     }
28 }
29
30 class R extends P {
31     R () {
32         super("D");
33     }
34
35     String get() {
36         return s + "E";
37     }
38 }
39
40 class G02 {
41     public static void main(String[] args) {
42         P p = new P();
43         P q = new Q();
44         P r = new R();
45         P s = new P("F");
46         P t = new Q("G");
47         System.out.println(p.get());
48         System.out.println(q.get());
49         System.out.println(r.get());
50         System.out.println(s.get());
51         System.out.println(t.get());
52     }
53 }
```

問題 III 次の Java プログラムは、全く同じ機能を持つ 2 つのクラスを宣言したものである。この 2 つのクラスは、ともに javax.swing.JPanel サブクラスであり、そのインスタンスが GUI 部品として画面に表示されると、中央に黄色で塗り潰された円を表示する。また、マウスでクリックすると、クリックされた位置が中心となるように黄色の円を移動させる。このプログラムの空欄を埋めてそれぞれのクラス宣言を完成させなさい。(30 点)

G03.java

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

class CirclePanel1 (1) {
    int ulx, uly, radius;

    CirclePanel1(int r) {
        radius = r;
        ulx = uly = (2);
        setPreferredSize(new Dimension(400, 400));
        addMouseListener((3));
    }

    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        g.setColor(Color.YELLOW);
        (4) (ulx, uly, 2*radius, 2*radius);
    }

    public void mousePressed(MouseEvent e) {
        ulx = e.getX() - radius;
        uly = e.getY() - radius;
        repaint();
    }

    public void mouseReleased(MouseEvent e) { }
    public void mouseEntered(MouseEvent e) { }
    public void mouseExited(MouseEvent e) { }
    public void mouseClicked(MouseEvent e) { }
}

class CirclePanel2 extends JPanel {
    int cx, cy, radius;

    CirclePanel2(int r) {
        radius = r;
        cx = cy = 200;
        setPreferredSize(new Dimension(400, 400));
        (5) (new (6) {
            (7)
        });
    }

    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        g.setColor(new Color((8), (9), (10)));
        (4)と同じ ((11), (12), 2*radius, 2*radius);
    }
}

```

問題 IV Stat クラスは、数値データをいくつか登録すると、その個数や平均値を教えてくれるようなオブジェクトのクラスである。このクラスのインスタンスには、あらかじめ登録できる数値データの範囲(有効範囲)が設定されており、この範囲外の数値データを登録しようとしても無視される。有効範囲はコンストラクタの引数で指定し、生成されたインスタンスの有効範囲を後で変更することはできない。

このクラスのインスタンスは、設定された有効範囲に関する情報以外に、次の2つの情報をインスタンス変数の値として保持しているが、これらのインスタンス変数は Stat クラスの外部からはアクセスできない。

- 登録されたデータの個数 (int 型)
- 登録されたデータの総和 (double 型)

これら2つの情報から、登録されたデータの平均値を計算することができるので、登録された個々の数値データ自身は記録されない。

次の表は、この Stat クラスのコンストラクタ、インスタンス変数、インスタンスメソッドをまとめたものであり、その下の G04.java は、Stat クラスを利用したプログラムの例である。デフォルトパッケージに属するクラスとして、このような Stat クラスのクラス宣言を書きなさい。(30点)

Stat クラス — 登録した数値データの平均値を計算するオブジェクトのクラス

コンストラクタ Stat() Stat(double min, double max)	有効範囲の最小値が 0.0、最大値が 100.0 であるインスタンス 有効範囲の最小値が min、最大値が max であるインスタンス
インスタンス変数 final double min final double max	有効範囲の最小値(これより小さいデータは無視される) 有効範囲の最大値(これより大きいデータは無視される)
インスタンスメソッド boolean addDatum(double d) int getCount() double getMean() void clear()	d が有効範囲内であれば、d を登録し、true を戻り値として返す。有効範囲外であれば登録せずに、false を返す。 登録されたデータの個数を戻り値として返す。 登録されたデータの平均値を戻り値として返す。登録されたデータの個数が 0 の場合は NaN を返す ¹ 。 登録されたデータをすべて抹消する(登録されたデータの個数も総和も 0 となる)。

G04.java

```

class G04 {
    public static void main(String[] args) {
        Stat s = new Stat();
        for (String a : args) {
            try {
                if (!s.addDatum(Double.parseDouble(a))) {
                    System.err.println("範囲外のデータは無視します : " + a);
                }
            }
            catch (NumberFormatException e) {
                System.err.println("書式誤りのデータは無視します : " + a);
            }
        }
        if (s.getCount() > 0)
            System.out.println("平均値は " + s.getMean() + " です");
    }
}
  
```

(定期試験問題終り)

¹Java では 0.0/0 を評価すると NaN となることに注意

問題 I

行番号	語	解答欄
4	public	1
4	main	7
6	Deck	2
9	add	5
14	d	8

行番号	語	解答欄
14	pickUp	5
16	suit	4
16	Suit	2
16	HEARTS	6
19	c	8

問題 II

47 System.out.println(p.get());

C

48 System.out.println(q.get());

B

49 System.out.println(r.get());

DE

50 System.out.println(s.get());

C

51 System.out.println(t.get());

G

問題 III

(1) extends JPanel implements ActionListener

(2) 200 - radius (3) this (4) g.fillOval

(5) addMouseListener (6) ActionListener ()

(7)

```
public void mousePressed(MouseEvent e) {
    cx = e.getX();
    cy = e.getY();
    repaint();
}
```

```
public void mouseReleased(MouseEvent e) { }
public void mouseEntered(MouseEvent e) { }
public void mouseExited(MouseEvent e) { }
public void mouseClicked(MouseEvent e) { }
```

(8) 255 (9) 255 (10) 0

(11) cx - radius (12) cy - radius

問題 IV

```
class Stat {
    final double min, max;
    private double sum;
    private int num;

    Stat () {
        this(0.0, 100.0);
    }

    Stat (double min, double max) {
        this.min = min;
        this.max = max;
    }

    boolean addDatum(double data) {
        if (min <= data && data <= max) {
            sum += data;
            num++;
            return true;
        }
        return false;
    }

    int getCount() {
        return num;
    }

    double getMean() {
        return sum / num;
    }

    void clear() {
        sum = 0.0;
        num = 0;
    }
}
```