

オブジェクト指向言語・中間テスト問題用紙 (2024年 06月 13日・ 09:35 ～ 10:20)

解答上、その他の注意事項

1. 問題は、問 I ～ II までである。
2. 解答用紙の右上の欄に学籍番号・名前を記入すること。
3. 解答欄を間違えないよう注意すること。
4. 解答中の文字 (特に a と d) がはっきりと区別できるよう注意すること。
5. 持ち込みは不可である。筆記用具・時計・学生証以外のものは、かばんの中などにしまうこと。
6. 中間テストの配点は 30 点である。

すべての問に対する補足

プログラムの空欄を埋める問題では、解答が長くなる可能性があるので、下の省略形 (○ 囲み文字) を用いても良い。(必ず ○ で囲むこと。)

(A) ActionListener (aA) addActionListener (AE) ActionEvent (K) KeyListener
(aK) addKeyListener (KE) KeyEvent (M) MouseListener (aM) addMouseListener
(ME) MouseEvent (pl) System.out.println (pf) System.out.printf

また、参考のために問題用紙の末尾に授業配布プリントの MouseTest.java, LeftRightButton3.java, のソースを掲載する。(main メソッドは省略している。)

I. 次の (1)～(2) の Java に関する多肢選択問題に答えよ。解答は各問の指示する選択肢から選べ。ただし、特に指定しない限り、選ぶべき選択肢は必ずしも一つとは限らない。

(1) 変数 `list` を `int` 型を保存するための、空の `ArrayList` に初期化するための宣言として正しいものはどれか? 一つ選べ。

- (A). `ArrayList<int> list = new ArrayList<int>()`
- (B). `int<ArrayList> list = new int<>()`
- (C). `ArrayList<Integer> list = new ArrayList<>()`
- (D). `Integer<ArrayList> list = new Integer<ArrayList>()`

(2) 次の文章のうち、正しいものはどれか?

- (A). Java は論理型言語に分類されるプログラミング言語の一つである。
- (B). Java 言語をブラウザ上で実行するために、ECMA で仕様を定めたものが JavaScript である。
- (C). Java では、配列の範囲外のアクセスを検出して例外を発生する。
- (D). Java は C++ よりも前に公表された、より古い言語である。

II. 次の枠内の文章は `java.lang.Math` クラスの `random` メソッド、`java.awt.Color` クラスの `darker` メソッド、`brighter` メソッド、`getHSBColor` メソッドの Java™ API 仕様からの抜粋である。（問題を解くのに関係ない部分は割愛している。また、翻訳を独自に変えたところがある。）

パッケージ `java.lang`

クラス `Math`

メソッドの詳細

`random`

```
public static double random()
```

0.0 以上で 1.0 未満の、正の `double` 値を返します。

戻り値:

0.0 以上 1.0 未満の疑似乱数の `double` 値。

パッケージ `java.awt`

クラス `Color`

メソッドの詳細

`brighter`

```
public Color brighter()
```

この `Color` をより明るくした新しい `Color` を作成します。

brighter と darker は逆の操作ですが、これらの 2 つのメソッドを連続して呼び出した場合、丸め誤差のために結果が一致しないことがあります。

戻り値:

alpha 値は同じで、この Color をより明るくした、新しい Color オブジェクト

darker

```
public Color darker()
```

この Color をより暗くした新しい Color を作成します。

brighter と darker は逆の操作ですが、これらの 2 つのメソッドを連続して呼び出した場合、丸め誤差のために結果が一致しないことがあります。

戻り値:

alpha 値は同じで、この Color をより暗くした、新しい Color オブジェクト

getHSBColor

```
public static Color getHSBColor(float h, float s, float b)
```

HSB カラーモデルに指定された値に基づいて、Color オブジェクトを作成します。

s 成分と b 成分は、0 と 1 の間の浮動小数点値 (0.0 から 1.0 までの範囲の数値) にするようにしてください。h 成分は、任意の浮動小数点数値にできます。この数の下限[†]が減算され、0 から 1 の間的小数部が作成されます。この小数に 360 が乗算され、HSB カラーモデルの色相角度が生成されます。

パラメーター:

h – 色相成分
s – 色の彩度
b – 色の明度

戻り値:

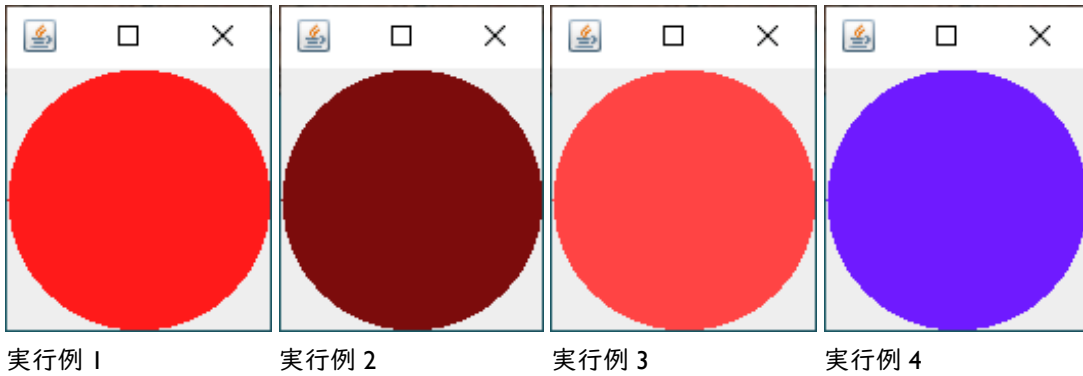
指定された色相、彩度、明度を持つ Color オブジェクト

[†]原文は “floor”、この数より大きくない最大の整数のこと

次の ColorCircle クラスはマウスのクリックに反応する GUI アプリケーションである。画面の中央に円を描き、

- 円の右半分でマウスがクリックされれば、円の色を明るくし、
- 円の左半分でマウスがクリックされれば、円の色を暗くし、
- 円の外側でマウスがクリックされれば、円の色を乱数を用いて変更する。

次のスクリーンショットは、順に、最初の状態、2 回左半分をクリックしたとき、5 回右半分をクリックしたとき、外側をクリックしたとき、の画面である。



ColorCircle クラスは JPanel クラスを継承している。以下の問に答えよ。

ファイル名: ColorCircle.java

```

1  import java.awt.*;
2  import java.awt.event.*;
3  import javax.swing.*;
4
5  public class ColorCircle (1) {
6      Color color;
7      float s0 = (float)0.9, b0 = (float)1;
8
9      public ColorCircle() {
10         setPreferredSize(new Dimension(128, 128));
11         color = Color.getHSBColor((float)1, s0, b0);
12         addMouseListener((2));
13     }
14
15     public void paintComponent(Graphics g) {
16         super.paintComponent(g);
17         g.setColor(color);
18         g.fillOval(0, 0, 128, 128);
19     }
20
21     public void mouseClicked(MouseEvent e) {
22         int x = e.getX(), y = e.getY();
23         int dx = x - 64, dy = y - 64;
24         if (dx * dx + dy * dy >= 64 * 64) {
25             color = Color.getHSBColor((3),
26                                     s0, b0);
27         } else if (dx < 0) {
28             (4)
29         } else {
30             (5)
31         }
32         repaint();
33     }
34
35     public void mousePressed(MouseEvent e) {}
36     public void mouseReleased(MouseEvent e) {}
37     public void mouseEntered(MouseEvent e) {}
38     public void mouseExited(MouseEvent e) {}
39
40     /* main は省略する */
41 }

```

(1) の空欄を埋めよ。

(2) の空欄を埋めよ。

(3) には、0.0 以上 1.0 未満の float 型の疑似乱数を表す式が入る。この式を答えよ。

(4) には、color 変数の値を、元の値をより暗くした Color に変更する文が入る。この文を答えよ。

(5) には、color 変数の値を、元の値をより明るくした Color に変更する文が入る。この文を答えよ。

さらに、この ColorCircle.java を匿名クラスを用いて次のように同等のプログラム ColorCircle2.java に書き換える。やはり ColorCircle2 クラスは JPanel クラスを継承している。以下の問に答えよ。

ファイル名: ColorCircle2.java

```
1 import java.awt.*;
2 import java.awt.event.*;
3 import javax.swing.*;
4
5 public class ColorCircle2 (6) {
6     Color color;
7     float s0 = (float)0.9, b0 = (float)1;
8
9     public ColorCircle2() {
10         setPreferredSize(new Dimension(128, 128));
11         color = Color.getHSBColor((float)1, s0, b0);
12         addMouseListener((7));
13     }
14     /* paintComponent は同じなので省略する */
15     /* main は省略する */
16 }
```

(6) の空欄を埋めよ。

(7) の空欄に入る式を、匿名クラスを用いて書け。ただし、ColorCircle.java と同じ部分は、`/* 元の AB 行目から YZ 行目と同じ */` のように省略して書いて良い。

以下に参考のために授業配布プリントの `MouseTest.java`, `LeftRightButton3.java` のソースを掲載する。

ファイル `MouseTest.java`

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 public class MouseTest extends JPanel implements MouseListener {
6     private int x = 50, y = 20;
7     public MouseTest() {
8         setPreferredSize(new Dimension(150, 150));
9         addMouseListener(this);
10    }
11
12    public void mouseClicked(MouseEvent e) {
13        x = e.getX(); y = e.getY(); repaint();
14    }
15    public void mousePressed(MouseEvent e) {}
16    public void mouseReleased(MouseEvent e) {}
17    public void mouseEntered(MouseEvent e) {}
18    public void mouseExited(MouseEvent e) {}
19
20    @Override
21    public void paintComponent(Graphics g) {
22        super.paintComponent(g);
23        g.drawString("HELLO WORLD!", x, y);
24    }
25
26    public static void main(String[] args) { /* 省略 */ }
27 }
```

ファイル `LeftRightButton2.java`

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 public class LeftRightButton3 extends JPanel {
6     private int x = 20;
7     public LeftRightButton3() {
8         setPreferredSize(new Dimension(200, 70));
9         JButton lBtn = new JButton("Left");
10        JButton rBtn = new JButton("Right");
11        lBtn.addActionListener(new ActionListener() {
12            public void actionPerformed(ActionEvent e) {
13                x -= 10; repaint();
14            }
15        });
16        rBtn.addActionListener(new ActionListener() {
17            public void actionPerformed(ActionEvent e) {
18                x += 10; repaint();
19            }
20        });
21        add(lBtn); add(rBtn);
22    }
23
24    @Override
25    public void paintComponent(Graphics g) {
26        super.paintComponent(g);
27        g.drawString("HELLO WORLD!", x, 55);
28    }
29
30    public static void main(String[] args) { /* 省略 */ }
31 }
```

オブジェクト指向言語・中間テスト解答用紙 (2024 年 06 月 13 日)

学籍番号		氏名	
------	--	----	--

I. (3, 3)

(1)		(2)	
-----	--	-----	--

II. (3,3,3,3,3,6)

(1)	
(2)	
(3)	
(4)	
(5)	
(6)	
(7)	

