

Упражнение

на компютърни науки първи курс

към курса СДиООП

Предоставен ви е частично направен проект, който представлява прозорец, по който има разни бутончета и красоти. Вие може да разгледате как работи проекта, но това не се очаква от вас. От вас се очаква да създадете 4 класа. Три от тях ще наследяват класа AbstractShape, който представлява следното:

```
import java.awt.*;
import java.util.*;

public abstract class AbstractShape {
    Point[] points;
    Graphics g;
    Color fg;
    protected boolean fill;

    private int[][] pointsToArrays(){
        int[][] new_points = new int[2][ points.length ];
        for (int i = 0; i < points.length; ++i){
            new_points[0][i] = points[i].x;
            new_points[1][i] = points[i].y;
        }
        return new_points;
    }

    public AbstractShape(Graphics g){
        this.g = g;
    }

    public AbstractShape(Graphics g, Color fg, boolean fill){
        this.g = g;
        this.fg = fg;
        this.fill = fill;
    }

    public void draw(){
        if(fill)fill();
        else drawContur();
    }
}
```

```

private void drawContur()
{
    int[][] p = pointsToArrays();
    if(fg != null){
        g.setColor(fg);
    }
    g.drawPolygon(p[0], p[1], p[0].length);
}

private void fill() {
    int[][] p = pointsToArrays();
    if(fg != null){
        g.setColor(fg);
    }
    g.drawPolygon( p[0], p[1], p[0].length );
    g.fillPolygon( p[0], p[1], p[0].length );
}

```

/**

* Никога не бих написал този метод принципно, но на теория не трябва да
 знаете нищо от awt/swing, за да си напишете задачата, тъй че реших да ви напиша
 * този толквоа сложен метод :D

*/

```

public Color createRGBColor(int r, int g, int b){
    return new Color(r, g, b);
}

```

/**

* Пресмята лицето на фигурата.
 * Не приема никакви аргументи
 * връща като стойност пресметнатото лице

*/

```

public abstract double getSurface();

```

/**

* Изчислява обиколката на фигурата по най-ефикасния начин. В случай, че
 фигурата е самопресичаща се, да не обръща внимание на това :D

* Не приема аргументи
 * Връща намерената обиколка

*/

```

public abstract int getCircumferance();

```

```

/**
 * Да чете три цели числа от диапазона 0 - 255. Ако някое от числата е извън
 този интервал да хвърля грешка, в която да се казва, че е въведен невалиден цвят.
 * Не трябва да получава аргументи.
 * Връща резултат цвята от прочетените стойности, като за създаване на цвята
 използва функцията createRGBColor(int, int, int).
 */
public Color readColor() throws Exception {
    throw new Exception("Method readColor is not implemented yet - go and
implement it!");
}

/**
 * Взима като аргумент цвят и задава fg да е равен на него, като в следствие на
 това цвята на фигурата се слага на този цвят.
 * Взима като аргумент обект от тип Color.
 * Не връща резултат
 */
public void setColor(Color a) throws Exception {
    throw new Exception("Method setColor(Color) is not implemented yet - go and
implement it!");
}

/**
 * Извиква readColor() и слага цвята, получен от нея за цвят на фигурата. Ако
 readColor хвърли грешка, която не е за неправилен цвят, да прехвърли грешката
 * нагоре. В противен случай да пробва още веднъж да прочете цвят и ако
 отново не успее, да остави стария цвят.
 * Не приема аргументи
 * Връща дали е сменила цвята успешно или се е провалила
 */
public void setColor() throws Exception {
    throw new Exception("Method setColor() is not implemented yet - go and
implement it!");
}
}

```

Всеки от наследяващите класове трябва да има конструктор с 4 аргумента Graphics, Point[]/ArrayList<Point>, Color, boolean. А четвъртия ще е InvalidColorException, който представлява изключение. Както се вижда, част от AbstractShape трябва да се довърши – направете го.

Очаква се да се направи клас за правоъгълник, който да се казва Rectangle, да наследява AbstractShape, клас Shape, който също да наследява AbstractShape и клас Triangle, който (ха познайте) да наследява класа AbstractShape. Като за всеки от тях трябва да има имплементирани абстрактните методи от AbstractShape.

Ако всичко се имплементира подходящо, в края на краищата ще имаме работеща програмка, която може да чертае запълнен или не червен, черен или цианов триъгълник, правоъгълник или произволен многоъгълник.