Үнэнсайхан Мэндсайхан (**20B1NUM1146**)
**Програмчлалын арга зүй**
CSII204

**Тайлан**

Номын Бүлэг 2

1. Type in the **HelloProgram.java** program exactly as it appears in this chapter and get it working. Change the message so that it reads “I love Java” instead. Add your name as a signature to the lower right corner.

```java
import acm.graphics.*;
import acm.program.*;

public class HelloProgram extends GraphicsProgram {

        public void run() {
                this.resize(200, 200);
                add(new GLabel("I love Java", 100, 75));
                add(new GLabel("Mendsaikhan", 125, 200));
        }

}
```

2. The following program was written without comments or instructions to the user, except for a couple of input prompts:

```java
import acm.program.*;

public class MyProgram extends ConsoleProgram {
   public void run() {
      double b = readDouble("Enter b: ");
      double h = readDouble("Enter h: ");
      double a = (b * h) / 2;
      println("a = " + a);
   }
}
```

Read through the program and figure out what it is doing. What result is it calculating? Rewrite this program so it is easier to understand, both for the user and for the programmer who must modify the program in the future.

19 lines (13 sloc) | 429 Bytes

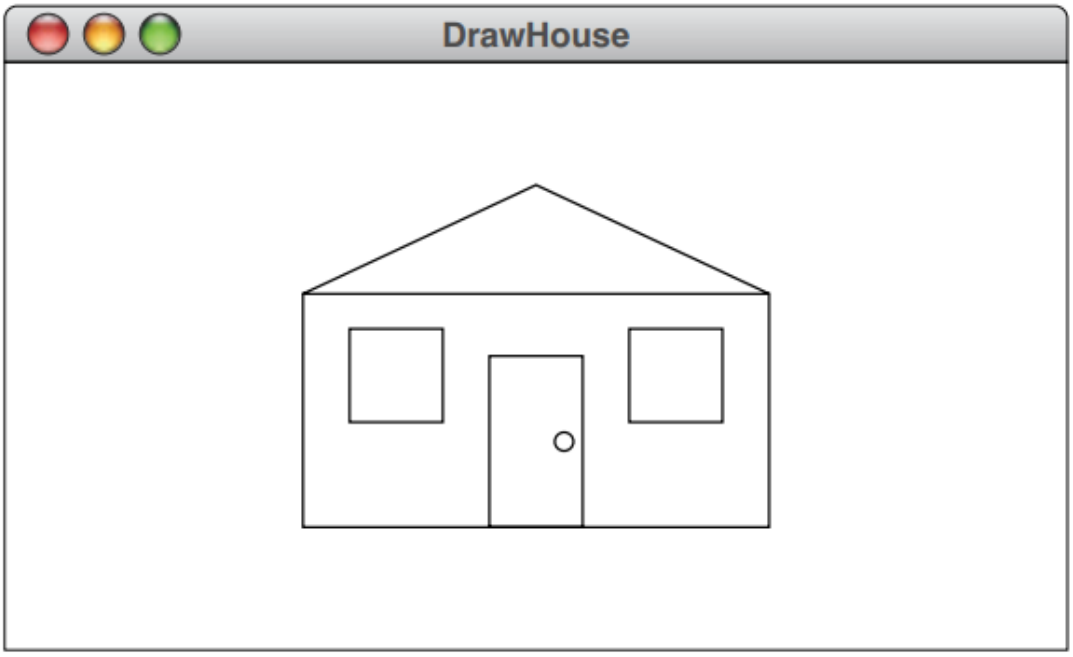
```java
import acm.program.*;

public class TriangleArea extends ConsoleProgram {

        public void run() {
                println("This program calculates S(area) of triangle.");

                // Getting input from user
                double b = readDouble("Enter base: ");
                double h = readDouble("Enter height: ");

                // Calculating area by formula
                double s = (b * h) / 2;

                // Printing calculated area
                println("Area of triangle = " + s);
        }

}
```

3. Extend the **`Add2Integers`** program shown in Figure 2-2 so that it adds three integers instead.

17 lines (11 sloc) | 348 Bytes

```
1   import acm.program.*;
2
3   public class Add3Integers extends ConsoleProgram {
4
5           public void run() {
6                   println("This program adds three integers.");
7
8                   int n1 = readInt("Enter n1: ");
9                   int n2 = readInt("Enter n2: ");
10                  int n3 = readInt("Enter n3: ");
11
12                  int total = n1 + n2 + n3;
13
14                  println("The total is " + total + ".");
15          }
16
17  }
```

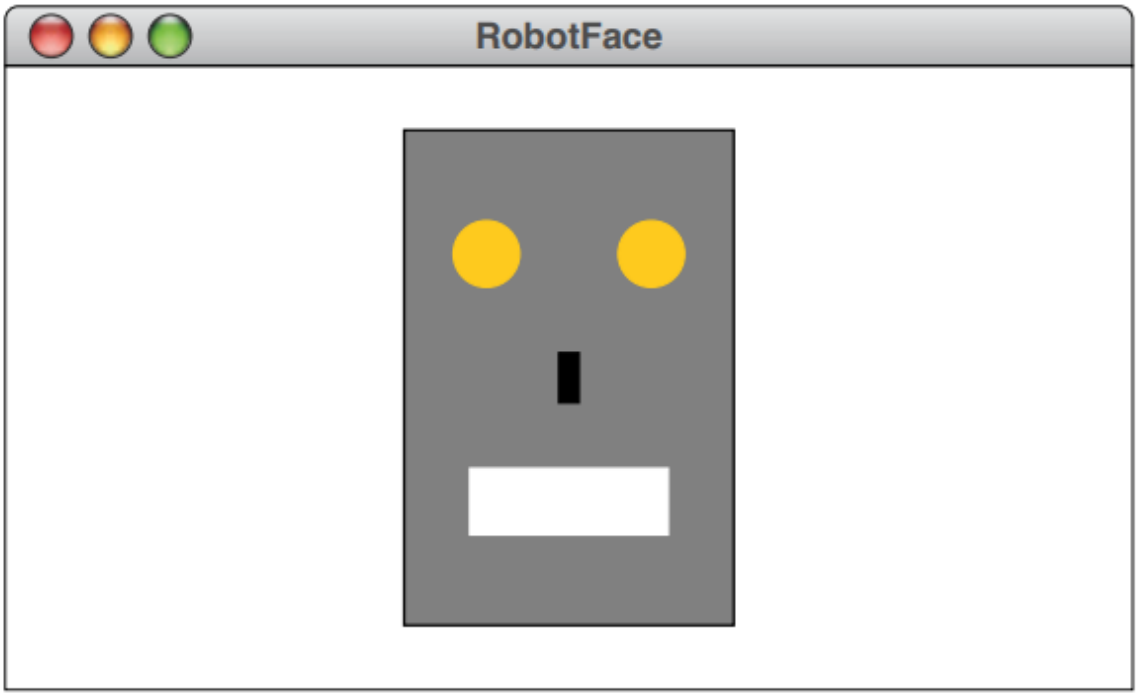
4. Write a **`GraphicsProgram`** that generates the following simple picture of a house with a peaked roof, two windows, and a door with a circular doorknob:



22 lines (13 sloc) | 399 Bytes

```
1   import acm.graphics.*;
2   import acm.program.*;
3
4   public class DrawHouse extends GraphicsProgram {
5           public void run() {
6
7                   add(new GRect(200, 150, 400, 200));
8
9                   add(new GRect(240, 200, 75, 75));
10
11                  add(new GRect(480, 200, 75, 75));
12
13                  add(new GRect(375, 240, 60, 110));
14
15                  add(new GOval(420, 290, 12, 12));
16
17                  add(new GLine(200, 150, 400, 50));
18
19                  add(new GLine(600, 150, 400, 50));
20          }
21
22  }
```

5. Write a **`GraphicsProgram`** that draws the following picture of a robot face:



35 lines (27 sloc) | 727 Bytes

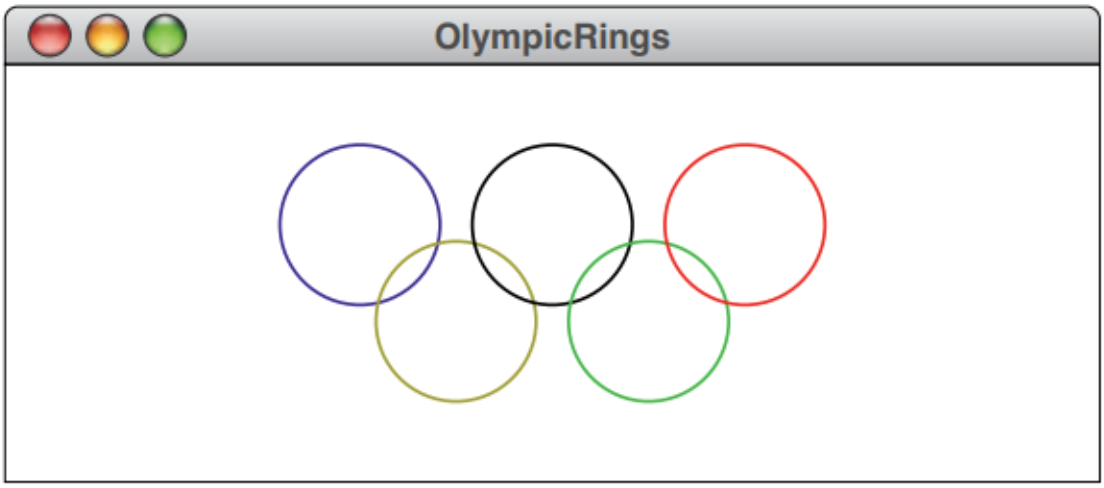
```java
import java.awt.Color;

import acm.program.*;
import acm.graphics.*;

public class RobotFace extends GraphicsProgram {

	public void run() {
		GRect face = new GRect(50, 50, 200, 300);
		face.setFilled(true);
		face.setFillColor(Color.GRAY);
		add(face);

		GOval eyeL = new GOval(75, 75, 50, 50);
		eyeL.setFilled(true);
		eyeL.setFillColor(Color.ORANGE);
		add(eyeL);

		GOval eyeR = new GOval(175, 75, 50, 50);
		eyeR.setFilled(true);
		eyeR.setFillColor(Color.ORANGE);
		add(eyeR);

		GRect nose = new GRect(140, 175, 20, 50);
		nose.setFilled(true);
		nose.setFillColor(Color.black);
		add(nose);

		GRect mouth = new GRect(60, 250, 180, 60);
		mouth.setFilled(true);
		mouth.setFillColor(Color.WHITE);
		add(mouth);
	}

}
```

6. Write a **`GraphicsProgram`** that draws the following picture of an archery target, which also happens to be the logo of a large discount chain:

30 lines (24 sloc) | 852 Bytes

```
1   import acm.graphics.*;
2   import acm.program.*;
3   import java.awt.*;
4
5   public class Target extends GraphicsProgram {
6
7           private static final int PIXELS_PER_INCH = 72;
8           private static final double RADIUS_OUTER_CIRCLE = 1.0;
9           private static final double RADIUS_WHITE_CIRCLE = 0.65;
10          private static final double RADIUS_INNER_CIRCLE = 0.3;
11
12
13          public void run() {
14                  double x = getWidth() / 2;
15                  double y = getHeight() /2;
16                  drawCircle(x, y, RADIUS_OUTER_CIRCLE * PIXELS_PER_INCH, Color.RED);
17                  drawCircle(x, y, RADIUS_WHITE_CIRCLE * PIXELS_PER_INCH, Color.WHITE);
18                  drawCircle(x, y, RADIUS_INNER_CIRCLE * PIXELS_PER_INCH, Color.RED);
19          }
20
21          private void drawCircle(double cx, double cy, double r, Color c) {
22                  double x = cx - r;
23                  double y = cy - r;
24                  GOval circle = new GOval(2*r, 2*r);
25                  circle.setColor(c);
26                  circle.setFilled(true);
27                  add(circle, x, y);
28          }
29
30  }
```

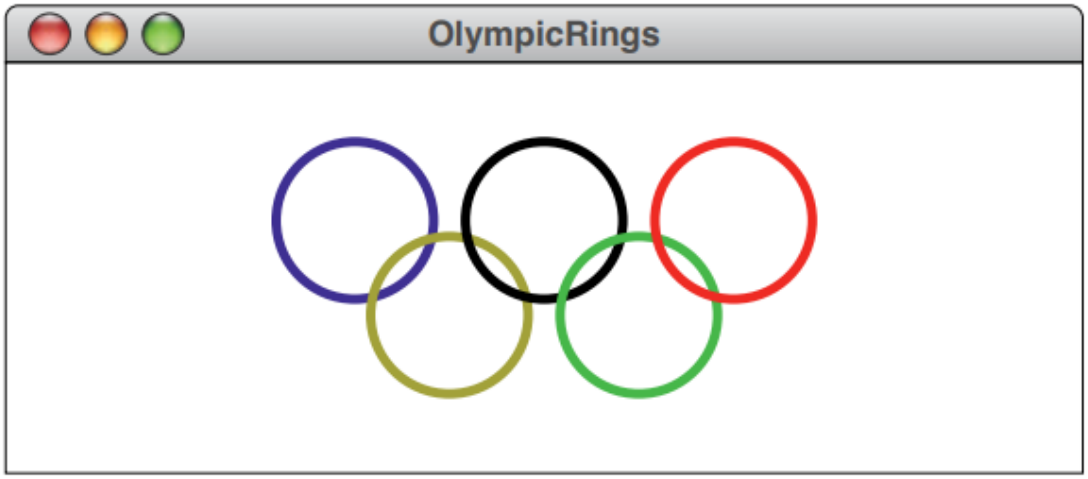
7. Write a **`GraphicsProgram`** that draws the five interlocking rings (blue, yellow, black, green, and red) that form the symbol of the Olympic Games:




32 lines (23 sloc) | 768 Bytes

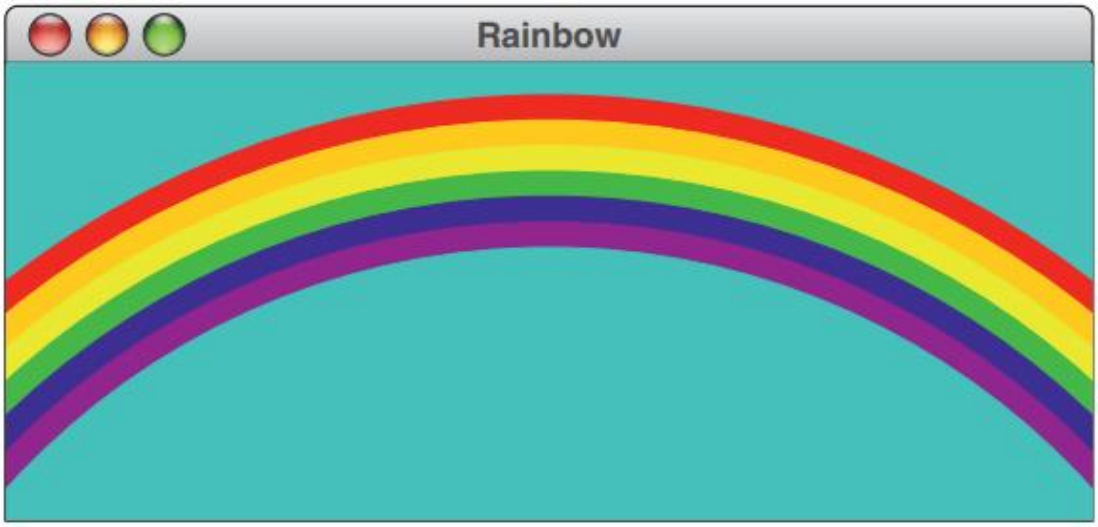
```java
import java.awt.Color;

import acm.program.*;
import acm.graphics.*;

public class OlympicRings extends GraphicsProgram {

        public void run() {
                GOval blueRing = new GOval(50, 50, RING_SIZE, RING_SIZE);
                blueRing.setColor(Color.BLUE);
                add(blueRing);

                GOval blackRing = new GOval(170, 50, RING_SIZE, RING_SIZE);
                blackRing.setColor(Color.BLACK);
                add(blackRing);

                GOval redRing = new GOval(290, 50, RING_SIZE, RING_SIZE);
                redRing.setColor(Color.RED);
                add(redRing);

                GOval yellowRing = new GOval(110, 110, RING_SIZE, RING_SIZE);
                yellowRing.setColor(Color.YELLOW);
                add(yellowRing);

                GOval greenRing = new GOval(230, 110, RING_SIZE, RING_SIZE);
                greenRing.setColor(Color.GREEN);
                add(greenRing);
        }

    private static final int RING_SIZE = 100;

}
```

8. On most output devices, the Olympic Games logo from the preceding exercise doesn't show up all that well because the yellow circle (and to a lesser extent the green one) tends to disappear against the white background of the window. Part of the problem is that the outlines that Java draws for the **`GOval`** class are only one pixel wide, which doesn't show up well when drawn in lighter colors. It would be easier to see the rings if the borders were three pixels wide, like this:



```java
import java.awt.Color;

import acm.program.*;
import acm.graphics.*;

public class OlympicRingsBold extends GraphicsProgram {

	public void run() {

		GOval blueRing1 = new GOval(50, 50, ringSize, ringSize);
		blueRing1.setColor(Color.BLUE);
		add(blueRing1);
		GOval blueRing2 = new GOval(51, 51, ringSize-2, ringSize-2);
		blueRing2.setColor(Color.BLUE);
		add(blueRing2);
		GOval blueRing3 = new GOval(52, 52, ringSize-4, ringSize-4);
		blueRing3.setColor(Color.BLUE);
		add(blueRing3);

		GOval blackRing1 = new GOval(170, 50, ringSize, ringSize);
		blackRing1.setColor(Color.BLACK);
		add(blackRing1);
		GOval blackRing2 = new GOval(171, 51, ringSize-2, ringSize-2);
		blackRing2.setColor(Color.BLACK);
		add(blackRing2);
		GOval blackRing3 = new GOval(172, 52, ringSize-4, ringSize-4);
		blackRing3.setColor(Color.BLACK);
		add(blackRing3);

		GOval redRing1 = new GOval(290, 50, ringSize, ringSize);
		redRing1.setColor(Color.RED);
		add(redRing1);
		GOval redRing2 = new GOval(291, 51, ringSize-2, ringSize-2);
		redRing2.setColor(Color.RED);
		add(redRing2);
		GOval redRing = new GOval(292, 52, ringSize-4, ringSize-4);
		redRing.setColor(Color.RED);
		add(redRing);

		GOval yellowRing1 = new GOval(110, 110, ringSize, ringSize);
		yellowRing1.setColor(Color.YELLOW);
		add(yellowRing1);
		GOval yellowRing2 = new GOval(111, 111, ringSize-2, ringSize-2);
		yellowRing2.setColor(Color.YELLOW);
		add(yellowRing2);
		GOval yellowRing3 = new GOval(112, 112, ringSize-4, ringSize-4);
		yellowRing3.setColor(Color.YELLOW);
		add(yellowRing3);

		GOval greenRing1 = new GOval(230, 110, ringSize, ringSize);
		greenRing1.setColor(Color.GREEN);
		add(greenRing1);
		GOval greenRing2 = new GOval(231, 111, ringSize-2, ringSize-2);
		greenRing2.setColor(Color.GREEN);
		add(greenRing2);
		GOval greenRing3 = new GOval(232, 112, ringSize-4, ringSize-4);
		greenRing3.setColor(Color.GREEN);
		add(greenRing3);
	}

	private static final int RING_SIZE = 100;

}
```

9. Write a **`GraphicsProgram`** that draws a picture of a rainbow that looks something like this:



60 lines (48 sloc) | 1.34 KB

```java
import java.awt.Color;

import acm.program.*;
import acm.graphics.*;

public class Rainbow extends GraphicsProgram {

	public void run() {
		this.resize(760, 500);

		GRect sky = new GRect(0, 0, 760, 500);
		sky.setColor(Color.CYAN);
		sky.setFilled(true);
		sky.setFillColor(Color.CYAN);
		add(sky);

		GOval red = new GOval(10, 120, 740, 740);
		red.setColor(Color.RED);
		red.setFilled(true);
		red.setFillColor(Color.RED);
		add(red);

		GOval orange = new GOval(40, 150, 680, 680);
		orange.setColor(Color.ORANGE);
		orange.setFilled(true);
		orange.setFillColor(Color.ORANGE);
		add(orange);

		GOval yellow = new GOval(70, 180, 620, 620);
		yellow.setColor(Color.YELLOW);
		yellow.setFilled(true);
		yellow.setFillColor(Color.YELLOW);
		add(yellow);

		GOval green = new GOval(100, 210, 560, 560);
		green.setColor(Color.GREEN);
		green.setFilled(true);
		green.setFillColor(Color.GREEN);
		add(green);

		GOval blue = new GOval(130, 240, 500, 500);
		blue.setColor(Color.BLUE);
		blue.setFilled(true);
		blue.setFillColor(Color.BLUE);
		add(blue);

		GOval magenta = new GOval(160, 270, 440, 440);
		magenta.setColor(Color.MAGENTA);
		magenta.setFilled(true);
		magenta.setFillColor(Color.MAGENTA);
		add(magenta);

		GOval cyan = new GOval(190, 300, 380, 380);
		cyan.setColor(Color.CYAN);
		cyan.setFilled(true);
		cyan.setFillColor(Color.CYAN);
		add(cyan);
	}

}
```