

OOP & GUI

Provimi periodik 2/2, Forma: A

Emri: _____

Nr. indeksit: _____

Drejtimi: _____

Data: _____

Për pyetjet 1–3 shqyrtoni aplikacionin vijues.

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

public class ColorFrame extends JFrame
{ private String[] model = {"Kuq", "Kalter", "Verdh"};
  private JList list = new JList(model);

  public ColorFrame()
  { Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(list);
    ColorButton button = new ColorButton("Ngjyros", this);
    cp.add(button);
    setTitle("Prapavija");
    pack();
    setVisible(true);
  }

  public void update()
  { int i = list.getSelectedIndex();
    String s = model[i];
    Container cp = getContentPane();
    if ( s.equals("Kuq") )
    { cp.setBackground(Color.red); }
    else if ( s.equals("Kalter") )
    { cp.setBackground(Color.blue); }
    else if ( s.equals("Verdh") )
    { cp.setBackground(Color.yellow); }
  }
}

import javax.swing.*;
import java.awt.event.*;
public class ColorButton extends JButton implements ActionListener
{ private ColorFrame frame;
  public ColorButton(String label, ColorFrame f)
  { super(label);
    frame = f;
    addActionListener(this);
  }

  public void actionPerformed(ActionEvent e)
  { frame.update(); }
}

public class ColorTest
{ public static void main(String[] args)
```

```
{ new ColorFrame(); }
}
```

1. Përshkruani çfarë ndodh pas startimit të aplikacionit `ColorTest`.
2. Çfarë ndodh në qoftë se klikohet butoni `Ngjyro` para se të selektohet ndonjë hyrje në listën? Shpjegoni. Modifikoni aplikacionin ashtu që në rast të shtypjes së tillë të butonit të mos ndodh asgjë.
3. Modifikoni aplikacionin ashtu që lista të afishojë edhe një hyrje shtesë: `Gjelbërt`, e cila, në qoftë se selektohet, vepron përkatësisht, në mënyrë analoge me hyrjet tjera.

Për pyetjet 4-8 shqyrtoni aplikacionin vijues.

```
/** Modelon numëruesin. */
public class Counter
{ private int count;
  /** Inicializon numëruesin.
   * @param start - vlera fillestare e numëruesit */
  public Counter(int start)
  { count = start; }
  /** Azhuron numëruesin. */
  public void increment()
  { count = count + 1; }
  /** I qaset numëruesit.
   * @return vlera e numëruesit count */
  public int countOf()
  { return count; }
}

import java.awt.event.*;
import javax.swing.*;
/** Definon një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view */
  public CountButton(String l, Counter m, Frame v)
  { super(l);
    view = v;
    model = m;
    addActionListener(this);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
   * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment();
    view.update();
  }
}

import java.awt.*;
import javax.swing.*;
/** Një kornizë labela e së cilës afishon
```

```

* sa herë është shtypur butoni i saj. */
public class Frame extends JFrame
{ private Counter count;
  private JLabel label = new JLabel("Numri = 0");
  /** Krijon kornizë me labelë dhe buton.
   * @param c - objekti model, numërues */
  public Frame(Counter c)
  { count = c;
    Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(label);
    cp.add(new CountButton("OK", count, this));
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }

  public void update()
  { label.setText("Numri = " + count.countOf()); }
}

/** Starton aplikacionin. */
public class TestApplication
{ public static void main(String[] args)
  { Counter model = new Counter(0);
    Frame view = new Frame(model);
  }
}

```

4. Përshkruani çfarë ndodh pas startimit të aplikacionit `TestApplication`. Vizatoni diagramin e klasave për këtë aplikacion.
5. Modifikoni klasën `Frame` ashtu që të jetë interfejs grafike për një aplikacion i cili ka tre objekte `Counter`, dhe të përmbajë tre butona, secili nga të cilët kur të klikohet rrit njërin nga numëruesit. Vlerat e tre numëruesve afishoni me anë të labelave.
6. Modifikoni zgjidhjen e detyrës paraprake ashtu që secili buton të afishojë vetë numrin sa herë është klikuar. (Udhëzim: Shfrytëzoni metodën `setText(String s)` të klasës `JButton`.)
7. Modifikoni aplikacionin e dhënë ashtu që aplikacioni të ketë një buton `Rrit`, një buton `Zvogëlo` dhe një labelë e cila afishon një numër të plotë. Secilën herë që shtypet butoni `Rrit` numri i plotë rritet, kurse secilën herë që shtypet butoni `Zvogëlo` numri zvogëlohet.
8. Modifikoni aplikacionin ashtu që klasa `Frame` të jetë interfejs grafik për një aplikacion i cili ka një objekt `Counter`, dhe të përmbajë një buton me ngjyrë të kuqe, një me ngjyrë të kaltër dhe një me ngjyrë të verdhë. Kur të klikohet cilido nga butonat rrit numëruesin. Vlera e numëruesit afishohet në një labelë. Ngjyra e labelës ndryshohet në ngjyrën e butonit të shtypur së fundi.
9. Krijoni një kalkulator për fëmijë: Interfejsi i tij afishon dy fusha teksti, një labelë, një listë me katër hyrje, ku hyrjet janë `+`, `-`, `*` dhe `/`, dhe një buton. Kur fëmiju të rradhisë dy numra thyesorë në dy fushat e tekstit, të selektojë njërin nga katër hyrjet dhe të klikojë butonin, kalkulatori llogarit operacionin e selektuar aritmetik mbi dy numrat dhe afishon rezultatin në labelën.
10. Krijoni një GUI i cili ka pamjen e një kalkulatori të thjeshtë. GUI afishon 12 butona (për dhjetë shifrat `0...9`, pikën decimale dhe butonin `C` për fshirje) të renditur në një rrjetë 4×3 , dhe një displej numerik (labelë) në pjesën e sipërme. Displeji afishon numrin përkatës sipas pullave të shtypura nga shfrytëzuesi.

Çelësi i provimit A

Për pyetjet 1-3 shqyrtoni aplikacionin vijues.

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

public class ColorFrame extends JFrame
{ private String[] model = {"Kuq", "Kalter", "Verdh"};
  private JList list = new JList(model);

  public ColorFrame()
  { Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(list);
    ColorButton button = new ColorButton("Ngjyros", this);
    cp.add(button);
    setTitle("Prapavija");
    pack();
    setVisible(true);
  }

  public void update()
  { int i = list.getSelectedIndex();
    String s = model[i];
    Container cp = getContentPane();
    if ( s.equals("Kuq") )
    { cp.setBackground(Color.red); }
    else if ( s.equals("Kalter") )
    { cp.setBackground(Color.blue); }
    else if ( s.equals("Verdh") )
    { cp.setBackground(Color.yellow); }
  }
}

import javax.swing.*;
import java.awt.event.*;
public class ColorButton extends JButton implements ActionListener
{ private ColorFrame frame;
  public ColorButton(String label, ColorFrame f)
  { super(label);
    frame = f;
    addActionListener(this);
  }

  public void actionPerformed(ActionEvent e)
  { frame.update(); }
}

public class ColorTest
{ public static void main(String[] args)
  { new ColorFrame(); }
}
```

1. Përshkruani çfarë ndodh pas startimit të aplikacionit `ColorTest`.

Përgjegjja: Afishohet një kornizë, me titull `Prapavija`, e cila përmban një listë me tri hyrje: `Kuq`, `Kaltër` dhe `Verdh`, dhe një pullë `Ngjyros`. Pas selektimit të njëres nga hyrjet dhe shtypjes së butonit `prapavija` (e panos së përmbajtjes) së kornizës ngjyroset në ngjyrën përkatëse.

2. Çfarë ndodh në qoftë se klikohet butoni `Ngjyros` para se të selektohet ndonjë hyrje në listën? Shpjegoni. Modifikoni aplikacionin ashtu që në rast të shtypjes së tillë të butonit të mos ndodh asgjë.

Përgjegjja: Në qoftë se klikohet butoni `Ngjyros` para se të selektohet ndonjë hyrje në listën, atëherë shkaktohet gabim gjatë kohës së ekzekutimit i tipit `ArrayIndexOutOfBoundsException`. Një gjë e tillë ndodh për shkak se metoda `getSelectedIndex` e `JList` para selektimit të një hyrjeje kthen vlerën `-1`, e cila shkakton gabimin gjatë përdorimit për indeksim të vargut `model` në metodën `update` të klasës `ColorFrame`.

Për të mos ndodhur një gjë e tillë, bëjmë modifikimin vijues në klasën `ColorFrame`.

```
// ... sikur më parë
public class ColorFrame extends JFrame
{ // ... sikur më parë
    public void update()
    { int i = list.getSelectedIndex();
      if ( i != -1 )
      { String s = model[i];
        Container cp = getContentPane();
        if ( s.equals("Kuq") )
        { cp.setBackground(Color.red); }
        else if ( s.equals("Kalter") )
        { cp.setBackground(Color.blue); }
        else if ( s.equals("Verdh") )
        { cp.setBackground(Color.yellow); }
      }
    }
}
```

3. Modifikoni aplikacionin ashtu që lista të afishojë edhe një hyrje shtesë: `Gjelbërt`, e cila, në qoftë se selektohet, vepron përkatësisht, në mënyrë analoge me hyrjet tjera.

Përgjegjja:

```
// ... sikur më parë
public class ColorFrame extends JFrame
{ private String[] model = {"Kuq", "Kalter", "Verdh", "Gjelbërt"};
  // ... sikur më parë
  public void update()
  { // ... sikur më parë
    else if ( s.equals("Gjelbërt") )
    { cp.setBackground(Color.green); }
  }
}
```

Për pyetjet 4-8 shqyrtoni aplikacionin vijues.

```
/** Modelon numëruesin. */
public class Counter
{ private int count;
  /** Inicializon numëruesin.
```

```

    * @param start - vlera fillestare e numëruesit */
public Counter(int start)
{ count = start; }
/** Azhuron numëruesin. */
public void increment()
{ count = count + 1; }
/** I qaset numëruesit.
    * @return vlera e numëruesit count */
public int countOf()
{ return count; }
}

import java.awt.event.*;
import javax.swing.*;
/** Definin një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  /** Ndërton kontrolluesin.
    * @param l - mbishkrimi në butonin
    * @param m - objekti model
    * @param v - objekti view */
  public CountButton(String l, Counter m, Frame v)
  { super(l);
    view = v;
    model = m;
    addActionListener(this);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
    * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment();
    view.update();
  }
}

import java.awt.*;
import javax.swing.*;
/** Një kornizë labela e së cilës afishon
    * sa herë është shtypur butoni i saj. */
public class Frame extends JFrame
{ private Counter count;
  private JLabel label = new JLabel("Numri = 0");
  /** Krijon kornizë me labelë dhe buton.
    * @param c - objekti model, numërues */
  public Frame(Counter c)
  { count = c;
    Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(label);
    cp.add(new CountButton("OK", count, this));
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }
}

```

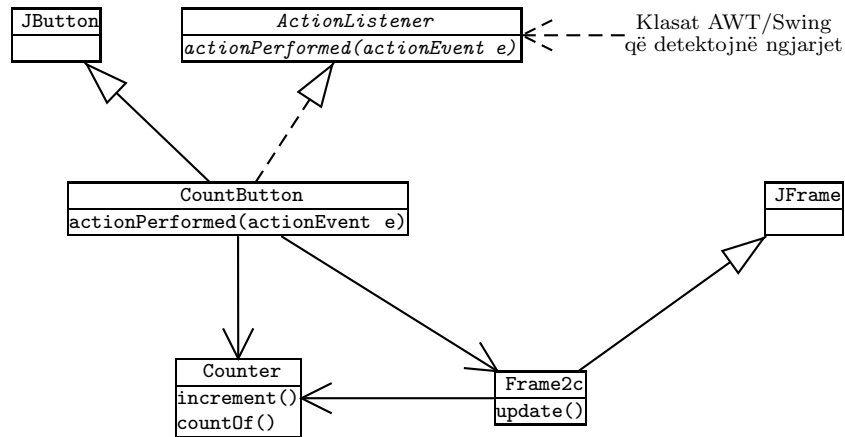


Fig. 1: Diagrami i klasave për aplikacionin TestApplication

```

public void update()
{ label.setText("Numri = " + count.countOf()); }
}

/** Starton aplikacionin. */
public class TestApplication
{ public static void main(String[] args)
  { Counter model = new Counter(0);
    Frame view = new Frame(model);
  }
}

```

4. Përshkruani çfarë ndodh pas startimit të aplikacionit TestApplication. Vizatoni diagramin e klasave për këtë aplikacion.

Përgjegjja: Afshohet një kornizë me titull Korniza për numërim në të cilën ndodhen një buton OK dhe një labelë me tekstin Numri = 0. Sa herë që shtypet butoni, numri i afishuar në labelën rritet për një.

Diagrami i klasave është dhënë në figurën.

5. Modifikoni klasën Frame ashtu që të jetë interfejs grafike për një aplikacion i cili ka tre objekte Counter, dhe të përmbajë tre butona, secili nga të cilët kur të klikohet rrit njërin nga numëruesit. Vlerat e tre numëruesve afshoni me anë të labelave.

Përgjegjja:

```

/** Një kornizë tri labelat e së cilës afishojnë
 * sa herë janë shtypur secili nga tre butonat e saj. */
public class Frame extends JFrame
{ private Counter[] counts;
  private JLabel[] labels = new JLabel[3];
  /** Krijon kornizë me tri labela dhe tre butona.
   * @param c - objekti model, numërues */
  public Frame(Counter[] c)
  { counts = c;
    for ( int i = 0; i < 3; i++ )

```

```

        { labels[i] = new JLabel("Numri = 0"); }
        Container cp = getContentPane();
        cp.setLayout(new GridLayout(3, 1));
        JPanel[] p = new JPanel[3];
        for ( int i = 0; i < 3; i++ )
        { p[i] = new JPanel();
          p[i].add(labels[i]);
          p[i].add(new CountButton("Rrite " + (i + 1), counts[i], this));
          cp.add(p[i]);
        }
        setTitle("Korniza për numërim");
        pack();
        setVisible(true);
    }

    public void update()
    { for ( int i = 0; i < 3; i++ )
      { labels[i].setText("Numri = " + counts[i].countOf()); }
    }
}

/** Starton aplikacionin. */
public class TestApplication
{ public static void main(String[] args)
  { Counter[] models =
    {new Counter(0), new Counter(0), new Counter(0)};
    Frame view = new Frame(models);
  }
}

```

6. Modifikoni zgjidhjen e detyrës paraprakeashtu që secili buton të afishojë vetë numrin sa herë është klikuar. (Udhëzim: Shfrytëzoni metodën `setText(String s)` të klasës `JButton`.)

Përgjegjja:

```

import java.awt.*;
import javax.swing.*;
/** Një kornizë butonat e së cilës afishojnë sa herë janë shtypur. */
public class Frame extends JFrame
{ private Counter[] counts;
  private CountButton[] buttons = new CountButton[3];
  /** Krijon kornizë me tre butona.
   * @param c - objekti model, numëruar */
  public Frame(Counter[] c)
  { counts = c;
    Container cp = getContentPane();
    cp.setLayout(new GridLayout(3, 1));
    for ( int i = 0; i < 3; i++ )
    { buttons[i] = new CountButton("Numri = 0", counts[i], this);
      cp.add(buttons[i]);
    }
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }
}

```



```

public void update()
{ for ( int i = 0; i < 3; i++ )
  { buttons[i].setText("Numri = " + counts[i].countOf()); }
}
}

```

7. Modifikoni aplikacionin e dhënë ashtu që aplikacioni të ketë një buton Rrit, një buton Zvogëlo dhe një labelë e cila afishon një numër të plotë. Secilën herë që shtypet butoni Rrit numri i plotë rritet, kurse secilën herë që shtypet butoni Zvogëlo numri zvogëlohet.

Përgjegjja: Modifikojmë klasat vijuese nga aplikacioni i dhënë fillimisht.

```

/** Modelon një numërues i cili mund të zvogëlohet. */
public class Counter
{ // ... sikur më parë
  /** Rrit numëruesin për hapin.
   * @param step - vlera e hapit */
  public void increment(int step)
  { count = count + step; }
  // ... sikur më parë
}

import java.awt.event.*;
import javax.swing.*;
/** Definon një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  private int step;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view
   * @param s - hapi i rritjes së numëruesit */
  public CountButton(String l, Counter m, int s, Frame v)
  { super(l);
    view = v;
    model = m;
    step = s;
    addActionListener(this);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
   * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment(step);
    view.update();
  }
}

import java.awt.*;
import javax.swing.*;
/** Një kornizë labela e së cilës azhurohet
 * përmes butonave përkatës. */
public class Frame extends JFrame
{ // ... sikur më parë
  /** Krijon kornizë me labelë dhe dy butona.

```

```

    * @param c - objekti model, numërues */
public Frame(Counter c)
{ // ... sikur më parë
    cp.add(label);
    JPanel p = new JPanel(new GridLayout(2, 1));
    p.add(new CountButton("Rrit", count, 1, this));
    p.add(new CountButton("Zvogëlo", count, -1, this));
    cp.add(p);
    setTitle("Korniza për numërim");
    // ... sikur më parë
}
// ... sikur më parë
}

```

8. Modifikoni aplikacionin ashtu që klasa `Frame` të jetë interfejs grafik për një aplikacion i cili ka një objekt `Counter`, dhe të përmbajë një buton me ngjyrë të kuqe, një me ngjyrë të kaltër dhe një me ngjyrë të verdhë. Kur të klikohet cilido nga butonat rrit numëruesin. Vlera e numëruesit afishohet në një labelë. Ngjyra e labelës ndryshohet në ngjyrën e butonit të shtypur së fundi.

Përgjegjja: Modifikojmë klasat vijuese nga aplikacioni i dhënë fillimisht.

```

import java.awt.*;
// ... sikur më parë
/** Definon një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  private Color color;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view
   * @param c - ngjyra e butonit */
  public CountButton(String l, Color c, Counter m, Frame v)
  { // ... sikur më parë
    color = c;
    setBackground(color);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
   * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment();
    view.setColor(color);
    view.update();
  }
}

// ... sikur më parë
/** Një kornizë labela e së cilës afishon me ngjyrë
 * sa herë janë shtypur butonat e saj. */
public class Frame extends JFrame
{ private Counter count;
  private JLabel label = new JLabel("Numri = 0");
  Color color = Color.black;
  /** Krijon kornizë me labelë dhe tre butona.
   * @param c - objekti model, numërues */

```

```

public Frame(Counter c)
{ // ... sikur më parë
  cp.add(label);
  cp.add(new ExamCountButton("OK", Color.red, count, this));
  cp.add(new ExamCountButton("OK", Color.blue, count, this));
  cp.add(new ExamCountButton("OK", Color.yellow, count, this));
  setTitle("Korniza për numërim");
  // ... sikur më parë
}

public void setColor(Color c)
{ color = c; }

public void update()
{ label.setForeground(color);
  label.setText("Numri = " + count.countOf());
}
}

```

9. Krijoni një kalkulator për fëmijë: Interfejsi i tij afishon dy fusha teksti, një labelë, një listë me katër hyrje, ku hyrjet janë +, -, * dhe /, dhe një buton. Kur fëmiu të rradhisë dy numra thyesorë në dy fushat e tekstit, të selektojë njëren nga katër hyrjet dhe të klikojë butonin, kalkulatori llogarit operacionin e selektuar aritmetik mbi dy numrat dhe afishon rezultatin në labelën.

Përgjegjja:

```

import javax.swing.*;
import java.awt.*;
/** Korniza e kalkulatorit për fëmijë. */
public class KidsCalcFrame extends JFrame
{ private JTextField operand1 = new JTextField("", 10);
  private JTextField operand2 = new JTextField("", 10);
  private JLabel result = new JLabel("");
  private String[] operators = {"+", "-", "*", "/"};
  private JList list = new JList(operators);
  /** Ndërton kornizën. */
  public KidsCalcFrame()
  { Container cp = getContentPane();
    cp.setLayout(new GridLayout(1,2));
    JPanel p1 = new JPanel(new GridLayout(3,1));
    p1.add(operand1);
    p1.add(operand2);
    p1.add(result);
    cp.add(p1);
    JPanel p2 = new JPanel(new BorderLayout());
    p2.add(list, BorderLayout.CENTER);
    p2.add(new KidsCalcButton("Llogarit", this), BorderLayout.SOUTH);
    cp.add(p2);
    pack();
    setVisible(true);
  }
  /** I shton stringun displejit. */
  public void calculate()
  { double op1 = new Double(operand1.getText().trim()).doubleValue();
    double op2 = new Double(operand2.getText().trim()).doubleValue();

```

```

        int i = list.getSelectedIndex();
        double answer = 0.0;
        if ( i != -1 )
        { char ch = operators[i].charAt(0);
          switch (ch)
          { case '+' : answer = op1 + op2;
            break;
            case '-' : answer = op1 - op2;
            break;
            case '*' : answer = op1 * op2;
            break;
            case '/' : answer = op1 / op2;
            }
          result.setText("" + answer);
        }
    }
}

import javax.swing.*;
import java.awt.event.*;
/** Butoni për llogaritje i kalkulatorit për fëmijë. */
public class KidsCalcButton extends JButton implements ActionListener
{ private KidsCalcFrame view;
  /** Ndërton butonin.
   * @param label - labela e butonit */
  public KidsCalcButton(String label, KidsCalcFrame v)
  { super(label);
    view = v;
    addActionListener(this);
  }
  /** Përpunon aksionin e shtypjes së butonit.
   * @param e - objekti aksion */
  public void actionPerformed(ActionEvent e)
  { view.calculate(); }
}

/** Starton kalkulatorin për fëmijë. */
public class KidsCalculator
{ public static void main(String[] args)
  { new KidsCalcFrame(); }
}

```

10. Krijoni një GUI i cili ka pamjen e një kalkulatori të thjeshtë. GUI afishon 12 butona (për dhjetë shifrat 0...9, pikën decimale dhe butonin C për fshirje) të renditur në një rrjetë 4×3 , dhe një displej numerik (labelë) në pjesën e sipërme. Displeji afishon numrin përkatës sipas pullave të shtypura nga shfrytëzuesi.

Përgjegjja:

```

import javax.swing.*;
import java.awt.*;
/** Korniza e kalkulatorit */
public class CalcFrame extends JFrame
{ private JLabel display = new JLabel("0");
  /** Ndërton kornizën. */
  public CalcFrame()

```

```

{ Container cp = getContentPane();
  cp.setLayout(new BorderLayout());
  JPanel p1 = new JPanel(new FlowLayout(FlowLayout.RIGHT));
  p1.add(display);
  cp.add(p1, BorderLayout.NORTH);
  JPanel p2 = new JPanel(new GridLayout(4, 3));
  p2.add(new DigitButton("0", this));
  p2.add(new DigitButton(".", this));
  p2.add(new CButton("C", this));
  for (int i = 1; i < 10; i++ )
  { p2.add(new DigitButton("" + i, this)); }
  cp.add(p2, BorderLayout.CENTER);
  pack();
  setVisible(true);
}

/** I shton stringun displejit.
 * @param d - stringu që shtohet */
public void addDisplay(String d)
{ String value = display.getText();
  if ( value.equals("0") && !d.equals("."))
  { display.setText(d); }
  else { display.setText(value + d); }
}

/** Pastron displejin. */
public void clearDisplay()
{ display.setText("0"); }
}

import javax.swing.*;
import java.awt.event.*;
/** Butoni i shifrave. */
public class DigitButton extends JButton implements ActionListener
{ private String value;
  private CalcFrame view;
  /** Ndërton butonin.
   * @param label - vlera e shifrës */
  public DigitButton(String label, CalcFrame v)
  { super(label);
    value = label;
    view = v;
    addActionListener(this);
  }
  /** Përpunon aksionin e shtypjes së butonit.
   * @param e - objekti aksion */
  public void actionPerformed(ActionEvent e)
  { view.addDisplay(value); }
}

import javax.swing.*;
import java.awt.event.*;
/** Butoni i shifrave. */
public class CButton extends JButton implements ActionListener
{ private CalcFrame view;
  /** Ndërton butonin. */
  public CButton(String label, CalcFrame v)
  { super(label);

```

```
        view = v;
        addActionListener(this);
    }
    /** Përpunon aksionin e shtypjes së butonit.
     * @param e - objekti aksion */
    public void actionPerformed(ActionEvent e)
    { view.clearDisplay(); }
}

/** Starton kalkulatorin. */
public class Calculator
{ public static void main(String[] args)
  { new CalcFrame(); }
}
```

OOP & GUI

Provimi periodik 2/2, Forma: B

Emri: _____

Nr. indeksit: _____

Drejtimi: _____

Data: _____

1. Krijoni një kalkulator për fëmijë: Interfejsi i tij afishon dy fusha teksti, një labelë, një listë me katër hyrje, ku hyrjet janë +, -, * dhe /, dhe një buton. Kur fëmiju të rradhisë dy numra thyesorë në dy fushat e tekstit, të selektojë njëren nga katër hyrjet dhe të klikojë butonin, kalkulatori llogarit operacionin e selektuar aritmetik mbi dy numrat dhe afishon rezultatin në labelën.

Për pyetjet 2-6 shqyrtoni aplikacionin vijues.

```
/** Modelon numëruesin. */
public class Counter
{ private int count;
  /** Inicializon numëruesin.
   * @param start - vlera fillestare e numëruesit */
  public Counter(int start)
  { count = start; }
  /** Azhuron numëruesin. */
  public void increment()
  { count = count + 1; }
  /** I qaset numëruesit.
   * @return vlera e numëruesit count */
  public int countOf()
  { return count; }
}

import java.awt.event.*;
import javax.swing.*;
/** Definin një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view */
  public CountButton(String l, Counter m, Frame v)
  { super(l);
    view = v;
    model = m;
    addActionListener(this);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
   * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment();
    view.update();
  }
}

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

```

/** Një kornizë labela e së cilës afishon
 * sa herë është shtypur butoni i saj. */
public class Frame extends JFrame
{ private Counter count;
  private JLabel label = new JLabel("Numri = 0");
  /** Krijon kornizë me labelë dhe buton.
   * @param c - objekti model, numërues */
  public Frame(Counter c)
  { count = c;
    Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(label);
    cp.add(new CountButton("OK", count, this));
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }

  public void update()
  { label.setText("Numri = " + count.countOf()); }
}

/** Starton aplikacionin. */
public class TestApplication
{ public static void main(String[] args)
  { Counter model = new Counter(0);
    Frame view = new Frame(model);
  }
}

```

2. Përshkruani çfarë ndodh pas startimit të aplikacionit `TestApplication`. Vizatoni diagramin e klasave për këtë aplikacion.
3. Modifikoni klasën `Frame` ashtu që të jetë interfejs grafik për një aplikacion i cili ka tre objekte `Counter`, dhe të përmbajë tre butona, secili nga të cilët kur të klikohet rrit njërin nga numëruesit. Vlerat e tre numëruesve afishoni me anë të labelave.
4. Modifikoni zgjidhjen e detyrës paraprake ashtu që secili buton të afishojë vetë numrin sa herë është klikuar. (Udhëzim: Shfrytëzoni metodën `setText(String s)` të klasës `JButton`.)
5. Modifikoni aplikacionin e dhënë ashtu që aplikacioni të ketë një buton `Rrit`, një buton `Zvogëlo` dhe një labelë e cila afishon një numër të plotë. Secilën herë që shtypet butoni `Rrit` numri i plotë rritet, kurse secilën herë që shtypet butoni `Zvogëlo` numri zvogëlohet.
6. Modifikoni aplikacionin ashtu që klasa `Frame` të jetë interfejs grafik për një aplikacion i cili ka një objekt `Counter`, dhe të përmbajë një buton me ngjyrë të kuqe, një me ngjyrë të kaltër dhe një me ngjyrë të verdhë. Kur të klikohet cilido nga butonat rrit numëruesin. Vlera e numëruesit afishohet në një labelë. Ngjyra e labelës ndryshohet në ngjyrën e butonit të shtypur së fundi.

Për pyetjet 7–9 shqyrtoni aplikacionin vijues.

```

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

public class ColorFrame extends JFrame
{ private String[] model = {"Kuq", "Kalter", "Verdh"};

```



```

private JList list = new JList(model);

public ColorFrame()
{ Container cp = getContentPane();
  cp.setLayout(new FlowLayout());
  cp.add(list);
  ColorButton button = new ColorButton("Ngjyros", this);
  cp.add(button);
  setTitle("Prapavija");
  pack();
  setVisible(true);
}

public void update()
{ int i = list.getSelectedIndex();
  String s = model[i];
  Container cp = getContentPane();
  if ( s.equals("Kuq") )
  { cp.setBackground(Color.red); }
  else if ( s.equals("Kalter") )
  { cp.setBackground(Color.blue); }
  else if ( s.equals("Verdh") )
  { cp.setBackground(Color.yellow); }
}
}

import javax.swing.*;
import java.awt.event.*;
public class ColorButton extends JButton implements ActionListener
{ private ColorFrame frame;
  public ColorButton(String label, ColorFrame f)
  { super(label);
    frame = f;
    addActionListener(this);
  }

  public void actionPerformed(ActionEvent e)
  { frame.update(); }
}

public class ColorTest
{ public static void main(String[] args)
  { new ColorFrame(); }
}

```

7. Përshkruani çfarë ndodh pas startimit të aplikacionit `ColorTest`.
8. Çfarë ndodh në qoftë se klikohet butoni `Ngjyros` para se të selektohet ndonjë hyrje në listën? Shpjegoni. Modifikoni aplikacionin ashtu që në rast të shtypjes së tillë të butonit të mos ndodh asgjë.
9. Modifikoni aplikacionin ashtu që lista të afishojë edhe një hyrje shtesë: `Gjelbërt`, e cila, në qoftë se selektohet, vepron përkatësisht, në mënyrë analoge me hyrjet tjera.
10. Krijoni një GUI i cili ka pamjen e një kalkulatori të thjeshtë. GUI afishon 12 butona (për dhjetë shifrat 0...9, pikën decimale dhe butonin `C` për fshirje) të renditur në një rrjetë 4×3 , dhe një displej numerik (labelë) në pjesën e sipërme. Displeji afishon numrin përkatës sipas pullave të shtypura nga shfrytëzuesi.

Çelësi i provimit B

1. Krijoni një kalkulator për fëmijë: Interfejsi i tij afishon dy fusha teksti, një labelë, një listë me katër hyrje, ku hyrjet janë +, -, * dhe /, dhe një buton. Kur fëmiu të rradhisë dy numra thyesorë në dy fushat e tekstit, të selektojë njëren nga katër hyrjet dhe të klikojë butonin, kalkulatori llogarit operacionin e selektuar aritmetik mbi dy numrat dhe afishon rezultatin në labelën.

Përgjegjja:

```
import javax.swing.*;
import java.awt.*;
/** Korniza e kalkulatorit për fëmijë. */
public class KidsCalcFrame extends JFrame
{ private JTextField operand1 = new JTextField("", 10);
  private JTextField operand2 = new JTextField("", 10);
  private JLabel result = new JLabel("");
  private String[] operators = {"+", "-", "*", "/"};
  private JList list = new JList(operators);
  /** Ndërton kornizën. */
  public KidsCalcFrame()
  { Container cp = getContentPane();
    cp.setLayout(new GridLayout(1,2));
    JPanel p1 = new JPanel(new GridLayout(3,1));
    p1.add(operand1);
    p1.add(operand2);
    p1.add(result);
    cp.add(p1);
    JPanel p2 = new JPanel(new BorderLayout());
    p2.add(list, BorderLayout.CENTER);
    p2.add(new KidsCalcButton("Llogarit", this), BorderLayout.SOUTH);
    cp.add(p2);
    pack();
    setVisible(true);
  }
  /** I shton stringun displejit. */
  public void calculate()
  { double op1 = new Double(operand1.getText().trim()).doubleValue();
    double op2 = new Double(operand2.getText().trim()).doubleValue();
    int i = list.getSelectedIndex();
    double answer = 0.0;
    if ( i != -1 )
    { char ch = operators[i].charAt(0);
      switch (ch)
      { case '+' : answer = op1 + op2;
        break;
        case '-' : answer = op1 - op2;
        break;
        case '*' : answer = op1 * op2;
        break;
        case '/' : answer = op1 / op2;
        }
    }
    result.setText("" + answer);
  }
}
```

```

    }
}

import javax.swing.*;
import java.awt.event.*;
/** Butoni për llogaritje i kalkulatorit për fëmijë. */
public class KidsCalcButton extends JButton implements ActionListener
{ private KidsCalcFrame view;
  /** Ndërton butonin.
   * @param label - labela e butonit */
  public KidsCalcButton(String label, KidsCalcFrame v)
  { super(label);
    view = v;
    addActionListener(this);
  }
  /** Përpunon aksionin e shtypjes së butonit.
   * @param e - objekti aksion */
  public void actionPerformed(ActionEvent e)
  { view.calculate(); }
}

/** Starton kalkulatorin për fëmijë. */
public class KidsCalculator
{ public static void main(String[] args)
  { new KidsCalcFrame(); }
}

```

Për pyetjet 2-6 shqyrtoni aplikacionin vijues.

```

/** Modelon numëruesin. */
public class Counter
{ private int count;
  /** Inicializon numëruesin.
   * @param start - vlera fillestare e numëruesit */
  public Counter(int start)
  { count = start; }
  /** Azhuron numëruesin. */
  public void increment()
  { count = count + 1; }
  /** I qaset numëruesit.
   * @return vlera e numëruesit count */
  public int countOf()
  { return count; }
}

import java.awt.event.*;
import javax.swing.*;
/** Definion një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view */

```

```

public CountButton(String l, Counter m, Frame v)
{ super(l);
  view = v;
  model = m;
  addActionListener(this);
}
/** Përpunon një ngjarje aksion -- shtypje butoni.
 * @param e - ngjarja aksion */
public void actionPerformed(ActionEvent e)
{ model.increment();
  view.update();
}
}

import java.awt.*;
import javax.swing.*;
/** Një kornizë labela e së cilës afishon
 * sa herë është shtypur butoni i saj. */
public class Frame extends JFrame
{ private Counter count;
  private JLabel label = new JLabel("Numri = 0");
  /** Krijon kornizë me labelë dhe buton.
   * @param c - objekti model, numërues */
  public Frame(Counter c)
  { count = c;
    Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(label);
    cp.add(new CountButton("OK", count, this));
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }

  public void update()
  { label.setText("Numri = " + count.countOf()); }
}

/** Starton aplikacionin. */
public class TestApplication
{ public static void main(String[] args)
  { Counter model = new Counter(0);
    Frame view = new Frame(model);
  }
}

```

2. Përshkruani çfarë ndodh pas startimit të aplikacionit `TestApplication`. Vizatoni diagramin e klasave për këtë aplikacion.

Përgjegjja: Afishohet një kornizë me titull `Korniza për numërim` në të cilën ndodhen një buton `OK` dhe një labelë me tekstin `Numri = 0`. Sa herë që shtypet butoni, numri i afishuar në labelën rritet për një.

Diagrami i klasave është dhënë në figurën.

3. Modifikoni klasën `Frame` ashtu që të jetë interfejs grafike për një aplikacion i cili ka tre objekte `Counter`, dhe të përmbajë tre butona, secili nga të cilët kur të klikohet rrit njërin nga numëruesit. Vlerat e tre numëruesve afishoni me anë të labelave.

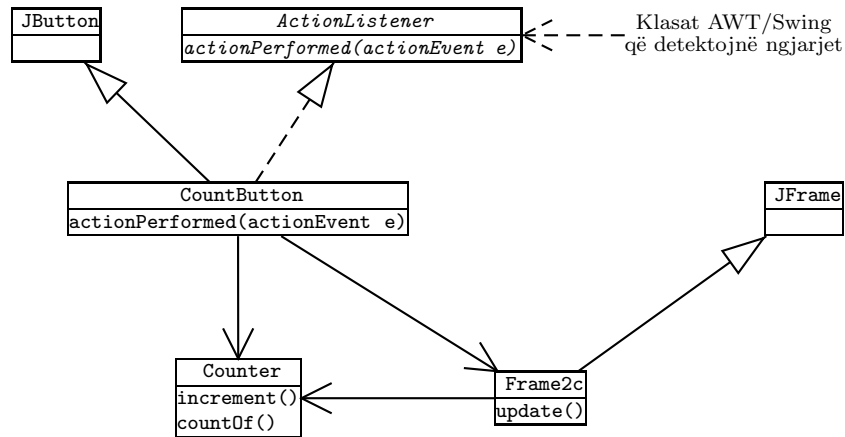


Fig. 2: Diagrami i klasave për aplikacionin TestApplication

Përgjegjja:

```

/** Një kornizë tri labelat e së cilës afishojnë
 * sa herë janë shtypur secili nga tre butonat e saj. */
public class Frame extends JFrame
{ private Counter[] counts;
  private JLabel[] labels = new JLabel[3];
  /** Krijon kornizë me tri labela dhe tre butona.
   * @param c - objekti model, numëruet */
  public Frame(Counter[] c)
  { counts = c;
    for ( int i = 0; i < 3; i++ )
    { labels[i] = new JLabel("Numri = 0"); }
    Container cp = getContentPane();
    cp.setLayout(new GridLayout(3, 1));
    JPanel[] p = new JPanel[3];
    for ( int i = 0; i < 3; i++ )
    { p[i] = new JPanel();
      p[i].add(labels[i]);
      p[i].add(new CountButton("Rrite " + (i + 1), counts[i], this));
      cp.add(p[i]);
    }
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }

  public void update()
  { for ( int i = 0; i < 3; i++ )
    { labels[i].setText("Numri = " + counts[i].countOf()); }
  }
}

/** Starton aplikacionin. */
public class TestApplication
{ public static void main(String[] args)
  { Counter[] models =
  
```

```

        {new Counter(0), new Counter(0), new Counter(0)};
        Frame view = new Frame(models);
    }
}

```

4. Modifikoni zgjidhjen e detyrës paraprakeashtu që secili buton të afishojë vetë numrin sa herë është klikuar. (Udhëzim: Shfrytëzoni metodën `setText(String s)` të klasës `JButon`.)

Përgjegjja:

```

import java.awt.*;
import javax.swing.*;
/** Një kornizë butonat e së cilës afishojnë sa herë janë shtypur. */
public class Frame extends JFrame
{ private Counter[] counts;
  private CountButton[] buttons = new CountButton[3];
  /** Krijon kornizë me tre butona.
   * @param c - objekti model, numërues */
  public Frame(Counter[] c)
  { counts = c;
    Container cp = getContentPane();
    cp.setLayout(new GridLayout(3, 1));
    for ( int i = 0; i < 3; i++ )
    { buttons[i] = new CountButton("Numri = 0", counts[i], this);
      cp.add(buttons[i]);
    }
    setTitle("Korniza për numërim");
    pack();
    setVisible(true);
  }

  public void update()
  { for ( int i = 0; i < 3; i++ )
    { buttons[i].setText("Numri = " + counts[i].countOf()); }
  }
}

```

5. Modifikoni aplikacionin e dhënë ashtu që aplikacioni të ketë një buton `Rrit`, një buton `Zvogëlo` dhe një labelë e cila afishon një numër të plotë. Secilën herë që shtypet butoni `Rrit` numri i plotë rritet, kurse secilën herë që shtypet butoni `Zvogëlo` numri zvogëlohet.

Përgjegjja: Modifikojmë klasat vijuese nga aplikacioni i dhënë fillimisht.

```

/** Modelon një numërues i cili mund të zvogëlohet. */
public class Counter
{ // ... sikur më parë
  /** Rrit numëruesin për hapin.
   * @param step - vlera e hapit */
  public void increment(int step)
  { count = count + step; }
  // ... sikur më parë
}

import java.awt.event.*;
import javax.swing.*;
/** Definon një kontrollues butoni. */

```

```

public class CountButton extends JButton implements ActionListener
{ private Frame view;
  private Counter model;
  private int step;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view
   * @param s - hapi i rritjes së numëruesit */
  public CountButton(String l, Counter m, int s, Frame v)
  { super(l);
    view = v;
    model = m;
    step = s;
    addActionListener(this);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
   * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment(step);
    view.update();
  }
}

import java.awt.*;
import javax.swing.*;
/** Një kornizë labela e së cilës azhurohet
 * përmes butonave përkatës. */
public class Frame extends JFrame
{ // ... sikur më parë
  /** Krijon kornizë me labelë dhe dy butona.
   * @param c - objekti model, numërues */
  public Frame(Counter c)
  { // ... sikur më parë
    cp.add(label);
    JPanel p = new JPanel(new GridLayout(2, 1));
    p.add(new CountButton("Rrit", count, 1, this));
    p.add(new CountButton("Zvogëlo", count, -1, this));
    cp.add(p);
    setTitle("Korniza për numërim");
    // ... sikur më parë
  }
  // ... sikur më parë
}

```

6. Modifikoni aplikacionin ashtu që klasa Frame të jetë interfejs grafik për një aplikacion i cili ka një objekt Counter, dhe të përmbajë një buton me ngjyrë të kuqe, një me ngjyrë të kaltër dhe një me ngjyrë të verdhë. Kur të klikohet cilido nga butonat rrit numëruesin. Vlera e numëruesit afishohet në një labelë. Ngjyra e labelës ndryshohet në ngjyrën e butonit të shtypur së fundi.

Përgjegjja: Modifikojmë klasat vijuese nga aplikacioni i dhënë fillimisht.

```

import java.awt.*;
// ... sikur më parë
/** Definon një kontrollues butoni. */
public class CountButton extends JButton implements ActionListener

```

```

{ private Frame view;
  private Counter model;
  private Color color;
  /** Ndërton kontrolluesin.
   * @param l - mbishkrimi në butonin
   * @param m - objekti model
   * @param v - objekti view
   * @param c - ngjyra e butonit */
  public CountButton(String l, Color c, Counter m, Frame v)
  { // ... sikur më parë
    color = c;
    setBackground(color);
  }
  /** Përpunon një ngjarje aksion -- shtypje butoni.
   * @param e - ngjarja aksion */
  public void actionPerformed(ActionEvent e)
  { model.increment();
    view.setColor(color);
    view.update();
  }
}

// ... sikur më parë
/** Një kornizë labela e së cilës afishon me ngjyrë
 * sa herë janë shtypur butonat e saj. */
public class Frame extends JFrame
{ private Counter count;
  private JLabel label = new JLabel("Numri = 0");
  Color color = Color.black;
  /** Krijon kornizë me labelë dhe tre butona.
   * @param c - objekti model, numërues */
  public Frame(Counter c)
  { // ... sikur më parë
    cp.add(label);
    cp.add(new ExamCountButton("OK", Color.red, count, this));
    cp.add(new ExamCountButton("OK", Color.blue, count, this));
    cp.add(new ExamCountButton("OK", Color.yellow, count, this));
    setTitle("Korniza për numërim");
    // ... sikur më parë
  }

  public void setColor(Color c)
  { color = c; }

  public void update()
  { label.setForeground(color);
    label.setText("Numri = " + count.countOf());
  }
}
}

```

Për pyetjet 7–9 shqyrtoni aplikacionin vijues.

```

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

```



```

public class ColorFrame extends JFrame
{ private String[] model = {"Kuq", "Kalter", "Verdh"};
  private JList list = new JList(model);

  public ColorFrame()
  { Container cp = getContentPane();
    cp.setLayout(new FlowLayout());
    cp.add(list);
    ColorButton button = new ColorButton("Ngjyros", this);
    cp.add(button);
    setTitle("Prapavija");
    pack();
    setVisible(true);
  }

  public void update()
  { int i = list.getSelectedIndex();
    String s = model[i];
    Container cp = getContentPane();
    if ( s.equals("Kuq") )
    { cp.setBackground(Color.red); }
    else if ( s.equals("Kalter") )
    { cp.setBackground(Color.blue); }
    else if ( s.equals("Verdh") )
    { cp.setBackground(Color.yellow); }
  }
}

import javax.swing.*;
import java.awt.event.*;
public class ColorButton extends JButton implements ActionListener
{ private ColorFrame frame;
  public ColorButton(String label, ColorFrame f)
  { super(label);
    frame = f;
    addActionListener(this);
  }

  public void actionPerformed(ActionEvent e)
  { frame.update(); }
}

public class ColorTest
{ public static void main(String[] args)
  { new ColorFrame(); }
}

```

7. Përshkruani çfarë ndodh pas startimit të aplikacionit ColorTest.

Përgjegjja: Afishohet një kornizë, me titull Prapavija, e cila përmban një listë me tri hyrje: Kuq, Kaltër dhe Verdh, dhe një pullë Ngjyros. Pas selektimit të njëres nga hyrjet dhe shtypjes së butonit prapavija (e panos së përmbajtjes) së kornizës ngjyroset në ngjyrën përkatëse.

8. Çfarë ndodh në qoftë se klikohet butoni Ngjyros para se të selektohet ndonjë hyrje në listën? Shpjegoni. Modifikoni aplikacionin ashtu që në rast të shtypjes së tillë të butonit të mos ndodh asgjë.

Përgjegjja: Në qoftë se klikohet butoni Ngjyros para se të selektohet ndonjë hyrje në listën, atëherë shkaktohet gabim gjatë kohës së ekzekutimit i tipit `ArrayIndexOutOfBoundsException`. Një gjë e tillë ndodh për shkak se metoda `getSelectedIndex` e `JList` para selektimit të një hyrjeje kthen vlerën `-1`, e cila shkakton gabimin gjatë përdorimit për indeksim të vargut `model` në metodën `update` të klasës `ColorFrame`.

Për të mos ndodhur një gjë e tillë, bëjmë modifikimin vijues në klasën `ColorFrame`.

```
// ... sikur më parë
public class ColorFrame extends JFrame
{ // ... sikur më parë
    public void update()
    { int i = list.getSelectedIndex();
      if ( i != -1 )
      { String s = model[i];
        Container cp = getContentPane();
        if ( s.equals("Kuq") )
        { cp.setBackground(Color.red); }
        else if ( s.equals("Kalter") )
        { cp.setBackground(Color.blue); }
        else if ( s.equals("Verdh") )
        { cp.setBackground(Color.yellow); }
      }
    }
}
```

9. Modifikoni aplikacionin ashtu që lista të afishojë edhe një hyrje shtesë: Gjelbërt, e cila, në qoftë se selektohet, vepron përkatësisht, në mënyrë analoge me hyrjet tjera.

Përgjegjja:

```
// ... sikur më parë
public class ColorFrame extends JFrame
{ private String[] model = {"Kuq", "Kalter", "Verdh", "Gjelbërt"};
  // ... sikur më parë
  public void update()
  { // ... sikur më parë
    else if ( s.equals("Gjelbërt") )
    { cp.setBackground(Color.green); }
  }
}
```

10. Krijoni një GUI i cili ka pamjen e një kalkulatori të thjeshtë. GUI afishon 12 butona (për dhjetë shifrat `0...9`, pikën decimale dhe butonin `C` për fshirje) të renditur në një rrjetë 4×3 , dhe një displej numerik (labelë) në pjesën e sipërme. Displeji afishon numrin përkatës sipas pullave të shtypura nga shfrytëzuesi.

Përgjegjja:

```
import javax.swing.*;
import java.awt.*;
/** Korniza e kalkulatorit */
public class CalcFrame extends JFrame
{ private JLabel display = new JLabel("0");
  /** Ndërton kornizën. */
  public CalcFrame()
```

```

{ Container cp = getContentPane();
  cp.setLayout(new BorderLayout());
  JPanel p1 = new JPanel(new FlowLayout(FlowLayout.RIGHT));
  p1.add(display);
  cp.add(p1, BorderLayout.NORTH);
  JPanel p2 = new JPanel(new GridLayout(4, 3));
  p2.add(new DigitButton("0", this));
  p2.add(new DigitButton(".", this));
  p2.add(new CButton("C", this));
  for (int i = 1; i < 10; i++ )
  { p2.add(new DigitButton("" + i, this)); }
  cp.add(p2, BorderLayout.CENTER);
  pack();
  setVisible(true);
}

/** I shton stringun displejit.
 * @param d - stringu që shtohet */
public void addDisplay(String d)
{ String value = display.getText();
  if ( value.equals("0") && !d.equals("."))
  { display.setText(d); }
  else { display.setText(value + d); }
}

/** Pastron displejin. */
public void clearDisplay()
{ display.setText("0"); }
}

import javax.swing.*;
import java.awt.event.*;
/** Butoni i shifrave. */
public class DigitButton extends JButton implements ActionListener
{ private String value;
  private CalcFrame view;
  /** Ndërton butonin.
   * @param label - vlera e shifrës */
  public DigitButton(String label, CalcFrame v)
  { super(label);
    value = label;
    view = v;
    addActionListener(this);
  }
  /** Përpunon aksionin e shtypjes së butonit.
   * @param e - objekti aksion */
  public void actionPerformed(ActionEvent e)
  { view.addDisplay(value); }
}

import javax.swing.*;
import java.awt.event.*;
/** Butoni i shifrave. */
public class CButton extends JButton implements ActionListener
{ private CalcFrame view;
  /** Ndërton butonin. */
  public CButton(String label, CalcFrame v)
  { super(label);

```

```
        view = v;
        addActionListener(this);
    }
    /** Përpunon aksionin e shtypjes së butonit.
     * @param e - objekti aksion */
    public void actionPerformed(ActionEvent e)
    { view.clearDisplay(); }
}

/** Starton kalkulatorin. */
public class Calculator
{ public static void main(String[] args)
  { new CalcFrame(); }
}
```