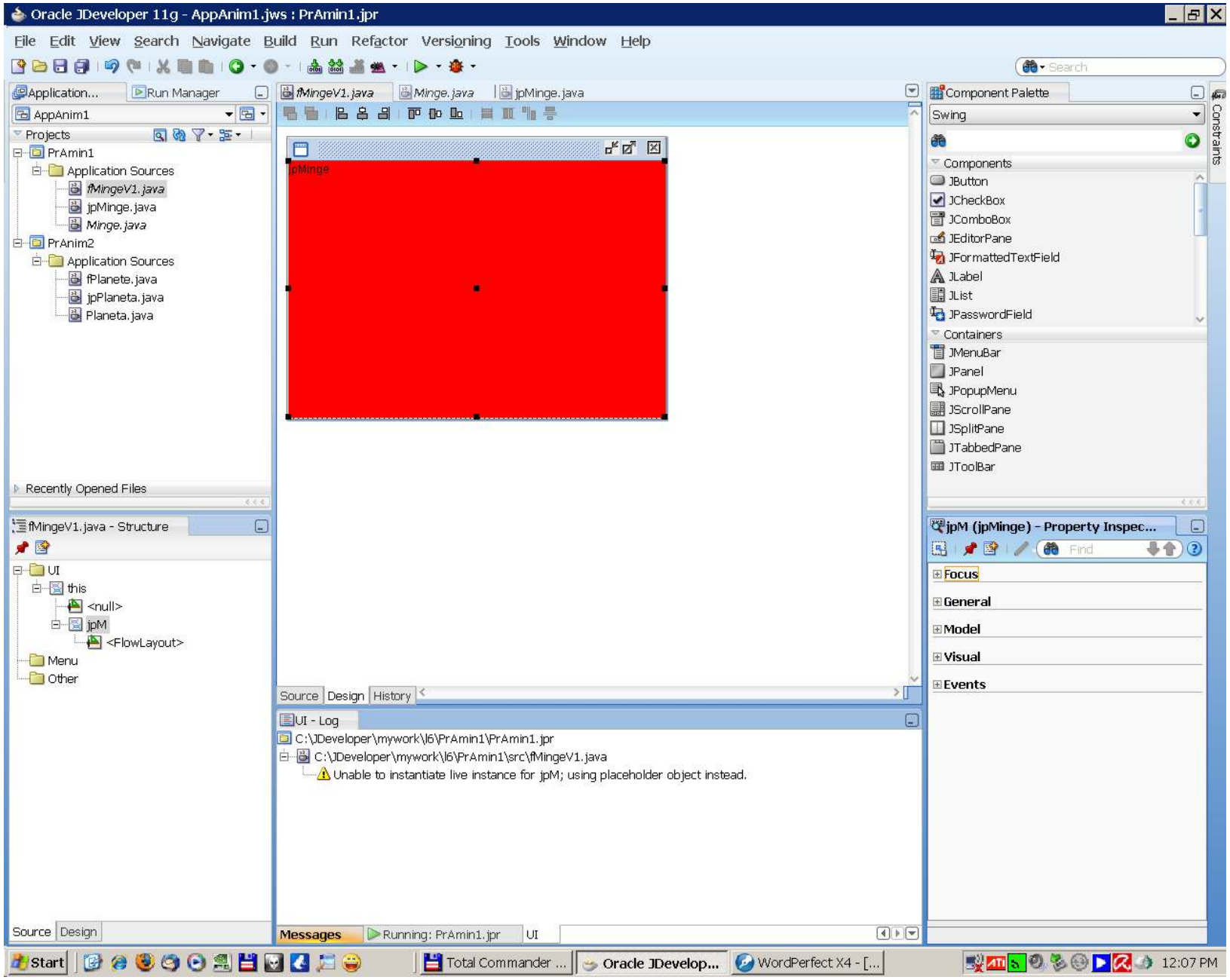
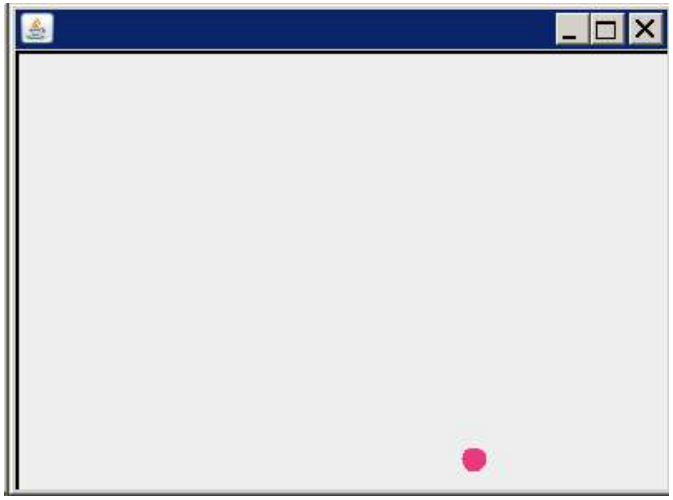

Oracle JDeveloper 11g - AppAnim1.jws : PrAmin1.jpr
File Edit View Search Navigate Build Run Refactor Versioning Tools Window Help
Application...
Run Manager
AppAnim1
Projects
PrAmin1
Application Sources
fMingeV1.java
jpMinge.java
Minge.java
PrAnim2
Application Sources
fPlanete.java
jpPlaneta.java
Planeta.java
Recently Opened Files
fMingeV1.java - Structure
UI
this
<null>
jpM
<FlowLayout>
Menu
Other
Source Design
fMingeV1.java
Minge.java
jpMinge.java
jpMinge
Source Design History
UI - Log
C:\JDeveloper\mywork\l6\PrAmin1\PrAmin1.jpr
C:\JDeveloper\mywork\l6\PrAmin1\src\fMingeV1.java
Unable to instantiate live instance for jpM; using placeholder object instead.
Messages
Running: PrAmin1.jpr
UI
Component Palette
Swing
Components
JButton
JCheckBox
JComboBox
JEditorPane
JFormattedTextField
JLabel
JList
JPasswordField
Containers
JMenuBar
JPanel
JPopupMenu
JScrollPane
JSplitPane
JTabbedPane
JToolBar
Constraints
jpM (jpMinge) - Property Inspec...
Find
Focus
General
Model
Visual
Events
Start
Total Commander ...
Oracle JDevelop...
WordPerfect X4 - [...
12:07 PM

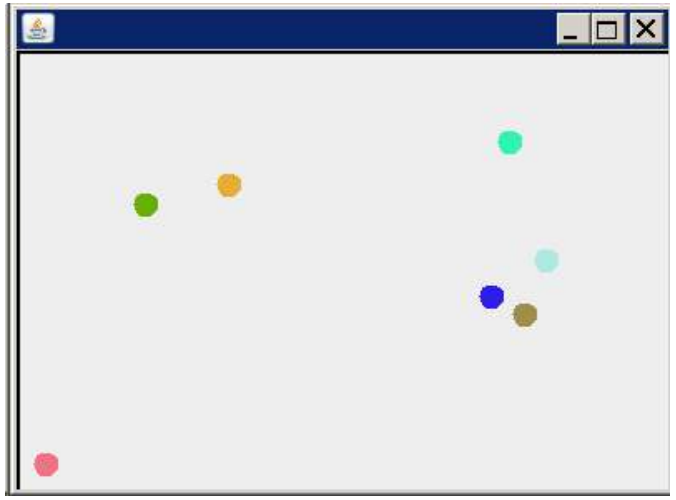
Clasa Minge.java:

```
import java.awt.Color;
import java.awt.Graphics;
import java.awt.geom.Ellipse2D;
import java.awt.geom.Rectangle2D;
import java.util.Random;

public class Minge {
   public double Px, Py; //pozitie minge
   public double Vx, Vy; //viteza minge
   public Color culoare;
   private static final int XSIZE = 15;
   private static final int YSIZE = 15;

   public Minge(int w, int h) {
      this.Px = aleator(w);
      this.Py = aleator(h);
      this.Vx = 2 - aleator(2); //1
      this.Vy = 2 - aleator(2); //1
      this.culoare = new Color(aleator(255) ,aleator(255), aleator(255));
   }
   public Minge(double Px, double Py) {
      this.Px = Px;
      this.Py = Py;
      this.Vx = 2 - aleator(2); //1
      this.Vy = 2 - aleator(2); //1
      this.culoare = new Color(aleator(255) ,aleator(255), aleator(255));
   }
   public int aleator(int i) {
      Random rnd = new Random();
      return rnd.nextInt(i);
   }
   public void move(Rectangle2D bounds)
     {
       Px += Vx;
       Py += Vy;
       if (Px < bounds.getMinX())
       {
         Px = bounds.getMinX();
         Vx = -Vx;
       }
       if (Px + XSIZE >= bounds.getMaxX())
       {
         Px = bounds.getMaxX() - XSIZE;
         Vx = -Vx;
       }
       if (Py < bounds.getMinY())
       {
         Py = bounds.getMinY();
         Vy = -Vy;
       }
       if (Py + YSIZE >= bounds.getMaxY())
       {
         Py = bounds.getMaxY() - YSIZE;
         Vy = -Vy;
       }
     }
   public Ellipse2D desenare(Graphics g)
   {
      g.setColor(culoare);
      return new Ellipse2D.Double(Px, Py, XSIZE, YSIZE);
   }
}
```

Clasa jpMinge.java:

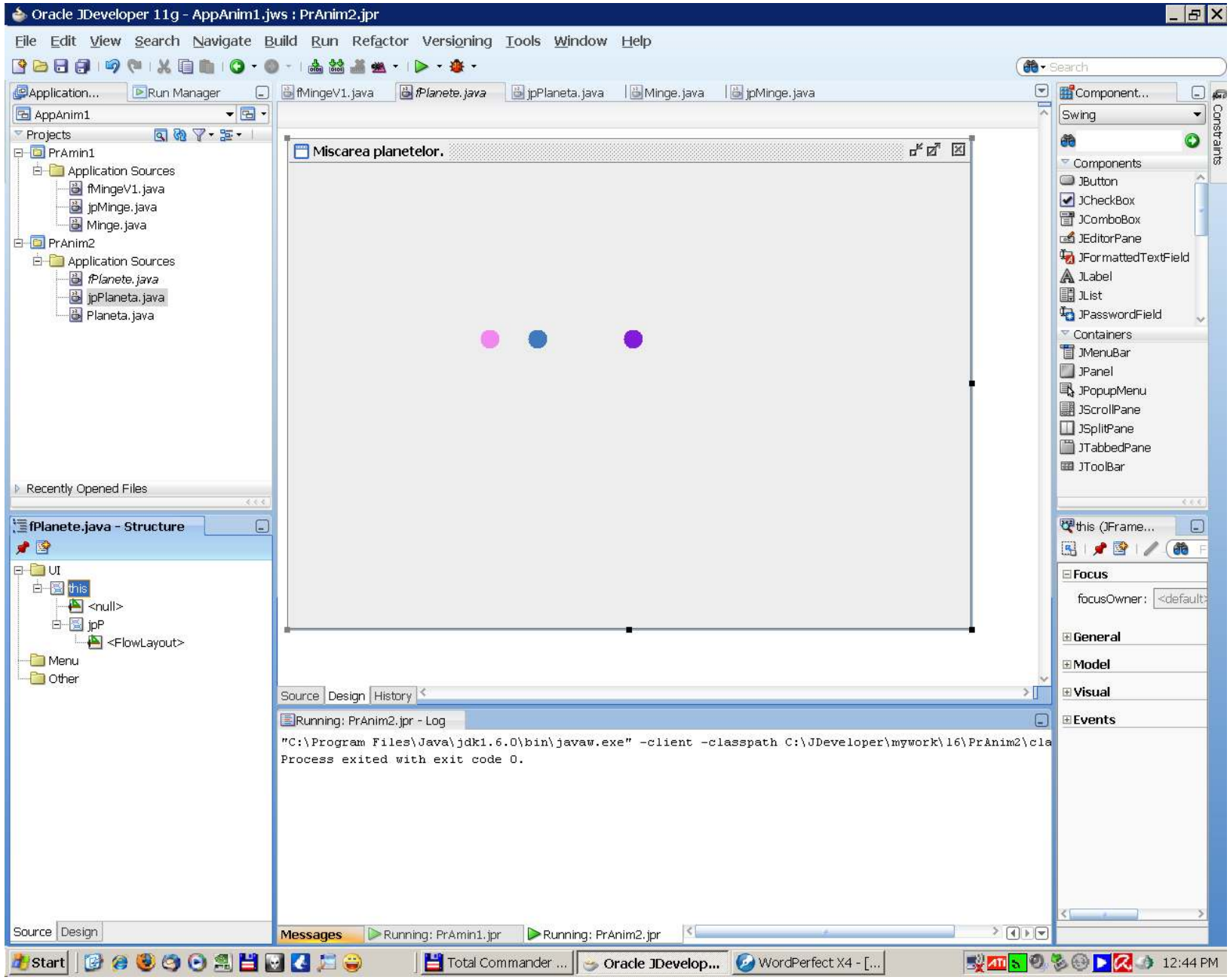
```
import java.awt.Graphics;
import java.awt.Graphics2D;
import javax.swing.JPanel;

public class jpMinge extends JPanel {
   public Minge b1;
   public Minge bile[] = new Minge[9]; //0-9
   public int NR_MINGI;

   public jpMinge(int n) {
      b1 = new Minge(100, 200);

      NR_MINGI = n - 1;
      try {
         for (int i = 0; i < NR_MINGI; ++i)
            bile[i] = new Minge(200, 200);
      } catch (ArrayIndexOutOfBoundsException e) {
         System.out.println("Numarul maxim de mingi poate fi 10!");
         //poate lucra cu cel mult 10 mingi
         System.exit(1);
      }
   }

   public void paintComponent(Graphics g) {
      super.paintComponent(g);
      Graphics2D g2 = (Graphics2D)g;

      //desenarea mingii b1
      g2.fill(b1.desenare(g2));

      //desenarea a NR_MINGI mingi
      for (int i = 0; i < NR_MINGI; ++i)
         g2.fill(bile[i].desenare(g2));
   }
}
```





Clasa fMingeV1.java:

```
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Rectangle;
import javax.swing.BorderFactory;
import javax.swing.JFrame;

public class fMingeV1 extends JFrame {
   public final int NRMINGI = 7;
   private jpMinge jpM = new jpMinge(NRMINGI);

   public fMingeV1() {
      try {
         jbInit();
      } catch (Exception e) {
         e.printStackTrace();
      }
   }

   private void jbInit() throws Exception {
      this.getContentPane().setLayout(null);
      this.setSize(new Dimension(400, 300));

      //da dimensiunil JPanel-ului
      jpM.setBounds(new Rectangle(0, 0, 395, 270));

      //il pune in coltul din stinga sus
      jpM.setLocation(0, 0);

      //ii trage chenarul
      jpM.setBorder(BorderFactory.createLineBorder(Color.black, 2));
      //il adauga la Frame
      this.getContentPane().add(jpM, null);
   }

   public void miscaMinge() {
      try {
         for (int i = 1; i <= 1000; i++) {
            jpM.b1.move(jpM.getBounds());

            for (int j = 0; j < jpM.NR_MINGI; ++j)
               jpM.bile[j].move(jpM.getBounds());
            /* normal, aici s-ar fi apelat repaint()
             * dar metoda miscaMinge() ocupa tot timpul
             * de executie si repaint() nu mai apuca sa
             * fie apelata
             */
            jpM.paint(jpM.getGraphics());
            Thread.sleep(10);
         }
      } catch (InterruptedException e) {
      }
   }

   public static void main(String[] args) {
      fMingeV1 frame = new fMingeV1();

      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      frame.setVisible(true);
      frame.miscaMinge();

   }
}
```


Oracle JDeveloper 11g - AppAnim1.jws : PrAnim2.jpr
File Edit View Search Navigate Build Run Refactor Versioning Tools Window Help
Application... Run Manager
AppAnim1
Projects
PrAmin1
Application Sources
fMingeV1.java
jpMinge.java
Minge.java
PrAnim2
Application Sources
fPlanete.java
jpPlaneta.java
Planeta.java
Recently Opened Files
fPlanete.java - Structure
UI
this
<null>
jpP
<FlowLayout>
Menu
Other
Source Design
fMingeV1.java
fPlanete.java
jpPlaneta.java
Minge.java
jpMinge.java
Miscarea planetelor.
Source Design History
Running: PrAnim2.jpr - Log
"C:\Program Files\Java\jdk1.6.0\bin\javaw.exe" -client -classpath C:\JDeveloper\mywork\16\PrAnim2\cla
Process exited with exit code 0.
Messages
Running: PrAmin1.jpr
Running: PrAnim2.jpr
Component...
Swing
Components
JButton
JCheckBox
JComboBox
JEditorPane
JFormattedTextField
JLabel
JList
JPasswordField
Containers
JMenuBar
JPanel
JPopupMenu
JScrollPane
JSplitPane
JTabbedPane
JToolBar
this (JFrame...
Focus
focusOwner:
General
Model
Visual
Events
Constraints
Start
Total Commander ...
Oracle JDevelop...
WordPerfect X4 - [...
12:44 PM

Clasa Planeta.java:

```
import java.awt.Color;
import java.awt.Graphics;
import java.awt.geom.Ellipse2D;
import java.util.Random;

class Planeta {
   public double Px, Py; //pozitia
   public double Vx, Vy; //viteza
   public double masa;
   public double raza;
   public Color culoare;

   private static final int XSIZE = 2;
   private static final int YSIZE = 2;

   public int aleator(int i) {
      Random rnd = new Random();
      return rnd.nextInt(i);
   }

   public Planeta(double x, double y, double vx, double vy, double masa,
            double raza) {
      this.Px = x;
      this.Py = y;
      this.Vx = vx;
      this.Vy = vy;
      this.masa = masa;
      this.raza = raza;
      this.culoare = new Color(aleator(255), aleator(255), aleator(255));
   }

   public void misca() {
      Px += Vx;
      Py += Vy;
   }

   public String toString() {
      return "P(" + Px + "," + Py + ")\n" +
         "V(" + Vx + "," + Vy + ")\n";
   }

   public Ellipse2D desenare(Graphics g) {
      g.setColor(culoare);
      return new Ellipse2D.Double(Px, Py, XSIZE * raza, YSIZE * raza);
   }
}
```

Clasa jpPlaneta.java:

```
import java.awt.Graphics;
import java.awt.Graphics2D;
import javax.swing.JPanel;

public class jpPlaneta extends JPanel {
   public Planeta planete[] = new Planeta[10];
   public int NrPlanete = 0;
   private static final double F_SCALA = 1000;


   public jpPlaneta() {
      /*
      NrPlanete = 2;
      planete[0] = new Planeta(100, 175, 0, -7, 50, 20) ;
      planete[1] = new Planeta(400, 250, 0, 7, 50, 10) ;
      */

      NrPlanete = 3;
      planete[0] = new Planeta(200, 175, 0, -0.5, 30, 10);
      planete[1] = new Planeta(250, 175, 0, 25, 1, 10);
      planete[2] = new Planeta(350, 175, 0, 13, 1, 10);

   }

   public void pozUrmat() {
      for (int i = 0; i < NrPlanete - 1; ++i)
         for (int j = i + 1; j < NrPlanete; ++j) {
            double dx = planete[i].Px - planete[j].Px;
            double dy = planete[i].Py - planete[j].Py;
            double d2 = dx * dx + dy * dy;
            double f = F_SCALA * planete[i].masa * planete[j].masa /
d2;
            double d = Math.sqrt(d2);

            double a_d = f / planete[i].masa / d;
            planete[i].Vx -= a_d * dx;
            planete[i].Vy -= a_d * dy;

            a_d = f / planete[j].masa / d;
            planete[j].Vx += a_d * dx;
            planete[j].Vy += a_d * dy;
         }
      /*
      for(int i=0;i<NrPlanete;++i)
         System.out.println(planete[i]);
      */
   }

   public void paintComponent(Graphics g) {
      super.paintComponent(g);
      Graphics2D g2 = (Graphics2D)g;

      for (int i = 0; i < NrPlanete; ++i)
         g2.fill(planete[i].desenare(g2));
   }
}
```

Clasa fPlanete.java:

```
import java.awt.Dimension;
import java.awt.Rectangle;
import javax.swing.JFrame;

public class fPlanete extends JFrame {
   private jpPlaneta jpP = new jpPlaneta();

   public fPlanete() {
      try {
         jbInit();
      } catch (Exception e) {
         e.printStackTrace();
      }
   }

   private void jbInit() throws Exception {
      this.getContentPane().setLayout(null);
      this.setSize(new Dimension(718, 518));
      this.setTitle("Miscarea planetelor.");
      jpP.setBounds(new Rectangle(5, 5, 720, 490));
      jpP.setLocation(0, 0);
      this.getContentPane().add(jpP, null);
   }

   public void miscaPlanete() {
      //pozitia initiala
      repaint();
      try {
         for (int i = 1; i <= 10000; i++) {
            jpP.pozUrmat();
            for (int j = 0; j < jpP.NrPlanete; ++j)
               jpP.planete[j].misca();
            jpP.paint(jpP.getGraphics());
            Thread.sleep(20);
         }
      } catch (InterruptedException e) {
      }
   }

   public static void main(String[] args) {
      fPlanete frame = new fPlanete();

      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      frame.setVisible(true);
      frame.miscaPlanete();

   }
}
```