```java
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class DirectionEffects extends JFrame {
    private final int DEFAULT_FRAME_WIDTH = 400;
    private final int DEFAULT_FRAME_HEIGHT = 400;
    private RectPanel panel;
    private JButton upButton;
    private JButton downButton;
    private JButton leftButton;
    private JButton rightButton;
    private JButton exitButton;
    public DirectionEffects() {

        //TODO 1: Add a button that should exit this application
        ActionListener list = new DirectionListener();
        setSize(400, 400);
        //construct components

        panel = new RectPanel();
        JPanel buttonPanel = new JPanel();

        ActionListener listener = new DirectionListener();
        //TODO 2: use anonymous class for action listener object exit button

        /* TODO 3: add two J Spinners under buttons that should set height and width
of the rectangle
            value shuold be min 20 and maximum width and height of frame.
            A new handler should be added as private class.
         */


        upButton = new JButton("Up");
        upButton.addActionListener(listener);

        downButton = new JButton("Down");
        downButton.addActionListener(listener);

        leftButton = new JButton("Left");
        leftButton.addActionListener(listener);

        rightButton = new JButton("Right");
        rightButton.addActionListener(listener);

        exitButton = new JButton("Exit");
        exitButton.addActionListener(list);

        // add components to content pane
        Container contentPane = getContentPane();
        contentPane.add(panel, "Center");

        buttonPanel.add(upButton);
        buttonPanel.add(downButton);
```

```
        buttonPanel.add(leftButton);
        buttonPanel.add(rightButton);
        buttonPanel.add(exitButton);
        this.setResizable(false);

        contentPane.add(buttonPanel, "South");
    }

    // inner class definition

    public static void main(String[] args) {
        DirectionEffects frame = new DirectionEffects();
        frame.setTitle("Direction Effects");
        frame.setVisible(true);
    }

    private class DirectionListener implements ActionListener {
        public void actionPerformed(ActionEvent event) {  // find the button that was
clicked

            Object source = event.getSource();

            if (source == upButton)
                panel.moveRectangle(0, -1);
            else if (source == downButton)
                panel.moveRectangle(0, 1);
            else if (source == leftButton)
                panel.moveRectangle(-1, 0);
            else if (source == rightButton)
                panel.moveRectangle(1, 0);
            else if (source == exitButton){
                System.exit(0);
                exitButton.addActionListener(this);
            }


               }
    }
}

class RectPanel extends JPanel {
    private final int RECT_WIDTH = 20;
    private final int RECT_HEIGHT = 20;
    private Rectangle rect;

    public RectPanel() {
        rect = new Rectangle(0, 0, RECT_WIDTH, RECT_HEIGHT);
    }

    public void paintComponent(Graphics g) {
        super.paintComponent(g);
        Graphics2D g2 = (Graphics2D) g;
        g2.draw(rect);
    }
```

```
    /**
     * Moves the rectangle and repaints it. The rectangle
     * is moved by multiples of its full width or height.
     *
     * @param dx the number of width units
     * @param dy the number of height units
     */
    public void moveRectangle(int dx, int dy) {

        //TODO 4:  Fix this code in such way that rectangle does not move out of
frame bounds

        rect.translate(dx * RECT_WIDTH, dy * RECT_HEIGHT);
        repaint();
    }
}
```