A类题目

——简易计算器

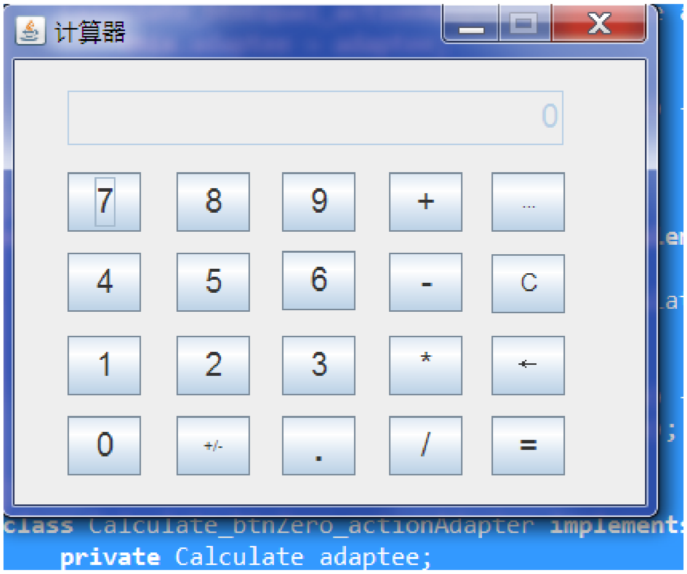
班级：信管151

姓名：牟江秀

学号：151961

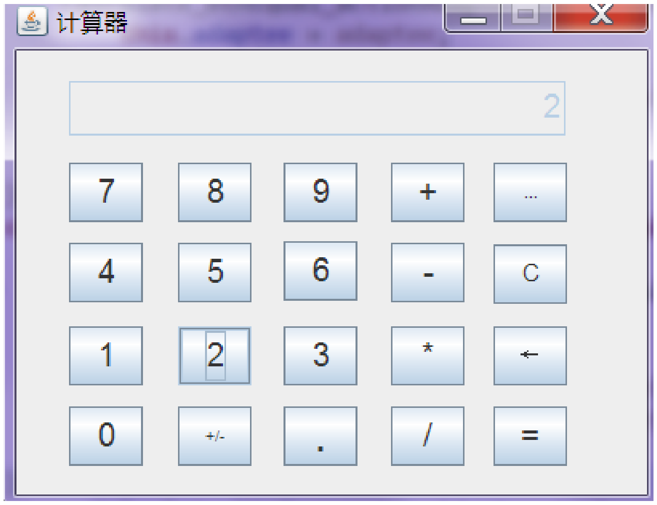
一、运行结果截图

计算器主界面

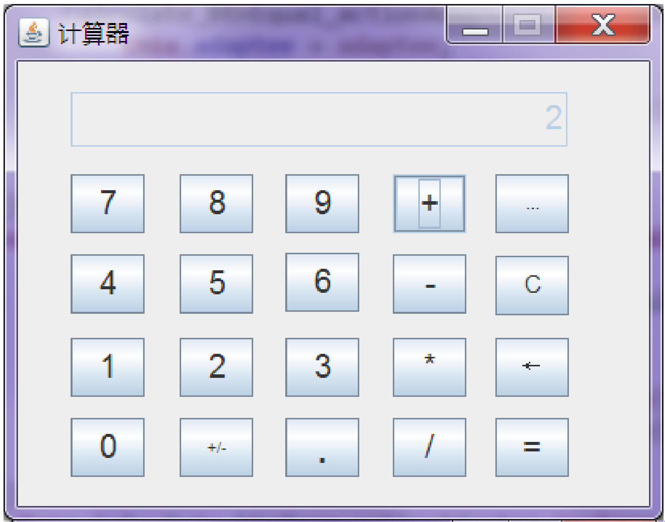


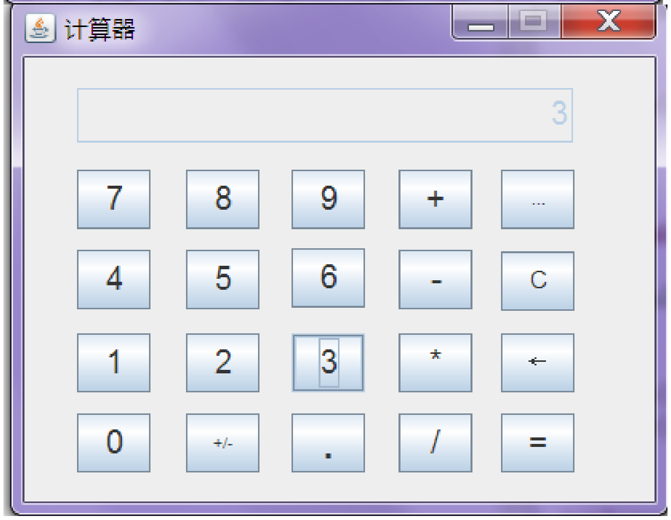
```
class Calculate_btnZero_actionAdapter implement
    private Calculate adaptee;
```

输入数字

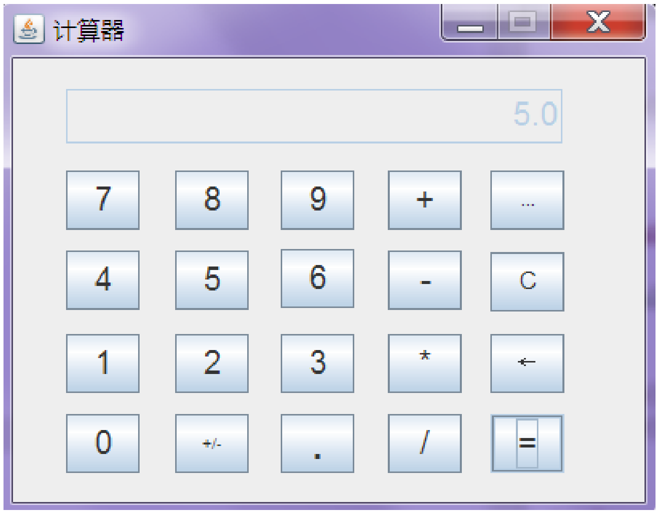


进行加减运算





得出结果



二、代码清单

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

public class Calculate extends JFrame {
    private String front = "", behind = ""; //分别用于记录加减乘除运算符之前,之后输入的内容
    private String op; //用于记录运算符
    private String re;//用于存储运算结果的字符串格式
    private boolean flag = false; //用于记录是否按下了运算符
    private boolean flag1 = false;//用于判断是否输入了点运算符
    private double result;//用于存储运算结果
    private boolean flag2 = false;//用于判断是否输入了数字
    private boolean flag3 = false;//用于判断是否按下了等号运算符

    JPanel contentPane;

    JTextField txtResult = new JTextField("0");
    JButton btnNull = new JButton("sqrt");
    JButton btnFour = new JButton("4");
    JButton btnFive = new JButton("5");
    JButton btnSix = new JButton("6");
    JButton btnDecrease = new JButton("-");
    JButton btnBegin = new JButton("C");
    JButton btnOne = new JButton("1");
    JButton btnTwo = new JButton("2");
    JButton btnThree = new JButton("3");
    JButton btnMultiply = new JButton("*");
    JButton btnCancel = new JButton("←");
    JButton btnZero = new JButton("0");
    JButton btnMinus = new JButton("+/-");
    JButton btnPoint = new JButton(".");
    JButton btnDivide = new JButton("/");
    JButton btnEqual = new JButton("=");
    JButton btnIncrease = new JButton("+");
    JButton btnSeven = new JButton("7");
    JButton btnEight = new JButton("8");
    JButton btnNine = new JButton("9");
public Calculate() {
      try {
           setDefaultCloseOperation(EXIT_ON_CLOSE);
           jbInit();
       } catch (Exception exception) {
           exception.printStackTrace();
               }
    }
    private void jbInit() throws Exception {
       contentPane = (JPanel) getContentPane();
       contentPane.setLayout(null);
```

```
        this.setResizable(false);
        setSize(new Dimension(400, 300));
        setTitle("计算器");
        txtResult.setEnabled(false);
        txtResult.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));
        txtResult.setEditable(false);

        txtResult.setHorizontalAlignment(SwingConstants.RIGHT);
        txtResult.setBounds(new Rectangle(33, 19, 310, 34));
        btnNull.setBounds(new Rectangle(298, 70, 46, 37));
        btnNull.setFont(new java.awt.Font("Dialog", Font.PLAIN, 12));

     //btnNull.addActionListener(new
FrameCalculate_btnNull_actionAdapter(this));
        btnFour.setBounds(new Rectangle(33, 120, 46, 37));
        btnFour.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnFive.setBounds(new Rectangle(101, 120, 46, 37));
        btnFive.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnSix.setBounds(new Rectangle(167, 119, 46, 37));
        btnSix.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnDecrease.setBounds(new Rectangle(234, 120, 46, 37));
        btnDecrease.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnBegin.setBounds(new Rectangle(298, 121, 46, 37));
        btnBegin.setFont(new java.awt.Font("Dialog", Font.PLAIN, 15));

        btnBegin.addActionListener(new Calculate_btnBegin_actionAdapter(this));
        btnOne.setBounds(new Rectangle(33, 172, 46, 37));
        btnOne.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnTwo.setBounds(new Rectangle(101, 172, 46, 37));
        btnTwo.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnThree.setBounds(new Rectangle(167, 172, 46, 37));
        btnThree.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnMultiply.setBounds(new Rectangle(234, 172, 46, 37));
        btnMultiply.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        btnCancel.setBounds(new Rectangle(298, 172, 46, 37));
        btnCancel.setFont(new java.awt.Font("Dialog", Font.PLAIN, 12));

       btnCancel.addActionListener(new
Calculate_btnCancel_actionAdapter(this));
        btnZero.setBounds(new Rectangle(33, 222, 46, 37));
        btnZero.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

        //加载数字0-9的监听事件
        btnZero.addActionListener(new Calculate_btnZero_actionAdapter(this));
        btnOne.addActionListener(new Calculate_btnZero_actionAdapter(this));
        btnTwo.addActionListener(new Calculate_btnZero_actionAdapter(this));
        btnThree.addActionListener(new Calculate_btnZero_actionAdapter(this));
```

```
btnFour.addActionListener(new Calculate_btnZero_actionAdapter(this));
btnFive.addActionListener(new Calculate_btnZero_actionAdapter(this));
btnSix.addActionListener(new Calculate_btnZero_actionAdapter(this));
btnSeven.addActionListener(new Calculate_btnZero_actionAdapter(this));
btnEight.addActionListener(new Calculate_btnZero_actionAdapter(this));
btnNine.addActionListener(new Calculate_btnZero_actionAdapter(this));
btnMinus.setBounds(new Rectangle(101, 222, 46, 37));
btnMinus.setFont(new java.awt.Font("Dialog", Font.PLAIN, 10));

btnMinus.addActionListener(new Calculate_btnMinus_actionAdapter(this));
btnPoint.setBounds(new Rectangle(167, 222, 46, 37));
btnPoint.setFont(new java.awt.Font("Dialog", Font.PLAIN, 30));
btnPoint.setHorizontalTextPosition(SwingConstants.CENTER);

btnPoint.addActionListener(new Calculate_btnPoint_actionAdapter(this));
btnDivide.setBounds(new Rectangle(234, 222, 46, 37));
btnDivide.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

btnEqual.setBounds(new Rectangle(298, 222, 46, 37));
btnEqual.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

btnEqual.addActionListener(new Calculate_btnEqual_actionAdapter(this));
btnIncrease.setBounds(new Rectangle(234, 70, 46, 37));
btnIncrease.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

//加载加减乘除运算符的监听事件
btnIncrease.addActionListener(new
                          Calculate_btnIncrease_actionAdapter(this));
btnDecrease.addActionListener(new
                           Calculate_btnIncrease_actionAdapter(this));
btnMultiply.addActionListener(new
                          Calculate_btnIncrease_actionAdapter(this));
btnDivide.addActionListener(new
                         Calculate_btnIncrease_actionAdapter(this));
btnSeven.setBounds(new Rectangle(33, 70, 46, 37));
btnSeven.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

btnEight.setBounds(new Rectangle(101, 70, 46, 37));
btnEight.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

btnNine.setBounds(new Rectangle(167, 70, 46, 37));
btnNine.setFont(new java.awt.Font("Dialog", Font.PLAIN, 20));

contentPane.add(btnFive);
contentPane.add(btnSix);
contentPane.add(btnDecrease);
contentPane.add(btnBegin);
contentPane.add(btnOne);
contentPane.add(btnTwo);
contentPane.add(btnThree);
contentPane.add(btnMultiply);
contentPane.add(btnCancel);
contentPane.add(btnMinus);
contentPane.add(btnPoint);
contentPane.add(btnDivide);
```

```
        contentPane.add(btnEqual);
        contentPane.add(btnEight);
        contentPane.add(btnNine);
        contentPane.add(btnFour);
        contentPane.add(btnSeven);
        contentPane.add(btnIncrease);
        contentPane.add(btnNull);
        contentPane.add(txtResult);
        contentPane.add(btnZero);
    }
public void btnZero_actionPerformed(ActionEvent e) {
        if (flag) { //如果刚刚按下了运算符
            txtResult.setText("");
            if (flag1) {//判断之前是否输入了点运算符
                txtResult.setText("0." + e.getActionCommand());
                flag1 = false;
            } else {
                txtResult.setText(e.getActionCommand());
            }
            flag2 = true;
        } else {
            int num = txtResult.getText().indexOf(".");
            if (num < 0 && !txtResult.getText().equals("0")) {
                txtResult.setText(txtResult.getText() + e.getActionCommand());
            } else if (num < 0 && txtResult.getText().equals("0")) {
                txtResult.setText(e.getActionCommand());
            } else if (num >= 0 && txtResult.getText().equals("0")) {
                txtResult.setText("0." + e.getActionCommand());
            } else if (num >= 0 && !txtResult.getText().equals("0")) {
                txtResult.setText(txtResult.getText() + e.getActionCommand());
            }
        }
        flag = false;
        flag3=false;
    }
    public void btnIncrease_actionPerformed(ActionEvent e) {
        if(flag3){
            txtResult.setText(txtResult.getText());
            op = e.getActionCommand(); //得到刚刚按下的运算符
            front = txtResult.getText(); //记录加减乘除运算符之前输入的内容
        }
        else if (flag2) {
            ActionEvent ee = new ActionEvent("qq", 1, "pp");
            btnEqual_actionPerformed(ee);
            op = e.getActionCommand(); //得到刚刚按下的运算符
            front = re;
            flag2 = false;
        } else {
            front = txtResult.getText(); //记录加减乘除运算符之前输入的内容
            op = e.getActionCommand(); //得到刚刚按下的运算符
        }
        flag3=false;
        flag = true; //记录已经按下了加减乘除运算符的其中一个
    }
```

```
public void btnEqual_actionPerformed(ActionEvent e) {
    if(!flag3)//未曾按下等于运算符
        behind = txtResult.getText();
    else
        front = re;
    try {
        double a1 = Double.parseDouble(front);
        double b1 = Double.parseDouble(behind);
        if (op == "+") {
            result = a1 + b1;
        } else if (op == "-") {
            result = a1 - b1;
        } else if (op == "*") {
            result = a1 * b1;
        } else {
            result = a1 / b1;
        }
        Double r = new Double(result);
        re = r.toString(result);
        txtResult.setText(re);
        } catch (ArithmeticException ce) {
            txtResult.setText("除数不能为零");
        } catch (Exception ee) {
        }
        if (!flag3)
            flag3 = true;
}
public void btnPoint_actionPerformed(ActionEvent e) {
    int num=txtResult.getText().indexOf(".");
    if(num<0 && !flag)
        txtResult.setText(txtResult.getText()+e.getActionCommand());
    if(flag)
        flag1=true;
}
public void btnBegin_actionPerformed(ActionEvent e) {//清零运算符事件处理
    flag=false;
    flag1=false;
    flag2=false;
    flag3=false;
    front="";
    behind="";
    re="";
    txtResult.setText("0");
}
public void btnMinus_actionPerformed(ActionEvent e) {//取反运算符事件处理
    if(txtResult.getText().equals("0")){//如果文本框内容为0
        txtResult.setText(txtResult.getText());
    }else if(txtResult.getText().indexOf("-")>=0){//若文本框中含有负号
        String a=txtResult.getText().replaceAll("-","");
        txtResult.setText(a);
    }else if(flag){
        txtResult.setText("0");
    }else{
        txtResult.setText("-"+txtResult.getText());
        }
    }
    public void btnCancel_actionPerformed(ActionEvent e) {//退格事件处理方法
        String str=txtResult.getText();
        if(str.length() == 1){//如文本框中只剩下最后一个字符,将文本框内容置为0
            txtResult.setText("0");
        }
        if(str.length()>1){
            str=str.substring(0,str.length()-1);
            txtResult.setText(str);
        }
    }
    public static void main(String[] args){
        Calculate fc = new Calculate();
        fc.setSize(400,310);
        fc.setLocation(200,150);
        fc.setVisible(true);
    }
}
class Calculate_btnCancel_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnCancel_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnCancel_actionPerformed(e);
    }
}
class Calculate_btnMinus_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnMinus_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnMinus_actionPerformed(e);
    }
}
class Calculate_btnBegin_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnBegin_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnBegin_actionPerformed(e);
    }
}
class Calculate_btnPoint_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnPoint_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnPoint_actionPerformed(e);
    }
}
```

```
class Calculate_btnEqual_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnEqual_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnEqual_actionPerformed(e);
    }
}
class Calculate_btnIncrease_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnIncrease_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnIncrease_actionPerformed(e);
    }
}
class Calculate_btnZero_actionAdapter implements ActionListener {
    private Calculate adaptee;
    Calculate_btnZero_actionAdapter(Calculate adaptee) {
        this.adaptee = adaptee;
    }
    public void actionPerformed(ActionEvent e) {
        adaptee.btnZero_actionPerformed(e);
    }
}
```