

实验六 异常处理与文本 I/O

学生姓名: 黄晨箬 学 号: 6109119066 专业班级: 计算机 193 班
实验类型: ☒ 验证 ☐ 综合 ☐ 设计 ☐ 创新 实验日期: 2021.6.3 实验成绩: _____

一、实验目的

- 1、掌握Java中异常处理方法，使用语句声明、抛出和捕获异常；
- 2、掌握Java中自定义异常类的定义及使用；
- 3、掌握Java中对文本文件的读写数据方法。

二、实验内容

习题1（检查危险品） 车站检查危险品的设备，如果发现危险品会发出警告。
编程模拟设备发现危险品。

编写一个Exception 的子类DangerException，该子类可以创建异常对象，该异常对象调用toShow()方法输出“属于危险品”。

编写一个Machine 类，该类的方法checkBag(Goods goods)当发现参数goods 是危险品时（goods 的isDanger 属性是true）将抛出DangerException 异常。

程序在主类的main()方法中的try-catch 语句的try 部分让Machine 类的实例调用checkBag (Goods goods)方法，如果发现危险品就在try-catch 语句的catch 部分处理危险品。

程序运行结果参考如下：

```
苹果不是危险品！苹果检查通过  
危险品！炸药被禁止！  
西服不是危险品！西服检查通过  
危险品！硫酸被禁止！  
手表不是危险品！手表检查通过  
危险品！硫磺被禁止！
```

习题2（星期输出） 要求设计一个GUI图形窗口程序，该程序让用户输入一个星期中的任意一天的数字1到7，然后输出该数字对应的星期几，程序的运行效果参考如图5-1所示，但是，当用户输入的数字不在1到7范围内时，如图5-2所示，程序应该弹出一个对话框以显示发生了异常（图5-3）。

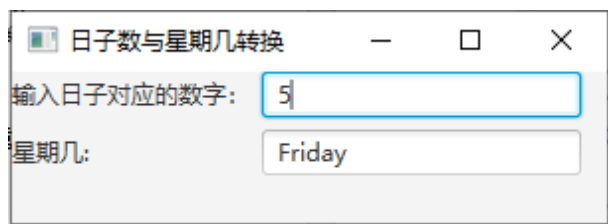


图5-1

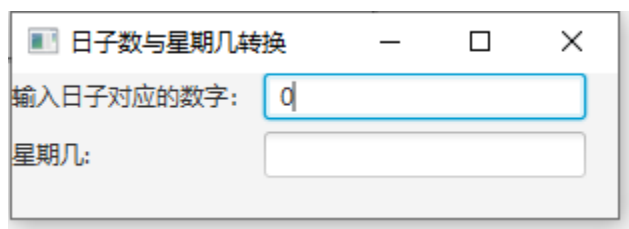


图5-2

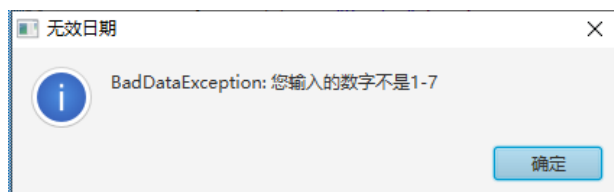


图5-3

习题 3（计算器） 编程实现，定义一个计算器应用程序，使之实现加、减、乘、除运算。要求：使用 GUI 界面实现数据的输入，如果输入的信息不是数据，要求提示用户“输入信息有误！”；在进行除法运算时，输入的除数为 0 时，要求：提示用户“输入的除数为 0，无法进行除法运算！”

习题 4（分析成绩单） 现在有如下格式的成绩单（文本格式）score.txt：

姓名:张三，数学 72 分，物理 67 分，英语 70 分.

姓名:李四，数学 92 分，物理 98 分，英语 88 分.

姓名:周五，数学 68 分，物理 80 分，英语 77 分.

要求编写程序按行读入取成绩单，并在该行的后面尾加上该同学的总成绩，然后再将该行写入到一个名字为 socreAnalysis.txt 的文件中。

运行效果如图所示：

```
姓名 张三, 数学72分, 物理67分, 英语70分, 总分: 209.0  
姓名 李四, 数学92分, 物理98分, 英语88分, 总分: 278.0  
姓名 周五, 数学68分, 物理80分, 英语77分, 总分: 225.0
```

三、实验要求

- 1、要求学生在实验前一定要非常清楚并灵活运用涉及章所讲过的内容；
- 2、在上机实验前编写好实验内容要求的程序，以便上机实验时调试、修改等。
- 3、上机实验后，每位学生必须对该次实验内容写一份实验报告，包括程序内容、调试过程、运行情况和结果等。

四、实验环境

- 1、PC微机；
- 2、DOS操作系统或 Windows 操作系统；
- 3、Eclipse程序集成环境。

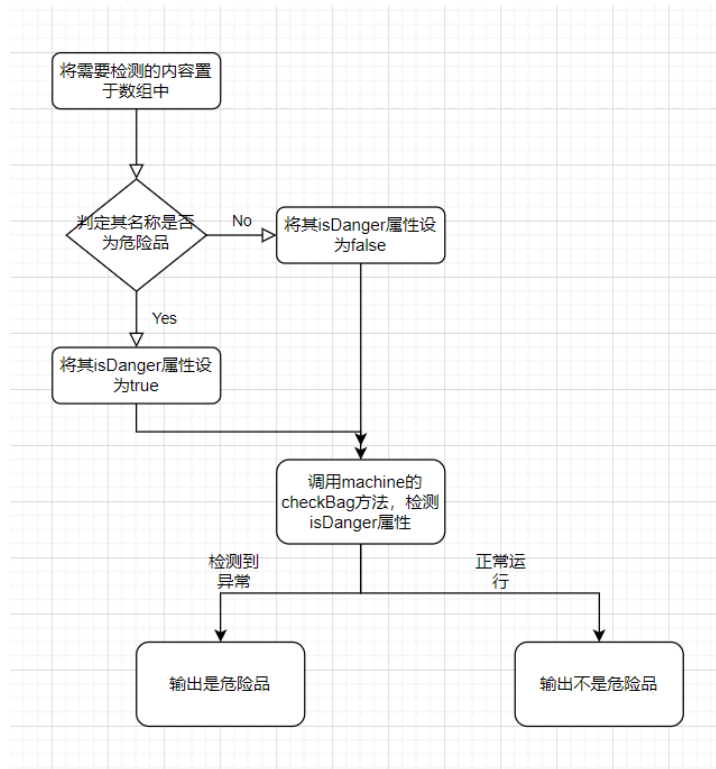
五、实验步骤

- 1、根据题目要求，画出程序流程图；
- 2、给出本程序的java数据结构；
- 3、编写出程序；
- 4、调试程序：给出一些测试数据，检查输出结果。

六、实验数据及处理结果

习题一：

[程序流程图]



[数据结构设计]

```
/*
    按照要求，分别建立各个类。
    对于未介绍的Goods类，需要包含构造方法，命名与取名的方法，判定是否危险的方法以及输出是否危险的方法。
    利用数组，将需要检查的内容放入数组里，依次进行检验。
    在Machine中，设定发现是danger时抛出错误，否则按要求输出。
    使用try结构，当检测到异常时，调用toShow方法输出结果，否则正常输出。
*/
```

[程序源代码]

```

public static class Goods {
    String name;
    boolean isDanger;

    Goods(boolean isDanger, String name){
        this.name = name;
        this.isDanger = isDanger;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getName() {
        return name;
    }

    public void setIsDanger(boolean b){
        isDanger = b;
    }

    public boolean isDanger(){
        return isDanger;
    }
}

```

```

static class DangerException extends Exception {
    void toShow() {
        System.out.print("危险品! ");
    }
}

static class Machine {
    public void checkBag(Goods goods) throws DangerException {
        if(goods.isDanger()) {
            throw new DangerException();
        }
        else
            System.out.println(goods.getName()+"不是危险品!"+goods.getName()+"检查通过");
    }
}

```

```

public static void main (String[] args){
    Machine machine = new Machine();
    Goods goods = new Goods( isDanger: false, name: null);
    String[] arr ={"苹果","炸药","西服","硫酸","手表","硫磺"};
    for (int i = 0; i < arr.length; i++) {
        goods.setName(arr[i]);
        if (arr[i].equals("硫酸") || arr[i].equals("炸药") || arr[i].equals("硫磺")) {
            goods.setIsDanger(true);
        }
        else {
            goods.setIsDanger(false);
        }

        try {
            machine.checkBag(goods);
        } catch (DangerException e) {
            eToShow();
            System.out.println(goods.getName() + "被禁止!");
        }
    }
}

```

[程序运行结果]

苹果不是危险品!苹果检查通过
危险品! 炸药被禁止!
西服不是危险品!西服检查通过
危险品! 硫酸被禁止!
手表不是危险品!手表检查通过
危险品! 硫磺被禁止!

习题二:

[程序流程图]

省略

[数据结构设计]

```
    /*  
        在BadDataDay中绘制窗体,  
        然后再actionPerformed方法中设置输入后调用getDayName方法获取输入之后相对应的星期,  
        当超出范围时抛出异常。  
    */
```

[程序源代码]

```
public class BadDataDays extends JFrame implements ActionListener {  
    private JTextField dayNumberField, dayNameField;  
    private int dayNumber;  
    private String dayName;  
  
    public BadDataDays() {  
        super( title: "日子数与星期几转换");  
        Container container = getContentPane();  
        container.setLayout(new GridLayout( rows: 2, cols: 2));  
        container.add(new JLabel( text: "输入数字", SwingConstants.LEFT));  
        dayNumberField = new JTextField();  
        container.add(dayNumberField);  
        dayNumberField.addActionListener( !: this);  
        container.add(new JLabel( text: "星期几", SwingConstants.LEFT));  
        dayNameField = new JTextField( columns: 10);  
        container.add(dayNameField);  
        setSize( width: 200, height: 100);  
        setVisible(true);  
    }  
}
```

```

public String getDayName(int dayNumber) throws BadDataException {
    switch(dayNumber) {
        case 7: return "Sunday";
        case 1: return "January";
        case 2: return "Tuesday";
        case 3: return "Wednesday";
        case 4: return "Thursday";
        case 5: return "Friday";
        case 6: return "Saturday";
        default: throw new BadDataException("您输入的数字不是1-7");
    }
}

static class BadDataException extends RuntimeException {
    private static final long serialVersionUID = 1L;
    public BadDataException() {
        super();
    }
    public BadDataException(String message) {
        super(message);
    }
}
}

```

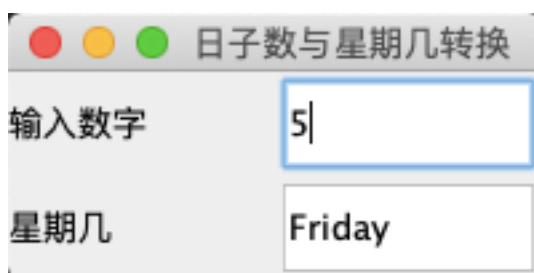
```

public void actionPerformed(ActionEvent event) {
    dayNameField.setText("");
    try {
        dayNumber = Integer.parseInt(dayNumberField.getText());
        dayName = getDayName(dayNumber);
        dayNameField.setText(dayName);
    }
    catch (BadDataException e) {
        JOptionPane.showMessageDialog(parentComponent: BadDataDays.this, e.toString(), "无效日期", JOptionPane.WARNING_MESSAGE);
    }
    catch (NumberFormatException ex) {
        System.out.println("I detected Exception" + ex.toString());
    }
}

public static void main(String[] args) {
    BadDataDays application = new BadDataDays();
    application.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}
}

```

[程序运行结果]



习题三:

[程序流程图]

省略

[数据结构设计]

省略

[程序源代码]

```
public void init() {  
    f = new JFrame( title: "CardLayout testing");  
    Container ct = f.getContentPane();  
    tf.setHorizontalAlignment(JTextField.RIGHT);  
    ct.add(tf, constraints: "North");  
    JPanel p2 = new JPanel();  
    ct.add(p2, constraints: "Center");  
    GridLayout c = new GridLayout( rows: 4, cols: 4);  
    p2.setLayout(c);  
    JButton b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b11, b12, b13, b14, b15, b16;
```

```
    b1 = new JButton( text: "1");  
    b2 = new JButton( text: "2");  
    b3 = new JButton( text: "3");  
    b4 = new JButton( text: "4");  
    b5 = new JButton( text: "5");  
    b6 = new JButton( text: "6");  
    b7 = new JButton( text: "7");  
    b8 = new JButton( text: "8");  
    b9 = new JButton( text: "9");  
    b10 = new JButton( text: "0");  
    b11 = new JButton( text: "." );  
    b12 = new JButton( text: "=" );  
    b13 = new JButton( text: "+" );  
    b14 = new JButton( text: "-" );  
    b15 = new JButton( text: "x" );  
    b16 = new JButton( text: "/" );  
    p2.add(b1);  
    b1.addActionListener( l: this);  
    p2.add(b2);  
    b2.addActionListener( l: this);  
    p2.add(b3);  
    b3.addActionListener( l: this);  
    p2.add(b13);  
    b13.addActionListener( l: this);  
    p2.add(b4);  
    b4.addActionListener( l: this);  
    p2.add(b5);  
    b5.addActionListener( l: this);  
    p2.add(b6);
```

```

p2.add(b14);
b14.addActionListener( B this);
p2.add(b7);
b7.addActionListener( B this);
p2.add(b8);
b8.addActionListener( B this);
p2.add(b9);
b9.addActionListener( B this);
p2.add(b15);
b15.addActionListener( B this);
p2.add(b10);
b10.addActionListener( B this);
p2.add(b11);
b11.addActionListener( B this);
p2.add(b12);
b12.addActionListener( B this);
p2.add(b16);
b16.addActionListener( B this);

f.addWindowListener(new WindowAdapter() {
    public void windowClosing(WindowEvent e) {
        System.exit( status: 0);
    }
});
f.setSize( width: 200, height: 180);
f.setVisible(true);
}

```

```

double sum = 0;
Vector v = new Vector( initialCapacity: 1, capacityIncrement: 1);
Vector v2 = new Vector( initialCapacity: 1, capacityIncrement: 1);

public Calculator() {
    v2.addElement(new String( original: "init"));
    v.addElement(new String( original: "init"));
}

public void actionPerformed(ActionEvent e) {
    String str = tf.getText();
    // 加号的实现
    if (e.getActionCommand().equals("+"))
    {
        if ("+".equals((String) v2.lastElement())
            || ("-").equals((String) v2.lastElement())
            || ("x").equals((String) v2.lastElement())
            || ("/").equals((String) v2.lastElement())
            || ("=").equals((String) v2.lastElement())) {
            tf.setText(String.valueOf(sum));
        } else if ("init".equals((String) v2.lastElement())) {
            tf.setText(String.valueOf(sum));
            v2.addElement(new String( original: "="));
        } else {
            double d = Double.parseDouble(str);
            if ("+".equals((String) v.lastElement())) {
                sum = sum + d;
                tf.setText(String.valueOf(sum));
            } else if ("-".equals((String) v.lastElement())) {
                sum = sum - d;
                tf.setText(String.valueOf(sum));
            }
        }
    }
}

```



```

    } else if (("x").equals((String) v.lastElement())) {
        sum = sum * d;
        tf.setText(String.valueOf(sum));
    } else if ("/".equals((String) v.lastElement())) {
        sum = sum / d;
        tf.setText(String.valueOf(sum));
    } else if ("=".equals((String) v.lastElement())) {
        sum = sum + d;
        tf.setText(String.valueOf(sum));
    } else {
        sum = sum + d;
        tf.setText(String.valueOf(sum));
    }
}

v.addElement(new String( original: "+"));
v2.addElement(new String( original: "+"));
}

```

```

// 减号的实现
if (e.getActionCommand().equals("-"))
{
    if (("+" ).equals((String) v2.lastElement())
        || ("-").equals((String) v2.lastElement())
        || ("x").equals((String) v2.lastElement())
        || ("/").equals((String) v2.lastElement())
        || ("=").equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
    } else if (("init").equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
        v2.addElement(new String( original: "="));
    } else {

```

```

        double d = Double.parseDouble(str);
        if (("+" ).equals((String) v.lastElement())) {
            sum = sum + d;
            tf.setText(String.valueOf(sum));
        } else if ("-".equals((String) v.lastElement())) {
            sum = sum - d;
            tf.setText(String.valueOf(sum));
        } else if ("x".equals((String) v.lastElement())) {
            sum = sum * d;
            tf.setText(String.valueOf(sum));
        } else if ("/".equals((String) v.lastElement())) {
            sum = sum / d;
            tf.setText(String.valueOf(sum));
        } else if ("=".equals((String) v.lastElement())) {
            sum = sum + d;
            tf.setText(String.valueOf(sum));
        } else {
            sum = sum + d;
            tf.setText(String.valueOf(sum));
        }
    }

    v.addElement(new String( original: "-"));
    v2.addElement(new String( original: "-"));
}

```

```
// 乘号的实现
if (e.getActionCommand().equals("x"))
{
    if (("+".equals((String) v2.lastElement())
        || ("-").equals((String) v2.lastElement())
        || ("x").equals((String) v2.lastElement())
        || ("/").equals((String) v2.lastElement())
        || ("=").equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
    } else if (("init").equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
        v2.addElement(new String( original: "="));
    } else {
        double d = Double.parseDouble(str);
        if (("+".equals((String) v.lastElement())) {
            sum = sum + d;
            tf.setText(String.valueOf(sum));
        } else if (( "-").equals((String) v.lastElement())) {
            sum = sum - d;
            tf.setText(String.valueOf(sum));
        } else if (( "x").equals((String) v.lastElement())) {
            sum = sum * d;
            tf.setText(String.valueOf(sum));
        } else if (( "/" ).equals((String) v.lastElement())) {
            sum = sum / d;
            tf.setText(String.valueOf(sum));
        } else if (( "=").equals((String) v.lastElement())) {
            sum = sum + d;
            tf.setText(String.valueOf(sum));
        } else {
            sum = sum + d;

```

```
            tf.setText(String.valueOf(sum));
        }
    }
    v.addElement(new String( original: "x"));
    v2.addElement(new String( original: "x"));
}

// 除号的实现
if (e.getActionCommand().equals("/"))
{
    if (("+".equals((String) v2.lastElement())
        || ("-").equals((String) v2.lastElement())
        || ("x").equals((String) v2.lastElement())
        || ("/").equals((String) v2.lastElement())
        || ("=").equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
    } else if (("init").equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
        v2.addElement(new String( original: "="));
    } else {
        double d = Double.parseDouble(str);
        if (("+".equals((String) v.lastElement())) {
            sum = sum + d;
            tf.setText(String.valueOf(sum));
        } else if (( "-").equals((String) v.lastElement())) {
            sum = sum - d;
            tf.setText(String.valueOf(sum));
        } else if (( "x").equals((String) v.lastElement())) {
            sum = sum * d;
            tf.setText(String.valueOf(sum));
        } else if (( "/" ).equals((String) v.lastElement())) {

```

```

        tf.setText(String.valueOf(sum));
    } else if ("/".equals((String) v.lastElement())) {
        sum = sum / d;
        tf.setText(String.valueOf(sum));
    } else if ("=".equals((String) v.lastElement())) {
        sum = sum + d;
        tf.setText(String.valueOf(sum));
    } else {
        sum = sum + d;
        tf.setText(String.valueOf(sum));
    }
}
v.addElement(new String( original: "/"));
v2.addElement(new String( original: "/"));
}

// 等号实现
if (e.getActionCommand() == "=")
{
    if ("+".equals((String) v2.lastElement())
        || "-".equals((String) v2.lastElement())
        || "x".equals((String) v2.lastElement())
        || "/".equals((String) v2.lastElement())
        || "=".equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
    } else if ("init".equals((String) v2.lastElement())) {
        tf.setText(String.valueOf(sum));
    } else {
        double d = Double.parseDouble(str);
        if ("+".equals((String) v.lastElement())) {
            sum = sum + d;

```

```

        tf.setText(String.valueOf(sum));
    } else if ("-".equals((String) v.lastElement())) {
        sum = sum - d;
        tf.setText(String.valueOf(sum));
    } else if ("x".equals((String) v.lastElement())) {
        sum = sum * d;
        tf.setText(String.valueOf(sum));
    } else if ("/".equals((String) v.lastElement())) {
        sum = sum / d;
        tf.setText(String.valueOf(sum));
    } else if ("=".equals((String) v.lastElement())) {
        sum = sum + d;
        tf.setText(String.valueOf(sum));
    } else {
        sum = sum + d;
        tf.setText(String.valueOf(sum));
    }
}
v.addElement(new String( original: "="));
v2.addElement(new String( original: "="));
}

if (e.getActionCommand() == "1" || e.getActionCommand() == "2"
    || e.getActionCommand() == "3" || e.getActionCommand() == "4"
    || e.getActionCommand() == "5" || e.getActionCommand() == "6"
    || e.getActionCommand() == "7" || e.getActionCommand() == "8"
    || e.getActionCommand() == "9" || e.getActionCommand() == "0"
    || e.getActionCommand() == ".")
{

```

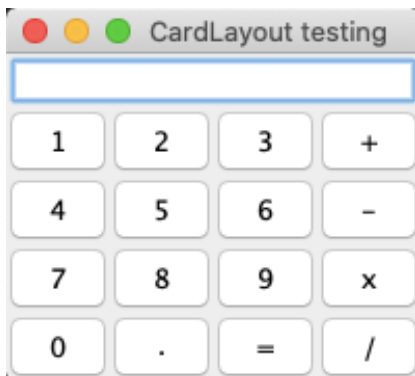
```

if (e.getActionCommand() == "1" || e.getActionCommand() == "2"
    || e.getActionCommand() == "3" || e.getActionCommand() == "4"
    || e.getActionCommand() == "5" || e.getActionCommand() == "6"
    || e.getActionCommand() == "7" || e.getActionCommand() == "8"
    || e.getActionCommand() == "9" || e.getActionCommand() == "0"
    || e.getActionCommand() == ".")
{
    if (("+").equals((String) v2.lastElement())
        || ("-").equals((String) v2.lastElement())
        || ("x").equals((String) v2.lastElement())
        || ("/").equals((String) v2.lastElement())
        || ("init").equals((String) v2.lastElement())) {
        tf.setText("");
        tf.setText(e.getActionCommand());
    } else if ("=".equals((String) v2.lastElement())) {
        sum = 0;
        tf.setText("");
        tf.setText(e.getActionCommand());
    } else {
        tf.setText(tf.getText() + e.getActionCommand());
    }
    v2.addElement(new String(e.getActionCommand()));
}
}

public static void main(String[] args) {
    new Calculator().init();
}

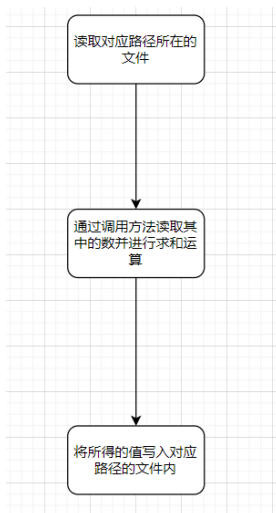
```

[程序运行结果]



习题四：

[程序流程图]



[数据结构设计]

```
/*  
    首先读取对应路径所在的文件，  
    然后调用类的方法读取其中的数并进行求和运算，得出总分  
    最后再将该数据写入需要写入的文件即可  
*/
```

[程序源代码]

```
public class Credit {  
    //设置类用其中的方法读取并计算总分  
    public static class Sum {  
        public int Score(String str){  
            Scanner scanner = new Scanner(str);  
            scanner.useDelimiter("[^0123456789.]+");  
            int sum = 0;  
            while (scanner.hasNext()){  
                try {  
                    int i = scanner.nextInt();  
                    System.out.println(i);  
                    sum = sum + i;  
                }catch(Exception e){  
                    scanner.next();  
                }  
            }  
            return sum;  
        }  
    }  
}
```

```

public static void main(String[] args) {
    //初始化
    FileReader fileReader = null;
    FileWriter fileWriter = null;
    BufferedReader bufferedReader = null;
    BufferedWriter bufferedWriter = null;

    try {
        fileReader = new FileReader( fileName: "/Users/sairen/javalearn/exp_6/cre.txt");
        fileWriter = new FileWriter( fileName: "/Users/sairen/javalearn/exp_6/newCre.txt");
        bufferedReader = new BufferedReader(fileReader);
        bufferedWriter = new BufferedWriter(fileWriter);
        String b;
        while ((b = bufferedReader.readLine()) != null) {
            System.out.println(b);

            Sum s = new Sum();
            int i = s.Score(b);

            bufferedWriter.write( str: b + "总分: " + i);
            bufferedWriter.newLine();
        }
    } catch (IOException e){
        e.printStackTrace();
    } finally{

```

```

    } finally{
        try {
            assert fileReader != null;
            fileReader.close();
            assert bufferedWriter != null;
            bufferedWriter.flush();
            fileWriter.close();
            bufferedReader.close();
            bufferedWriter.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

```

[程序运行结果]



七、选做题

1、(统计英文单词)编写程序使用Scanner 类和正则表达式统计一篇英文文本中的单词，要求如下：

- 一共出现了多少个单词。
- 有多少个互不相同的单词。
- 按单词出现频率大小输出单词。

运行效果参考如下：

```
共有12个英文单词  
有5个互不相同英文单词  
按出现频率排列：  
are:0.333  students:0.333  We:0.167  you:0.083  goods:0.083
```

八、实验总结与体会

在本次实验中，巩固了对异常处理与文本IO的相关知识，能够更加快速的处理相对应的异常处理问题，能够使用语句声明、抛出和捕获异常。

在实验过程中，对于实验的问题三，要求对用户输入的数据进行检测以免出现输入非数据的情况。既然如此，利用按钮规定他们所能够输入的内容就可以了，可以避免一次检测处理异常，因此我采用了另外一种模式，让用户通过按动按钮来进行计算的使用。