1 ）加法功能模块 ：两个数据可进行加操作，可以为小数。

2 ）减法功能模块 ：两个数据可进行减操作，可以为负数相减。

3 ）乘法功能模块 ：两个数相乘操作。

4 ）除法功能模块 ：两个数相除操作。

5 ）开平方功能模块 ：对任意一个数进行开平方运算，可以是小数。

6 ）百分号功能模块 ：对任意一个数进行百分号运算。

/\*\*

\* 构造函数

\*/

public Calculator() {

super();

// 初始化计算器

init();

// 设置计算器的背景颜色

this.setBackground(Color.LIGHT\_GRAY);

this.setTitle("计算器");

// 在屏幕(500, 300)坐标处显示计算器

this.setLocation(500, 300);

// 不许修改计算器的大小

this.setResizable(false);

// 使计算器中各组件大小合适

this.pack();

}

// 文本框中的内容采用右对齐方式

resultText.setHorizontalAlignment(JTextField.RIGHT);

// 不允许修改结果文本框

resultText.setEditable(false);

// 设置文本框背景颜色为白色

resultText.setBackground(Color.WHITE);

// 初始化计算器上键的按钮，将键放在一个画板内

JPanel calckeysPanel = new JPanel();

// 用网格布局器，4行，5列的网格，网格之间的水平方向间隔为3个象素，垂直方向间隔为3个象素

calckeysPanel.setLayout(new GridLayout(4, 5, 3, 3));

for (int i = 0; i < KEYS.length; i++) {

keys[i] = new JButton(KEYS[i]);

calckeysPanel.add(keys[i]);

keys[i].setForeground(Color.blue);

}

// 运算符键用红色标示，其他键用蓝色表示

keys[3].setForeground(Color.red);

keys[8].setForeground(Color.red);

keys[13].setForeground(Color.red);

keys[18].setForeground(Color.red);

keys[19].setForeground(Color.red);

// 初始化功能键，都用红色标示。将功能键放在一个画板内

JPanel commandsPanel = new JPanel();

// 用网格布局器，1行，3列的网格，网格之间的水平方向间隔为3个象素，垂直方向间隔为3个象素

commandsPanel.setLayout(new GridLayout(1, 3, 3, 3));

for (int i = 0; i < COMMAND.length; i++) {

commands[i] = new JButton(COMMAND[i]);

commandsPanel.add(commands[i]);

commands[i].setForeground(Color.red);

}

// 初始化M键，用红色标示，将M键放在一个画板内

JPanel calmsPanel = new JPanel();

// 用网格布局管理器，5行，1列的网格，网格之间的水平方向间隔为3个象素，垂直方向间隔为3个象素

calmsPanel.setLayout(new GridLayout(5, 1, 3, 3));

for (int i = 0; i < M.length; i++) {

m[i] = new JButton(M[i]);

calmsPanel.add(m[i]);

m[i].setForeground(Color.red);

}

// 下面进行计算器的整体布局，将calckeys和command画板放在计算器的中部，

// 将文本框放在北部，将calms画板放在计算器的西部。

// 新建一个大的画板，将上面建立的command和calckeys画板放在该画板内

JPanel panel1 = new JPanel();

// 画板采用边界布局管理器，画板里组件之间的水平和垂直方向上间隔都为3象素

panel1.setLayout(new BorderLayout(3, 3));

panel1.add("North", commandsPanel);

panel1.add("Center", calckeysPanel);

// 建立一个画板放文本框

JPanel top = new JPanel();

top.setLayout(new BorderLayout());

top.add("Center", resultText);

// 整体布局

getContentPane().setLayout(new BorderLayout(3, 5));

getContentPane().add("North", top);

getContentPane().add("Center", panel1);

getContentPane().add("West", calmsPanel);

1 ．系统总体分析

对于此次运算器程序，首先设计其 GUI 界面，总体界面有一个文本框。将calckeys和command画板放在计算器的中部，将文本框放在北部，将calms画板放在计算器的西部，界面设计完成。布局设计好后再添加按钮事件。对于此程序考虑到点击加减乘除按钮时是否点击数字按钮，如果是第一次点击数字按钮或者是点击数字按钮前点击了加减乘除按钮则直接把数字按钮的数值设置到文本框中，否则应把文本框中的内容加上用户所点击按钮的数值设置到文本框中。在进行加、减、乘、除计算时点击加、减、乘、除按钮之前的数值保存下来，运算时是这个被保存下来的数值和文本框中的数值加、减、乘、除。

2. 系统界面部分

主窗体包含的控件中要有：标签，用于显示提示语句 ; 文本文档，用于用户数值的；按钮键，用于处理操作。界面初始化时，将一些按键，例如完成加、减、乘、除运算按键设置成不可用，这样可以防止误操作，每次按钮按下时，都有相应的按钮属性被更改。

3 ．系统后台部分

后台部分，核心就是产生随即算术式，用一个字符串数组存放加和减的操作符，产生一个 0-9 的随机数，通过此随机数对应操作符数组中的字符，决定产生的算术式为加法或减法，然后进行两个数值的四则运算，通过以上运算求出结果，保存在私有变量中，同时对应的产生一个通过连接完成的字符串，作为界面题目输出的内容，以上便完成随机算术式的生成。