

哈尔滨工业大学

计算机科学与技术学院/国家示范性软件学院

2021 年秋季学期

《软件过程与工具》课程

实验报告

Lab 1 : 增量模型应用+结对编程实践

姓名	学号	联系方式
惠丹妮	1191000713	18329806688
袁文宇	1180100406	18845771094

目 录

1 项目概述.....	1
1.1 总体需求分析.....	1
1.2 软件系统总体设计.....	1
1.3 开发任务计划.....	2
1.4 开发环境与工具.....	3
2 第 1 轮迭代：基本系统.....	3
2.1 迭代目标.....	3
2.2 需求分析.....	3
2.3 设计方案.....	3
2.4 结对编程过程记录.....	8
2.5 单元测试.....	9
2.6 集成测试.....	10
3 第 2 轮迭代：增量 1.....	10
3.1 迭代目标.....	11
3.2 需求分析.....	11
3.3 设计方案.....	11
3.4 结对编程过程记录.....	17
3.5 单元测试.....	18
3.6 集成测试.....	19
4 第 3 轮迭代：增量 2.....	20
4.1 迭代目标.....	20
4.2 需求分析.....	20
4.3 设计方案.....	20
1.1 结对编程过程记录.....	27
1.2 单元测试.....	28
1.3 集成测试.....	29
5 项目总结.....	29
5.1 增量模型应用总结.....	29
5.2 结对编程总结.....	29

1 项目概述

(1) 项目名称: 综合计算器

(2) 项目简要介绍: 综合计算机共有三个主要功能, 普通计算器, 科学计算器和历史记录。其中普通计算器可以实现基本的加减乘除等功能, 更适合日常简单计算使用; 而科学计算器可以实现指数、对数、括号、阶乘, 二进制和十进制转换等复杂计算, 很适合学生、老师等人群进行计算; 最后还有一个历史功能, 可以用来记录历史计算信息, 使用时可以减少重复运算的麻烦。

1.1 总体需求分析

(1) 项目要开发的软件的目的和意义 (回答为什么要开发该项目)。

开发目的: 日常生活中有许多的计算需求需要用到计算器, 计算器非常便捷, 能帮助人们解决手算慢、心算易出错的问题。

开发意义: 不同的人群有不同的需求, 因此我们帮助不同的人群解决计算的问题, 并且

(2) 软件系统的涉众分析 (列出该系统跟哪些相关人员有关, 其关心的核心内容)。

相关人员: 该软件与学生, 老师, 会计等有计算需求且使用电脑的人有关;
关心的核心内容: 计算的正确性、计算器的易操作性、计算器功能的丰富性。

(3) 软件系统的功能需求: 可从用户视角来描述功能需求 (即用户故事)。

可以计算加减乘除, 对数, 指数, 根号, 平方, e 的幂次, 10 的幂次等等功能, 常用的功能集成于普通计算器, 相对常用的功能集成于科学计算器, 能查看计算的历史记录, 以查询之前的计算记录, 但历史记录在不需要的时候也可以删除, 要能动态管理。

同时, 界面要整洁易懂, 风格良好, 计算器的功能要便于使用, 便于用户理解, 不能计算出错, 要有稳定的表现。

(4) 软件系统的非功能需求: 包括性能要求、界面要求、技术指标等。

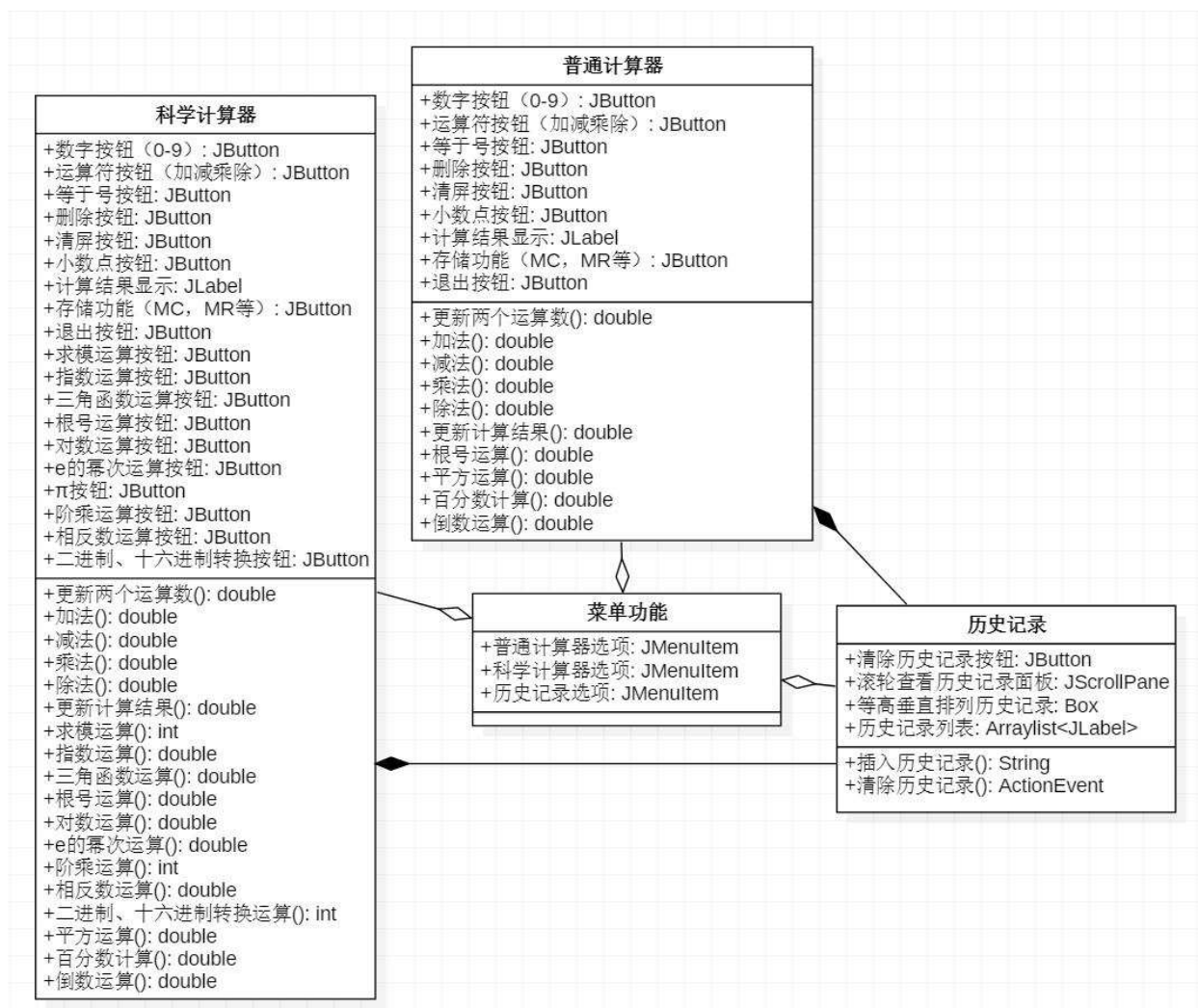
性能要求: 防止溢出, 鲁棒性, 计算快速;

界面要求: 简洁美观易上手;

技术指标: 计算准确。

1.2 软件系统总体设计

(1) 软件系统功能结构设计方案: 给出软件功能模块及模块间的关联关系。



(2) 软件系统相关的数据结构设计：给出必要的数据结构设计，包括配置参数、缓冲数据结构等。

①在 History 历史记录类中，使用链表维护当前存储的历史记录，随时准备动态增删。

```
private List<JLabel> history = new ArrayList<>(); //动态管理历史记录
```

②在 Calculator 计算器类中，使用 double 维护当前的两个运算数，用 operator 维护当前运算类型，用 firstnum 判断是否是第一个数字，用 M 实现 MR, MC, M+, M- 功能，用 strresult 字符串存储计算器计算结果的变化字符串。

```

/*****运算使用变量*****/
private boolean firstnum = true; //判断是否是第一个数字
private char operator = '='; //用于等号判断进行的运算类型
private String strresult1 = "0"; //保存按数字键后文本框的数值
private String strresult2 = "0"; //保存清空前文本框的数值
private double result1 = 0.0; //上行计算结果保存
private double result2 = 0.0; //输入栏数字保存
private double result3 = 0.0; //保存result2
private double M = 0.0; //存储功能

```

1.3 开发任务计划

迭代过程的划分, 各迭代计划完成的增量任务, 进度安排。

(1) 第 1 轮迭代的任务: 实现简单计算功能(加、减、乘、除、等于号、清零、删除键、数字键、小数点、退出键等基本功能), 2021/10/14-2021/10/17。

(2) 第 2 轮迭代的任务: 实现科学计算功能(三角函数、指数、对数、求模运算、根号运算、e 的幂次运算、阶乘运算、相反数运算、二进制和十六进制转换运算等), 2021/10/17-2021/10/18。

(3) 第 3 轮迭代的任务: 添加、查看历史记录与删除历史记录, 2021/10/18-2021/10/19

1.4 开发环境与工具

开发语言: Java 1.8,

开发环境: windows 10,

开发工具: eclipse, StarUML

2 第 1 轮迭代: 基本系统

2.1 迭代目标

简要介绍本轮迭代的增量任务目标, 所需完成的任务。

(1) 任务 1: 了解学习 java 的 GUI 开发,

(2) 任务 2: 先实现 java 程序中窗口的基本设计(一些按钮和界面), 搭出基本的框架。

(3) 任务 3: 实现按钮的监听, 完成底层功能。

2.2 需求分析

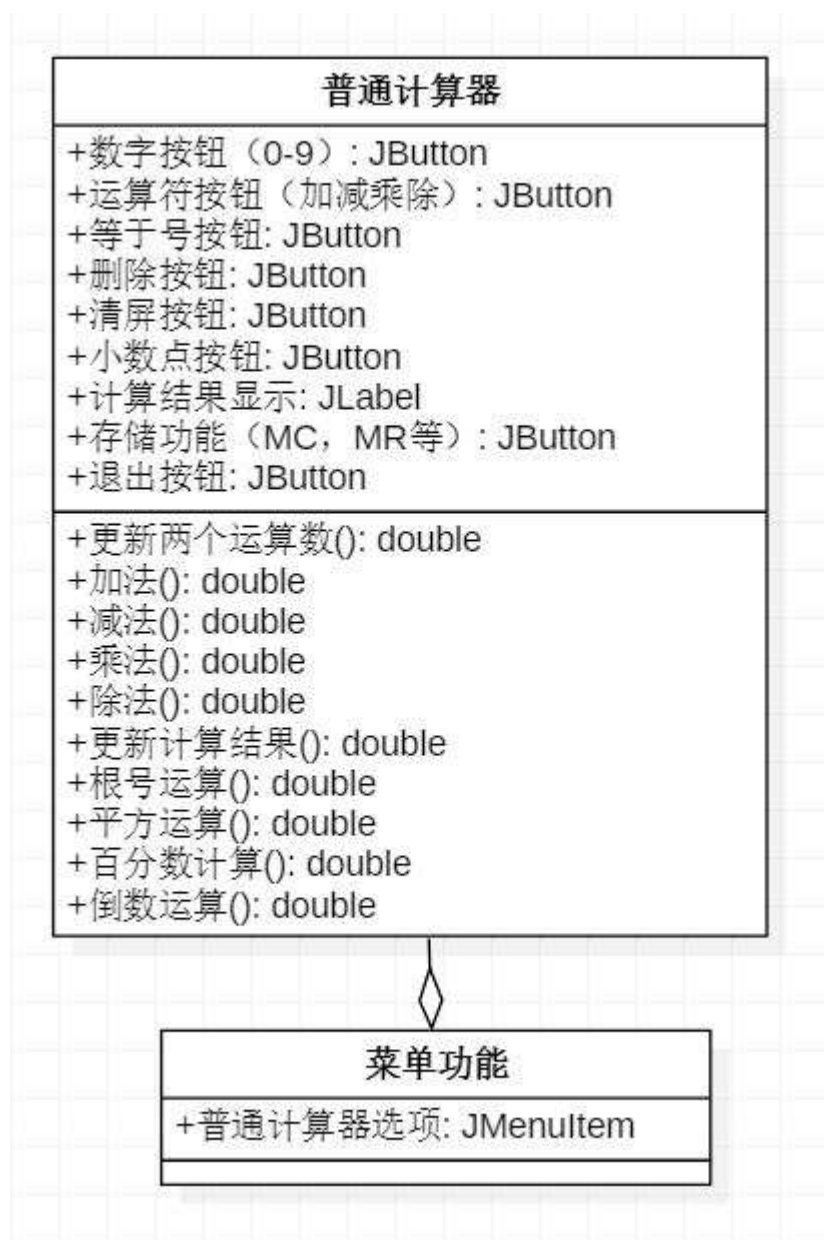
设计一个计算准确、快速, 界面清晰整洁, 用户友好, 便于使用, 便于用户理解, 风格良好, 满足基本计算需求的标准计算器。满足加减乘除、平方、根号、倒数等基本运算功能需求。

2.3 设计方案

针对本轮迭代的开发内容给出设计方案:

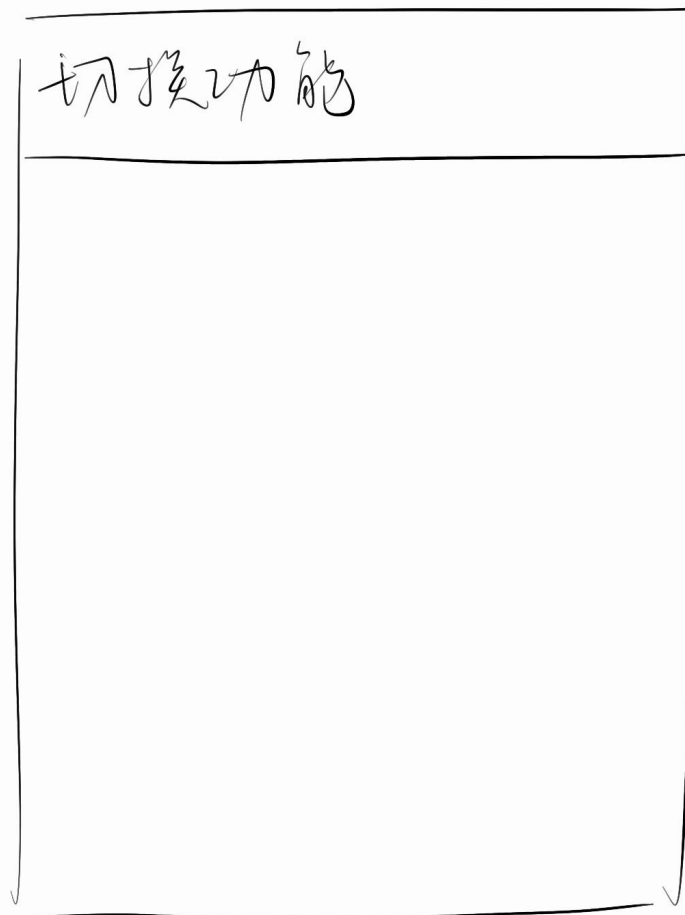
(1) 详细设计方案: 模块功能结构设计, 界面原型设计等。

模块功能设计:



界面原型设计:

初始界面：



切换功能			
0			
MC	MR	M+	M-
mod	$\sqrt{\quad}$	x^2	$\frac{1}{x}$
CE	C	$\boxed{\times}$	\div
7	8	9	\times
4	5	6	-
1	2	3	+
ESC	0	-	=

(2) 模块相关的设计类 (Design Class): 给出每个类的名字、属性、方法。



(3) 所需的数据结构。

在 Calculator 计算器类中，使用 double 维护当前的两个运算数，用 operator 维护当前运算类型，用 firstnum 判断是否是第一个数字，用 M 实现

MR,MC,M+,M- 功能，用 strresult 字符串存储计算器计算结果的变化字符串。

```

/*****运算使用变量*****/
private boolean firstnum = true;           //判断是否是第一个数字
private char operator = '=';               //用于等号判断进行的运算类型
private String strresult1 = "0";           //保存按数字键后文本框的数值
private String strresult2 = "0";           //保存清空前文本框的数值
private double result1 = 0.0;              //上行计算结果保存
private double result2 = 0.0;              //输入栏数字保存
private double result3 = 0.0;              //保存result2
private double M = 0.0;                    //存储功能

```

2.4 结对编程过程记录

(1) 角色切换与任务分工

表 1-1 第 1 轮迭代过程结对编程角色与任务分工

日期	时间(HH:MM - HH:MM)	驾驶员角色	领航员角色	本段时间的任务
2021/10/14	18:30-21:00	袁文字	惠丹妮	完成基本的窗口类的设计
2021/10/14	21:30-23:59	袁文字	惠丹妮	完成整个面板的外观搭建
2021/10/15	15:30-18:00	惠丹妮	袁文字	完成按钮的监听
2021/10/16	19:40-23:00	惠丹妮	袁文字	完成底层功能的实现
2021/10/17	20:00-23:30	袁文字	惠丹妮	完成整体 debug，实现能运行的软件

(2) 工作日志

由领航员负责记录，记录结对编程期间的遇到的问题、两人如何通过交流合作解决每个问题的。

表 1-2 第 1 轮迭代过程结对编程工作日志

日期/时间	问题描述	最终解决方法	交流过程
2021/10/14 20:36	按钮不显示	如果将按钮 add 到 JPanel 上，最后必须要 add(JPanel)，否则直接 add 按钮。	肯定是 add 的位置不对；自定义的 JPanel 应该不会自动被设置成默认 panel，尝试 add 它到 JFrame 也许可以解决问题。
2021/10/14 23:30	窗口中的按钮位置不好看，难以设置	将 layout 设置为 null 才可以自由设置每个按钮的具体位置；计算器排版使用 BorderLayout 很适合。	可能不同 layout 对组件管理方式不一样，查找一下各个 layout 的样例程序。
2021/10/15	无法实例化监听类	监听函数不能在 main	改变一下监听函数的位置，

日期/时间	问题描述	最终解决方法	交流过程
17:20		函数里面设置,他只能用于实例化的实例中。	将它放到类的函数里而不是在 main 里使用。
2021/10/17 20:10	对于连加连减之类的不按等于号的计算有 bug	用 operator 变量存储上一步计算的运算符,分类讨论各种情况	代码考虑到的情况少了,应该还要考虑上一步的计算是什么;我们应该加上分类讨论的情况。

(3) 结对编程工作现场照片

请其他同学帮助拍摄结对编程现场照片至少 2 张。



第 1 次迭代结对编程现场照片 1



第 1 次迭代结对编程现场照片 2

2.5 单元测试

对本轮迭代所开发的每个类进行单元测试。

表 1-3 第 1 轮迭代过程的单元测试记录

日期/时间	类名: 方法名	输入数据	输出数据	结论
2021/10/17 22:18	Calculator: UI()	点击初始页面左上角的“切换类型(S)”, 点击弹出的“标准(Standard)”	弹出选项“标准(Standard)”, 点击进入了标准计算器, 页面完整符合预期	功能完善
2021/10/17 22:20	Calculator: backspace()	在 \geq 两位数时(10)点击删除键; 在只有一位数时(1)点击删除键	分别为 1 和 0	功能完善
2021/10/17 22:25	Calculator: add() Calculator: result()	连续点击 $1+1+1=$	得到 3	加法和等于按钮功能完善
2021/10/17	Calculator: sub()	连续点击 $1-1-1=$	得到-1	功能

日期/时间	类名: 方法名	输入数据	输出数据	结论
22:26	Calculator: result()			完善
2021/10/17 22:27	Calculator: multi() Calculator: result()	连续点击 2*2*2=	得到 8	功 能 完善
2021/10/17 22:28	Calculator: division() Calculator: result()	连续点击 8/2/3=	得到 1.33333	功 能 完善
2021/10/17 22:30	Calculator: divi() Calculator: result()	连续点击 8/0=	显示“非法被除数”	功 能 完善
2021/10/17 22:32	Calculator: btnum() Calculator: c()	连续点击 0 和 1/x 按钮 然后清屏,连续点击 2 和 1/x 按钮	分别显示无法输入” 和 0.5	倒数、 清 屏 功 能 完善
2021/10/17 22:35	Calculator: square()	连续点击 3 和平方按钮	显示 9	功 能 完善
2021/10/17 22:37	Calculator: sroot()	连续点击 3 和根号按钮; 连续点击-2 和根号按钮	分别显示 1.73205... 和 NaN	功 能 完善, 不 会 因 为 违 规 输 入 而 报 错
2021/10/17 22:39	Calculator: ce()	连续点击 3*2, 然后点击 CE 按钮, 然后点击 3	显示 9	功 能 完善
2021/10/17 22:40	Calculator: percent()	连续点击 3 和百分号 按钮	显示 0.03	功 能 完善

2.6 集成测试

对本轮迭代所开发功能模块进行测试。

表 1-4 第 1 轮迭代过程的集成测试记录

日期/时间	功能名称	测试用例	测试结果描述	结论
2021/10/17 22:45	加减乘除计算功能	连续点击 1+1*2/3-6	显 示 - 4.666666666666667	基 本 计 算 功 能 正 确
2021/10/17 23:00	存储功能	连 续 点 击 1,MR,C,M+,M-,MC,M +	分别显示 1, 1, 0, 1,0,0	基 本 计 算 功 能

日期/时间	功能名称	测试用例	测试结果描述	结论
				正确

3 第 2 轮迭代：增量 1

3.1 迭代目标

简要介绍本轮迭代的增量任务目标，所需完成的任务。

(1) 任务 1: 完成科学计算器的菜单选项卡添加，能随意切换到科学计算器和普通计算器。

(2) 任务 2: 完成科学计算器的按钮排版。

(3) 任务 3: 完成底层计算的功能监听、整合。

3.2 需求分析

普通计算器只能满足较少的常用计算器功能使用，而一些较为常用的功能并没有整合进去，比如高中、大学学生和老师对计算的需求不仅限于加减乘除、根号平方等计算功能，还有 e 的指数次幂、三角函数、指数、对数、求模运算、根号运算、阶乘运算、相反数运算、二进制和十六进制转换运算等的计算需求，因此针对该部分人群我们需要设计功能更为丰富的科学计算器。

3.3 设计方案

针对本轮迭代的开发内容给出设计方案：

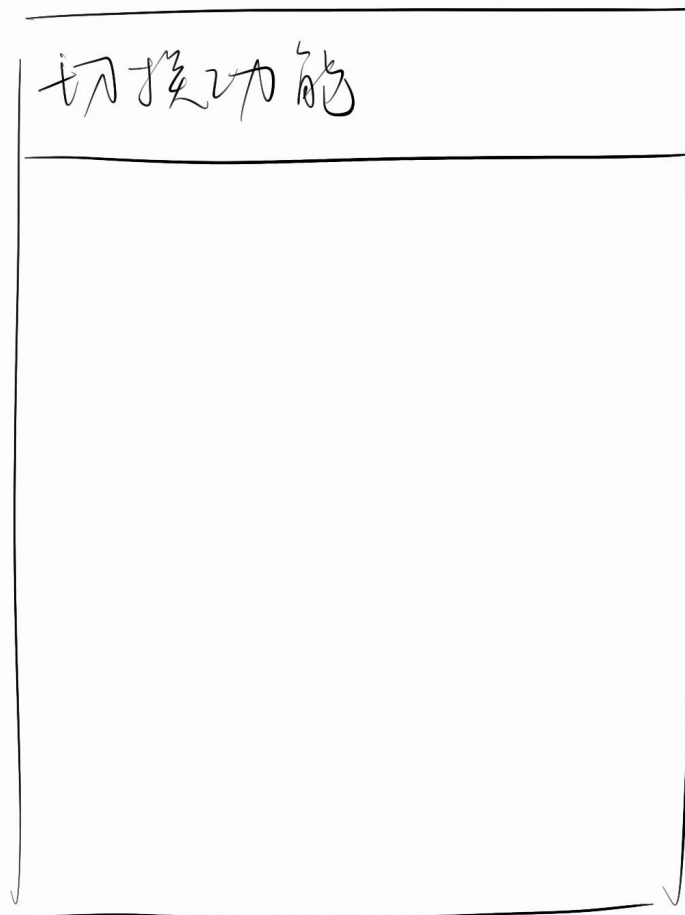
(1) 详细设计方案：模块功能结构设计，界面原型设计等。

模块功能结构设计：



界面原型设计:

初始界面：



切换功能			
0			
MC	MR	M+	M-
mod	$\sqrt{\quad}$	x^2	$\frac{1}{x}$
CE	C	$\boxed{\times}$	\div
7	8	9	\times
4	5	6	-
1	2	3	+
ESC	0	-	=

切换功能				
0				
MC	MR	M+	M-	mod
x^2	x^y	sin	cos	tan
$\sqrt{\quad}$	10^x	二进制	log	e^x
ln	CE	C	$\boxed{\times}$	\div
π	7	8	9	\times
n!	4	5	6	-
相反数	1	2	3	+
ESC	十六进制	0	.	=

(2) 模块相关的设计类 (Design Class): 给出每个类的名字、属性、方法。

Calculator
+frame: JFrame +xx: int +yy: int +isDragging: boolean +fcolor: Color +bcolor1: Color +bcolor2: Color +bcolor3: Color +tclock: Color +ocolor1: Color +ocolor2: Color +fmnu: Font +fsubmenu: Font +panel0: JPanel +panel: JPanel +panel1: JPanel +sfpanel1: JPanel +jtxt1: JTextField +jlib2: JLabel +firstnum: boolean +operator: char +strresult1: String +strresult2: String +result1: double +result2: double +result3: double +M: double +sdsbtn: JButton +sfsbtn: JButton +jmb: JMenuBar +menu1: JMenu +item1: JMenuItem +item2: JMenuItem +item3: JMenuItem
+Calculator() +actionPerformed(ActionEvent): void +mr(): void +mc(): void +mplus(): void +msub(): void +backspace(): void +ce(): void +c(): void +result(): void +add(): void +sub(): void +mult(): void +division(): void +divi(): void +percent(): void +sroot(): void +square(): void +btrnum(): void +main(String[]): void +xy(): void +y(): void +tan(): void +cos(): void +sin(): void +get10x(): void +bin(): void +hex(): void +log(): void +exp(): void +mod(): void +ln(): void +fact(): void +getPi(): void +reverst(): void

(3) 所需的数据结构。

在 Calculator 计算器类中，使用 double 维护当前的两个运算数，用 operator

维护当前运算类型，用 `firstnum` 判断是否是第一个数字，用 `M` 实现 `MR,MC,M+,M-` 功能，用 `strresult` 字符串存储计算器计算结果的变化字符串。

```

/*****运算使用变量*****/
private boolean firstnum = true;           //判断是否是第一个数字
private char operator = '=';               //用于等号判断进行的运算类型
private String strresult1 = "0";          //保存按数字键后文本框的数值
private String strresult2 = "0";          //保存清空前文本框的数值
private double result1 = 0.0;              //上行计算结果保存
private double result2 = 0.0;              //输入栏数字保存
private double result3 = 0.0;              //保存result2
private double M = 0.0;                    //存储功能

```

3.4 结对编程过程记录

(1) 角色切换与任务分工

表 2-1 第 2 轮迭代过程结对编程角色与任务分工

日期	时间(HH:MM - HH:MM)	驾驶员角色	领航员角色	本段时间的任务
2021/10/18	9:00-10:30	袁文字	惠丹妮	完成科学计算器的排版
2021/10/18	10:30-11:30	惠丹妮	袁文字	初步为新的按钮添加监听函数实现功能
2021/10/18	14:00-17:30	惠丹妮	袁文字	进一步为新的按钮添加监听函数实现功能
2021/10/18	19:00-21:30	袁文字	惠丹妮	进行各个功能的 debug 和改进完善

(2) 工作日志

由领航员负责记录，记录结对编程期间的遇到的问题、两人如何通过交流合作解决每个问题的。

表 2-2 第 2 轮迭代过程结对编程工作日志

日期/时间	问题描述	最终解决方法	交流过程
2021/10/18 15:45	求十六进制的函数抛出了 <code>exception</code>	发现一开始将阶乘的字符串获取函数设置成了 <code>Integer.parseInt()</code> ，但输入很可能是一个浮点数，所以应该用 <code>Double.parseDouble</code> ，然后强制转换成 <code>int</code> 来求解	我们输入的数字全程设置的是小数，但阶乘计算应该是整数才对；所以报错了，应该考虑到一切不规范输入的情况，要保持代码健壮性，不论什么输入都能不崩溃、不出错
2021/10/18 17:00	连续 <code>mod</code> 出错，余数全变 0 了	每次判断是否是第一次输入数，更新被除数	我们有没有更新被除数？设置了但报错了。把这几

日期/时间	问题描述	最终解决方法	交流过程
			个变量都打印出来看看。 我们设置错了。
2021/10/18 20:40	进入科学计算器界面之后,再点回普通计算器没反应	不是没反应,而是多添加了一个普通计算器。删除多添加的 panel 就好了。	这个应该不是没反应,而是多了一个。我们想让初始界面不是空白的,把普通计算器添加到了初始界面,但这样就多添加了这个界面,后续不好删除。还是使用一开始设置好的初始界面比较好。

(3) 结对编程工作现场照片

请其他同学帮助拍摄结对编程现场照片至少 2 张。



第 2 次迭代结对编程现场照片 1



第 2 次迭代结对编程现场照片 2

3.5 单元测试

对本轮迭代所开发的每个类进行单元测试。

表 2-3 第 2 轮迭代过程的单元测试记录

日期/时间	类名: 方法名	输入数据	输出数据	结论
2021/10/18 21:40	Calculator: fact()	先输入 2.3, 再点击阶乘按钮	2.0	功能正常
2021/10/18 21:40	Calculator: UI()	点击初始页面左上角的“切换类型(S)”, 点击弹出的“科学(Scientific)”	弹出选项“科学(Scientific)”, 点击进入了科学计算器, 页面完整符合预期	功能完善
2021/10/18 21:42	Calculator: xy() Calculator: y()	先输入 2, 再点击指数按钮, 再输入 3, 点击等于号	8.0	功能正常
2021/10/18 21:43	Calculator: mod()	先输入 19, 再点击 mod 按钮, 再输入 7,	5.0	功能正常

日期/时间	类名: 方法名	输入数据	输出数据	结论
2021/10/18 21:44	Calculator: tan() Calculator: getPi()	点击等于号 先点击 π 按钮, 再 点击 tan 按钮	先输出 3.141592653..., 然 后 输 出 了 1.2246467991473532E- 16	$\tan\pi$ 趋 近 于 0, 功 能 正常
2021/10/18 21:45	Calculator: cos() Calculator: getPi()	先点击 π 按钮, 再 点击 cos 按钮	-1.0	功 能 正常
2021/10/18 21:47	Calculator: sin() Calculator: getPi()	先点击 π 按钮, 再 点击 sin 按钮	1.2246467991473532E- 16	$\sin\pi$ 趋 近 于 0, 功 能 正常
2021/10/18 21:48	Calculator: get10x()	先输入 2, 再点击 10 的指数按钮; 输 入-3, 再点击 10 的 指数按钮	100.0 0.001	功 能 正常
2021/10/18 21:50	Calculator: bin()	先输入 6, 再点击 转换二进制按钮; 再输入-2, 点击转 换二进制按钮	110 111111111111...1111111 0	功 能 正常
2021/10/18 21:53	Calculator: hex()	先输入 12, 再点击 转换十六进制按钮; 再输入-2, 点击转 换十六进制按钮	c fffffffe	功 能 正常
2021/10/18 21:55	Calculator: ln()	先输入 3, 再点击 ln 按钮; 再输入-2, 点击 ln 按钮	1.0986122886681098 应为正数	功 能 正常
2021/10/18 22:01	Calculator: log()	先输入 3, 再点击 log 按钮; 再输入-2, 点击 log 按钮	0.47712125471966244 NaN	功 能 正常
2021/10/18 22:05	Calculator: exp()	先输入 3, 再点击 exp 按钮; 再输入-2, 点击 exp 按钮	20.085536923187664 0.1353352832366127	功 能 正常

3.6 集成测试

对本轮迭代所开发功能模块进行测试。

表 2-4 第 2 轮迭代过程的集成测试记录

日期/时间	功能名称	测试用例	测试结果描述	结论
2021/10/18 22:10	整体计算功能+ 显示计算结果功能	输入 3, 点击 10 的幂次方按钮, 再点击 log 按钮, 再连续点击*2=, 再点击 bin, 再点击 hex, 再点击指数按钮, 输入 2, 再连续点击-9=	1000.0 2.9999999999999999 6 5.9999999999999999 101 65 4216.0	功 能 连 贯 性 良 好, 整 体 计 算 能 力 良 好
2021/10/18 22:20	窗口切换功能	启动程序, 点击切换类型, 点击科学, 再点击标准, 再点击科学; 启动程序, 点击切换类型, 点击标准, 再点击标准, 再点击科学, 再点击科学;	切换良好	窗 口 切 换 功 能 良好

4 第 3 轮迭代: 增量 2

4.1 迭代目标

简要介绍本轮迭代的增量任务目标, 所需完成的任务。

- (1) 任务 1: 在菜单中添加对历史记录选项。
- (2) 任务 2: 实现对 Calculator 按钮的监听, 每次出现计算结果时存储历史记录, 不仅限于按等于号的时候。
- (3) 任务 3: 实现历史记录的良好排版、良好查看风格样式, 保证数据清晰完整, 不会因为过多过少而出现差错或遗漏。
- (4) 任务 4: 实现历史记录的删除功能, 一键清空当前历史记录, 然后点击切换类型之后再选择历史记录选项就能看到历史记录被清空。

4.2 需求分析

客户有时需要完成大量的计算, 以至于忘记了之前的计算结果, 这种时候就需要历史记录进行查看, 否则重新算一遍费时费力, 用户体验不好。同时, 历史记录要能够一键清除, 防止用户不需要历史记录时它依旧占用用户的存储空间。

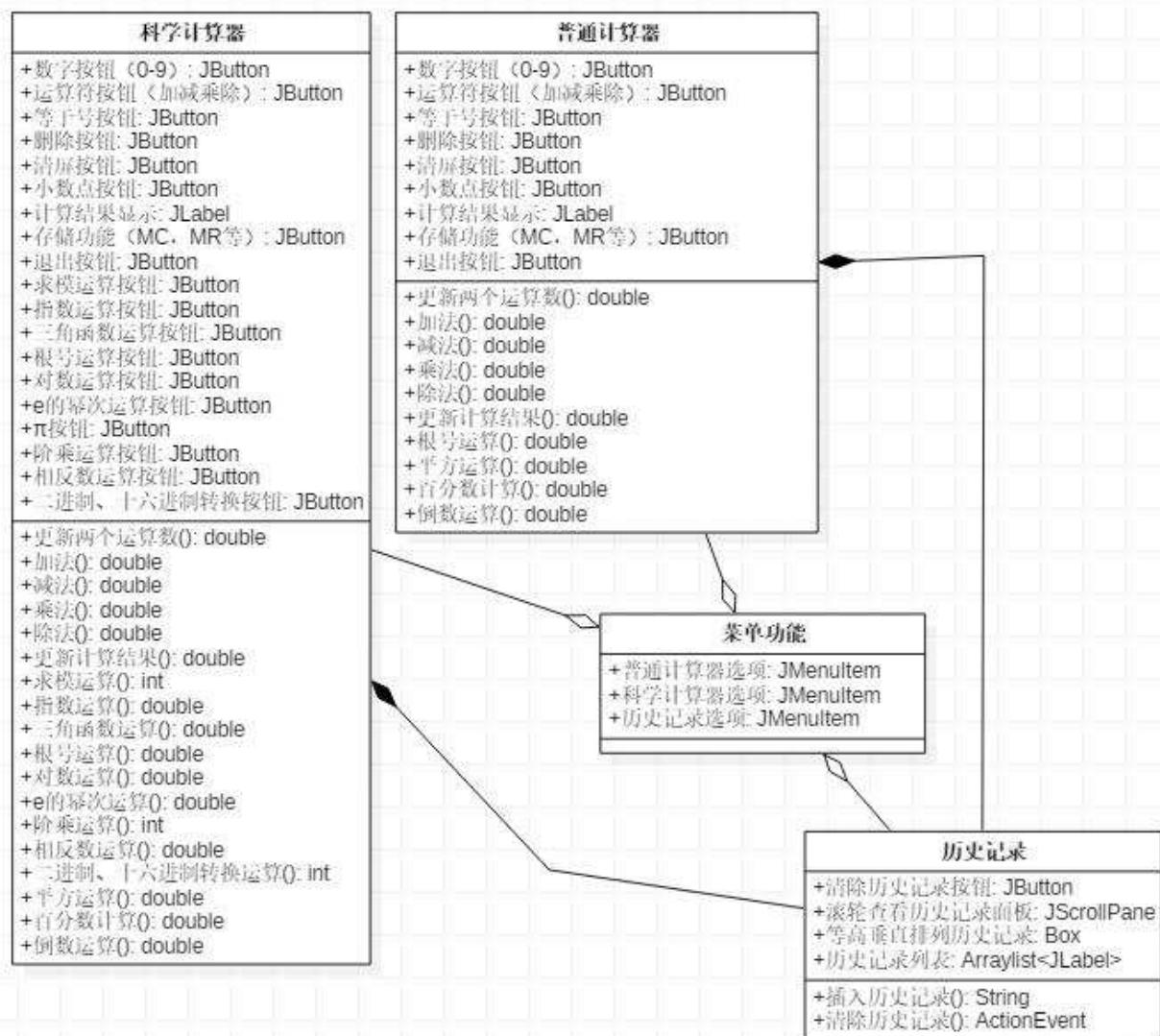
有的时候历史记录很长, 但字号变小会看不清, 字号大又显示不全, 所以要做一个横纵滚轮可以查看被挡住的内容。

4.3 设计方案

针对本轮迭代的开发内容给出设计方案：

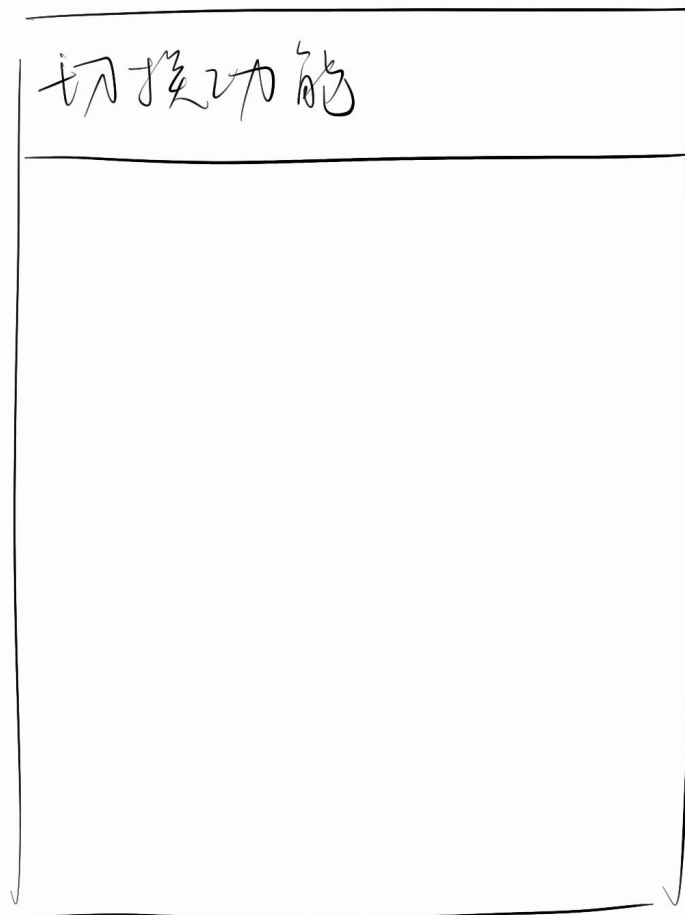
(1) 详细设计方案：模块功能结构设计，界面原型设计等。

模块功能结构设计：



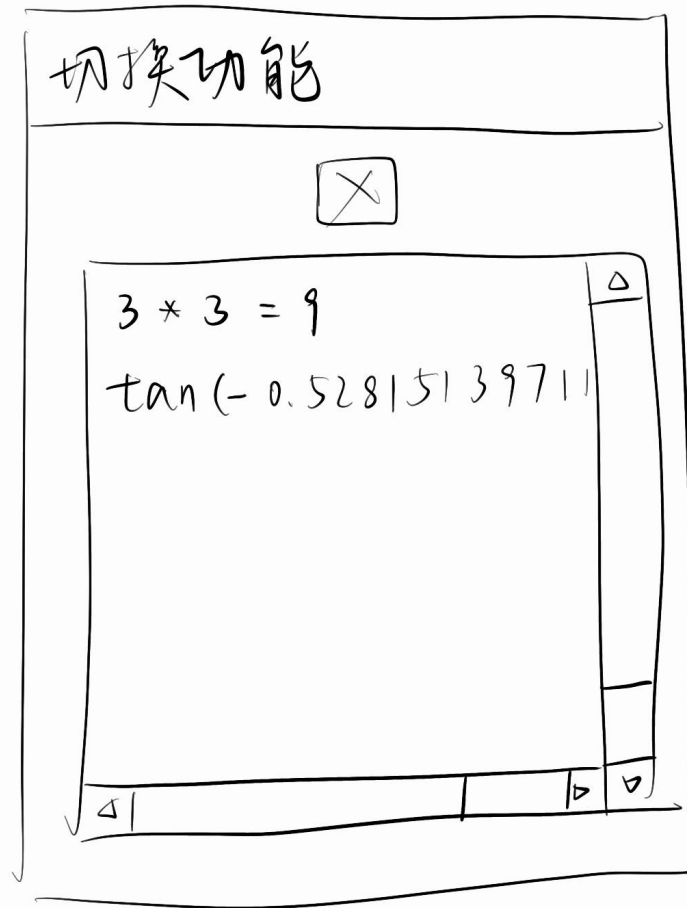
界面原型设计：

初始界面：

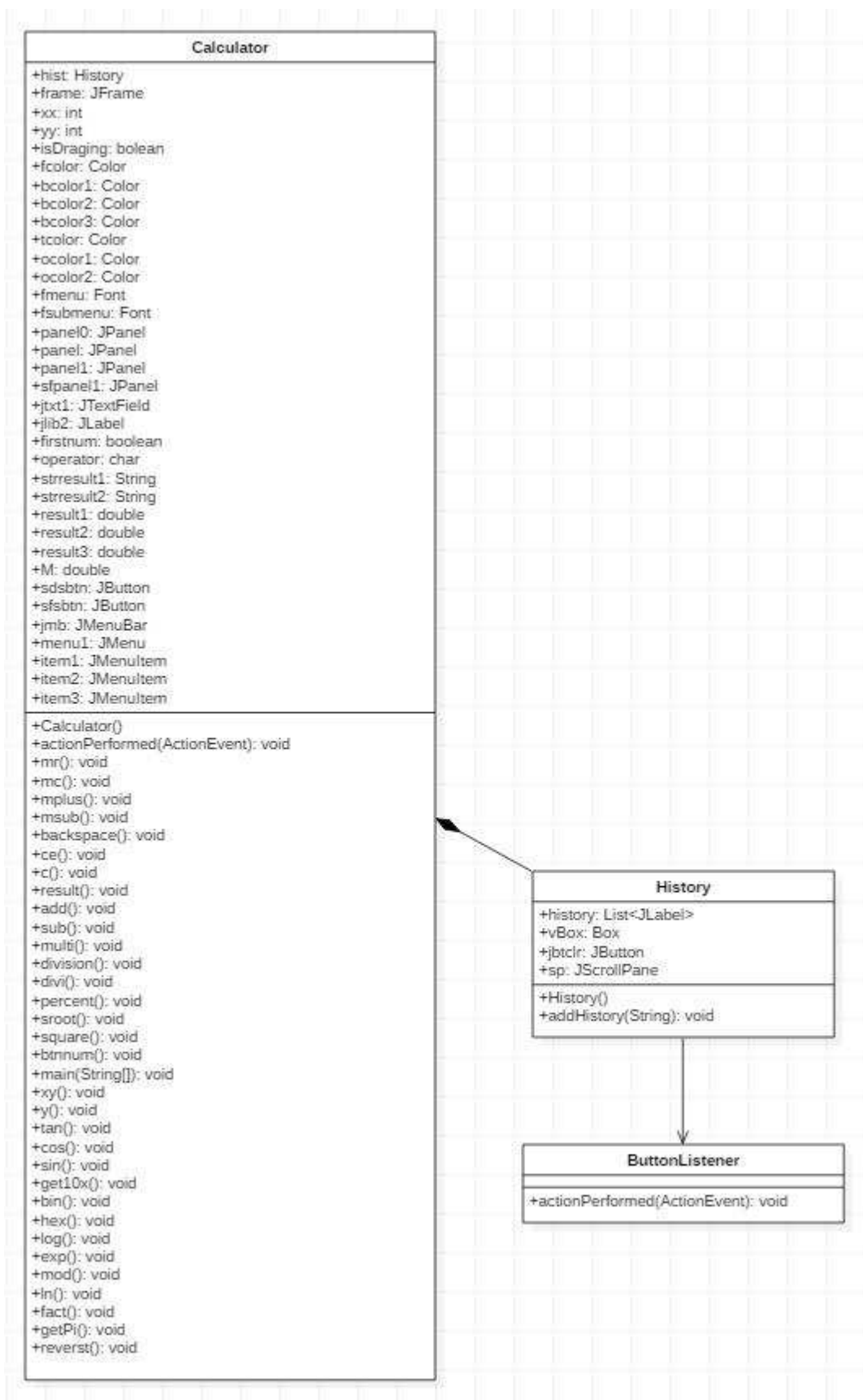


切换功能			
0			
MC	MR	M+	M-
mod	$\sqrt{\quad}$	x^2	$\frac{1}{x}$
CE	C	$\boxed{\times}$	\div
7	8	9	\times
4	5	6	-
1	2	3	+
ESC	0	-	=

切换功能				
0				
MC	MR	M+	M-	mod
x^2	x^y	sin	cos	tan
$\sqrt{\quad}$	10^x	二进制	log	e^x
ln	CE	C	$\boxed{\times}$	\div
π	7	8	9	\times
n!	4	5	6	-
相反数	1	2	3	+
ESC	+六进制	0	.	=



(2) 模块相关的设计类 (Design Class): 给出每个类的名字、属性、方法。



(3) 所需的数据结构。

①在 History 历史记录类中，使用链表维护当前存储的历史记录，随时准备

动态增删。

```
private List<JLabel> history = new ArrayList<>(); //动态管理历史记录
```

②在 Calculator 计算器类中, 使用 double 维护当前的两个运算数, 用 operator 维护当前运算类型, 用 firstnum 判断是否是第一个数字, 用 M 实现 MR,MC,M+,M-功能, 用 strresult 字符串存储计算器计算结果的变化字符串。

```

/*****运算使用变量*****/
private boolean firstnum = true;           //判断是否是第一个数字
private char operator = '=';               //用于等号判断进行的运算类型
private String strresult1 = "0";          //保存按数字键后文本框的数值
private String strresult2 = "0";          //保存清空前文本框的数值
private double result1 = 0.0;             //上行计算结果保存
private double result2 = 0.0;             //输入栏数字保存
private double result3 = 0.0;             //保存result2
private double M = 0.0;                   //存储功能

```

1.1 结对编程过程记录

(1) 角色切换与任务分工

表 2-1 第 2 轮迭代过程结对编程角色与任务分工

日期	时间(HH:MM - HH:MM)	驾驶员角色	领航员角色	本段时间的任务
2021/10/19	9:40-12:00	袁文字	惠丹妮	设置好历史记录的菜单选项, 搭建起历史记录页面面板。
2021/10/19	13:40-17:00	袁文字	惠丹妮	完善页面面板, 设置 list 存储历史记录, 增加添加历史记录的函数, 初步增加监听计算结果的功能。
2021/10/19	18:00-22:30	袁文字	惠丹妮	完善监听功能, 使得每条计算结果都能够加入历史记录。

(2) 工作日志

由领航员负责记录, 记录结对编程期间的遇到的问题、两人如何通过交流合作解决每个问题的。

表 2-2 第 2 轮迭代过程结对编程工作日志

日期/时间	问题描述	最终解决方法	交流过程
2021/10/19 11:10	采用 gridlayout 会使历史记录过少时,	采用 box 取代 gridlayout, 因为 box	不用 gridlayout 了, 它的布局就是平均分配页面空间,

日期/时间	问题描述	最终解决方法	交流过程
	一条记录占用一整个面板，记录间空隙过大	行高固定	与我们的需求不符，页面不好看。搜一搜有什么行高固定的方法。
2021/10/19 11:40	滚轮一直在那里不美观	将 ALWAYS 参数改成 AS_NEEDED	查一下滚轮设置的函数参数，也许有按需出现的功能。
2021/10/19 14:30	没办法跨类监听按钮	使 calculate 维护一个 history 实例，在 calculator 的监听函数中直接调用该实例的 add 函数	转换一下思维，也许没必要非要实现跨类监听也能达到效果，可以试试写个监听接口之类的。

(3) 结对编程工作现场照片

请其他同学帮助拍摄结对编程现场照片至少 2 张。



第 2 次迭代结对编程现场照片 1



第 2 次迭代结对编程现场照片 2

1.2 单元测试

对本轮迭代所开发的每个类进行单元测试。

表 2-3 第 2 轮迭代过程的单元测试记录

日期/时间	类名: 方法名	输入数据	输出数据	结论
2021/10/19 22:37	Calculator: UI()	点击菜单选项，选择“历史记录 (History)”	弹出的选项中有“历史记录 (History)”，点进去之后界面符合预期	功能正常
2021/10/19 22:40	History: addHistory(String)	在计算器面板连续点击 2+2=，并测试其他所有计算方法；最后测试 1/3*2=	在历史记录里面出现了字符串“2+2=4”，其他计算方法字符串也符合预期；最后一条比较长，	功能正常

日期/时间	类名: 方法名	输入数据	输出数据	结论
			为滚轮自动出现了	
2021/10/19 23:30	History: actionPerformed(ActionEvent)	在有几条历史记录的历史记录面板点击清除按钮, 重新进入该面板	历史记录被清除	功能正常

1.3 集成测试

对本轮迭代所开发功能模块进行测试。

表 2-4 第 2 轮迭代过程的集成测试记录

日期/时间	功能名称	测试用例	测试结果描述	结论
2021/10/19 23:40	历史记录纵向滚轮条功能+历史记录存储功能	进行大量数据计算	记录都被正确保存, 没有遗漏; 最后超出页面长度时, 纵向、横向滚轮条均出现	功能正常

5 项目总结

5.1 增量模型应用总结

对增量模型应用过程进行总结, 包括但不限于以下几方面的内容:

- 增量划分是根据工作量划分还是根据用户需求来划分?

根据用户需求。

- 第 1 个增量你们团队是如何确定的?

分析面向的群体可能具有的用户需求, 参考市面上已经存在的完善的计算器软件, 最终进行了决策。

- 你认为增量模型的好处在哪里? 有什么不好之处吗? 它的适用场合是什么?

好处在于可以将复杂的功能进行分段式实现, 最开始实现简单的可运行功能, 可以让用户快速了解产品风格并提出意见和建议, 不必等待产品完全做好。同时便于测试各项功能, 不易遗漏。

5.2 结对编程总结

对结对编程过程进行总结, 包括但不限于以下几方面的内容:

- 与 2 人分别编程相比, 是否体验到编程效率的提高、编程质量的提高? 明显感受到了这一点, 因为一个人编程容易陷入思维定式, 多一个人提

出不同角度的意见和建议能有效帮助代码的 debug 和功能实现，效率明显得到了提高。

● 你认为，你们 2 人本次合作是否真的提高了效率和质量？有什么成功的体验和需要反省的不足或教训？

确实提高了效率和质量。两个人结对编程，不容易存在偷懒的心理，看到队友完成的很认真，自己也受到了鼓舞和感动，同时两个人讨论问题加速了问题的解决、提高了问题解决的质量。

不足之处是，我们的沟通还不够彻底，比如偶尔会出现同时修改代码、难以整合的现象，所以还需要进一步加强沟通。感受到了沟通的重要性，没有沟通就没有团队协作。

● 你认为结对编程的优势在哪里、有什么不适应之处？它的适用场合是什么？

优势：能够提高问题解决的质量和效率，提高沟通讨论问题的能力，学习他人代码、思维的长处并提升自己。

不适应之处：从前都是自己编码，很少与人沟通交流，所有不是很适应每一步修改都需要与人商讨的编码方式。

适用场合：实验室、工作室等可以两个人讨论交流、对着同一台电脑进行编程的情况。