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** |
| 顾海耀 | 1190201423 | 1190201423@stu.hit.edu.cn |
| 王尧 | 1190201812 | 982045862@qq.com |

目 录

[1 项目概述 1](#_Toc26295)

[1.1 总体需求分析 1](#_Toc52)

[1.2 软件系统总体设计 2](#_Toc20834)

[1.3 开发任务计划 3](#_Toc12056)

[1.4 开发环境与工具 4](#_Toc25507)

[2 第1轮迭代：基本系统 4](#_Toc22703)

[2.1 迭代目标 4](#_Toc3696)

[2.2 需求分析 4](#_Toc1968)

[2.3 设计方案 4](#_Toc216)

[2.4 结对编程过程记录 6](#_Toc2390)

[2.5 单元测试 7](#_Toc27628)

[2.6 集成测试 7](#_Toc2091)

[3 第2轮迭代：增量1 8](#_Toc32480)

[3.1 迭代目标 8](#_Toc12823)

[3.2 需求分析 8](#_Toc9562)

[3.3 设计方案 8](#_Toc27658)

[3.4 结对编程过程记录 9](#_Toc8450)

[3.5 单元测试 10](#_Toc26543)

[3.6 集成测试 11](#_Toc8405)

[4 第3轮迭代：增量2 11](#_Toc21522)

[4.1 迭代目标 11](#_Toc19396)

[4.2 需求分析 12](#_Toc26653)

[4.3 设计方案 12](#_Toc8622)

[4.4 结对编程过程记录 13](#_Toc15962)

[4.5 单元测试 14](#_Toc27823)

[4.6 集成测试 14](#_Toc29527)

[5 项目总结 15](#_Toc3671)

[5.1 增量模型应用总结 15](#_Toc17570)

[5.2 结对编程总结 16](#_Toc3922)

[文档全部完成之后，请在上述区域点击右键，选择“更新域”，在打开的对话框中选择“更新整个目录”]

# 项目概述

（1）项目名称：实用仿真计算器

（2）项目简要介绍：（200字之内）

计算器是一种十分常用的学生工具，它能有效的帮助学生完成相应的计算，简化做题时间。我们通过增量模型进行计算器开发，做成一个具有实用意义与画面感较强的计算器

本项目让用户可以通过图形化计算器界面上按键来实现以下功能：

1. 在显示屏幕上输出用户选择的数据
2. 将用户选择的数据与选择的运算符号进行相应的计算，并将结果展示在显示屏幕中
3. 在用户选择数据和符号后发声提示，并将最终结果输出时念出数据的值

## 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）

目的与意义：

计算器是学生使用十分频繁的一个工具，学生对于计算的速度，计算结果的准确性以及计算使用的体验都有较高的要求。我们希望作为学生，在了解学生诉求后通过设计这样一个计算器，来实现学生的诉求

（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）。

学生（需求方）：

1. 计算器运算结果是否正确
2. 计算速度是否迅速
3. 计算机的使用是否舒服，即操作界面符合基本诉求

开发者（软件编程者）：

1. 软件开发过程是否简单
2. 软件的注释是否清晰，即是否可通过增量模型进行后续编程
3. 软件与不同操作系统的适配性是否高

监督人员（应用商店管理员）

1. 软件是否安全可靠，既不含有违规信息
2. 软件价格为多少

（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）。

从用户角度思考，我希望计算器能实现以下诉求：

1. 计算器的计算速度迅速
2. 计算器的结果准确且清晰
3. 计算器能回删输错的数据
4. 计算器的运算选择丰富，能适应大部分操作
5. 计算器的界面简洁美观，便于找到相应的运算符和数据
6. 计算器有暂时存档性，即一次运算后的结果能直接进行后续运算

（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：

1. 计算器的计算速度迅速
2. 计算器的结果准确且清晰
3. 计算器能回删输错的数据
4. 计算器有暂时存档性，即一次运算后的结果能直接进行后续运算

界面要求：

1.计算器的界面简洁美观，便于找到相应的运算符和数据

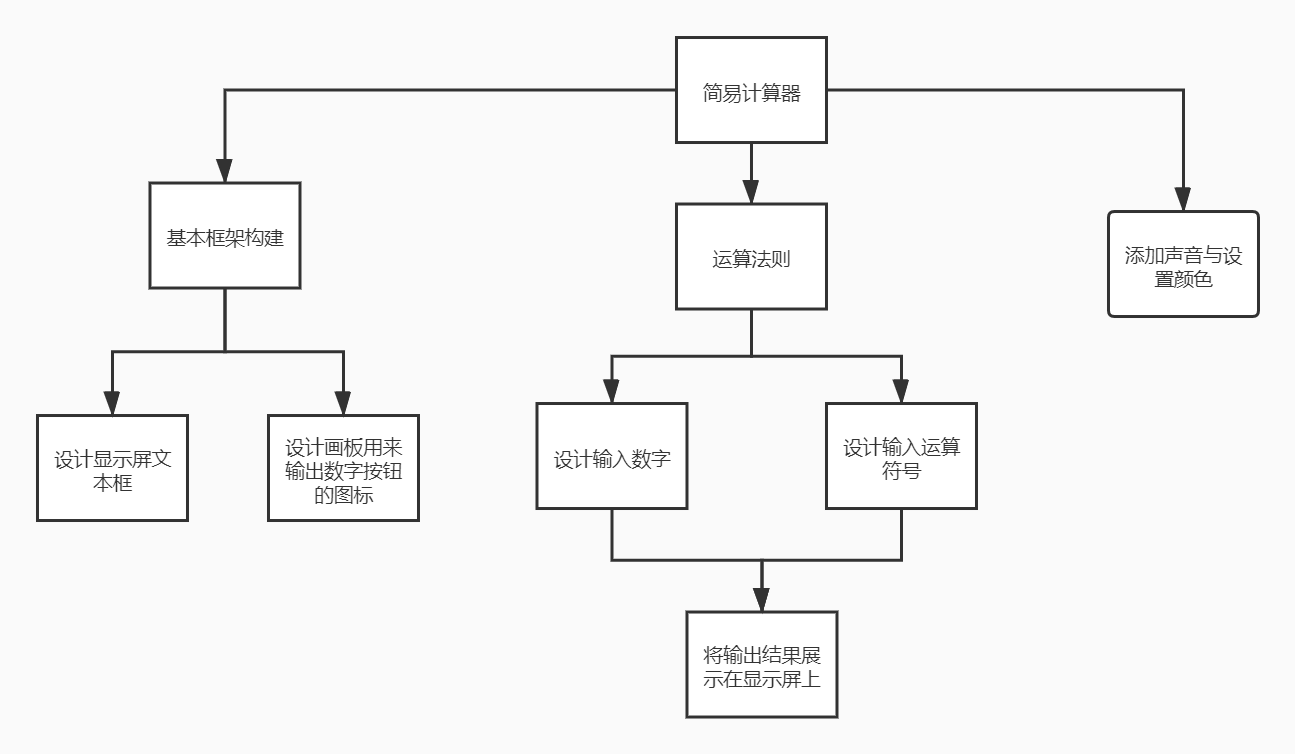
2.对于要实现的功能有着简单且清晰的按键

技术要求：  
1.软件适配多种操作系统

2.软件有着较好的可拓展性，便于后续迭代

## 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。



1. 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。

HandleAC 初始化界面

Handelenumber 确认输入的数字

Handeleoperator 确定输入的运算符号

Printresult 输出结果

Handlesaying 输出鼓励语句

Handlecolor 确定颜色

Handlesound 确定是否使用数字语音

变量定义：



## 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：完成整体的操作界面。

（2）第2轮迭代的任务：实现+,-,\*,/ 等功能。

（3）第3轮迭代的任务：实现换颜色，加数字声音等功能。

## 开发环境与工具

开发语言：java

开发环境：JDK 1.8

工具：IDEA

# 第1轮迭代：基本系统

## 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：设计计算器的窗口并将其初始化。

（2）任务2：添加文本框和画板

## 需求分析

本轮迭代所开发模块的详细需求分析。

本轮开发完成计算器的基本框架，即添加画板用来展示数字按钮，添加显示屏来输出运算结果，因此我们这次迭代要完成的就是计算器的图形化界面设计，

因此，此阶段我们完成的主要是calculator 中对计算器框架的设计，我们分为两部分，一部分是界面结构窗口的初始化，一个是对计算器的基础设定

## 设计方案

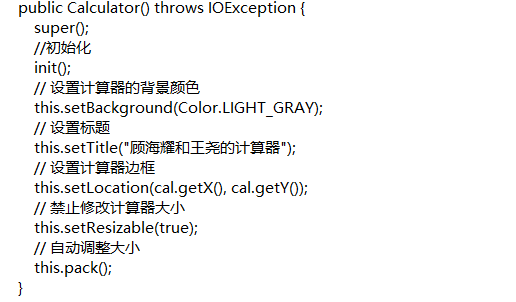
针对本轮迭代的开发内容给出设计方案：

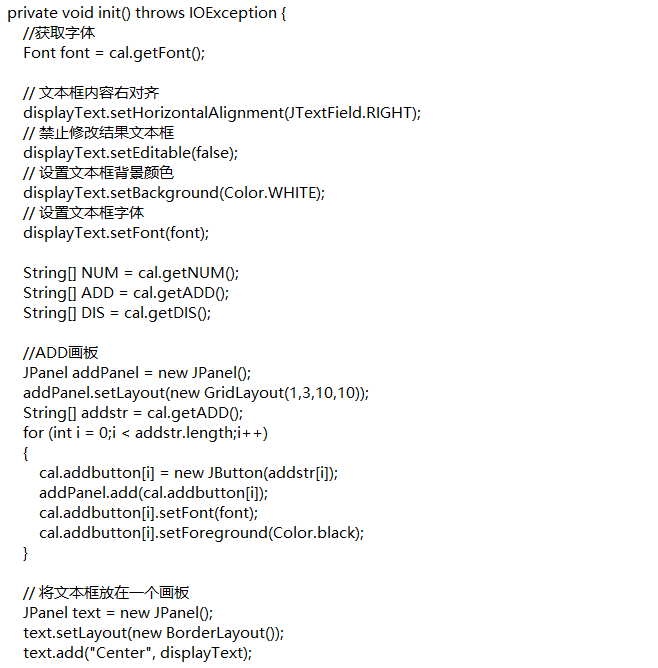
1. 详细设计方案：

图形界面预期如下：



1. 模块相关的设计类（Design Class）：。







## 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 |
| --- | --- | --- | --- | --- |
| 10.14 | 9：00-11；30 | 顾海耀 | 王尧 | 完成本实验变量的以及计算器窗口的初始化 |
| 10.14 | 15；30-16；30 | 顾海耀 | 王尧 | 完成计算器基本框架的设定 |

【注意】该表格可自行增加更多的行。

**（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。

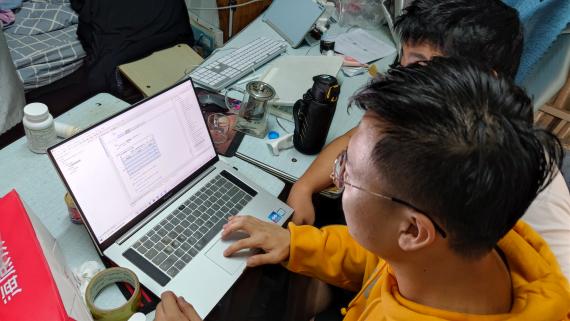
表1-2 第1轮迭代过程结对编程工作日志

| 日期/时间 | 问题描述 | 最终解决方法 | 交流过程 |
| --- | --- | --- | --- |
| 10.14 | 计算器界面的基本布局 | 采用大小不同的按键来区分不同功能的按键 | 双方结合自身小时候经历以及电脑上自带计算器的布局确定 |
| 10.14 | Swing的使用 | 学会正确使用Swing | 先自行学习，再双方交流 |
|  |  |  |  |

【注意】该表格可自行增加更多的行

**（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

## 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 |
| --- | --- | --- | --- | --- |
| 10.14 | 无 | 无 | 一个跟预期设想相同的表格 | True |

【注意】该表格可自行增加更多的行

## 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 |
| --- | --- | --- | --- | --- |
| 10.14 | 整体框架展示 | 无 | 表格正确展示 | True |

【注意】该表格可自行增加更多的行

表格展示：



# 第2轮迭代：增量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AC功能的设计

（2）任务2：完成对数据输入的设计。

（3）任务3：添加运算符号进行运算

## 需求分析

本轮迭代所开发模块的详细需求分析。

本轮要完成的是计算器的核心要求，即能初始化和进行运算，因此我们需要完成的便是关于这些要求的内容，我们将运算分为三部分，分别是清除，输入数据，运算符号的选择。

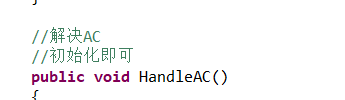
## 设计方案

针对本轮迭代的开发内容给出设计方案：

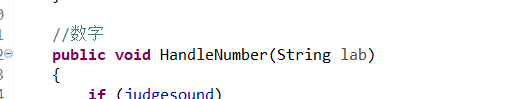
（1）详细设计方案：用三个方法来进行不同要求的实现，并在计算器中完成，在这些功能完成后，我们需要完成一些辅助功能，比如，从文本框中取出数据，将数据输出到文本框中等

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

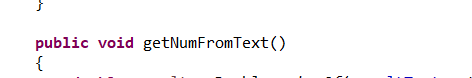
初始化AC



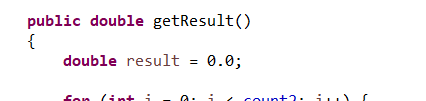
输入数据



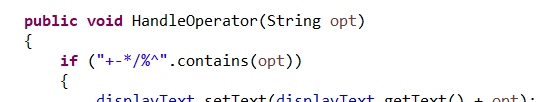
从文本框中取出数据



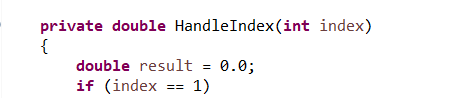
得到运算结果



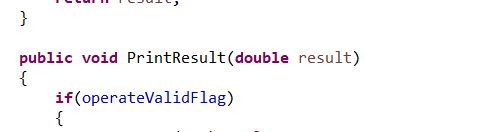
运算符号的选择和使用



处理除法（除法需要单独运算，因为有些时候需要先知道除数才能确定是否可以执行）



将数据输出到显示屏上



## 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 |
| --- | --- | --- | --- | --- |
| 10.16 | 14；00-17；30 | 王尧 | 顾海耀 | 完成AC以及数据的输入模块 |
| 10.16 | 19；00-21；30 | 王尧 | 顾海耀 | 完成运算符号的使用 |
| 10.17 | 8；30-11；45 | 顾海耀 | 王尧 | 完成其余辅助功能的书写 |

【注意】该表格可自行增加更多的行。

**（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。

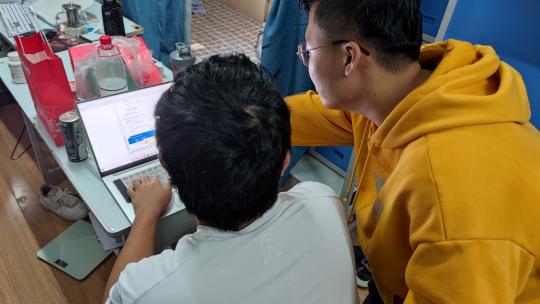
表2-2 第2轮迭代过程结对编程工作日志

| 日期/时间 | 问题描述 | 最终解决方法 | 交流过程 |
| --- | --- | --- | --- |
| 10.17 | 辅助功能与核心功能的统筹 | 在计算机方法中对这些方法进行统一使用 | 在csdn学习简单计算器的代码思路 |
| 10.16 | 按钮功能的实现，即确认用户选择的数据 | 添加监听器功能，来充当按钮反应功能 | 双方对于使用键盘进行数据还是图形界面进行了沟通 |

【注意】该表格可自行增加更多的行

**（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

## 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 |
| --- | --- | --- | --- | --- |
| 10.16 | HandleAC | 输入任意数字，按AC | 文本框为空 | True |
| 10.16 | HandleNum | 输入123 | 文本框显示为123 | True |
| 10.17 | Handleoperator | 输入7，按+，在输入5 | 文本框显示为12 | True |







## 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 | 类名：方法名 | 输入数据 | 输出数据 | 结论 |
| --- | --- | --- | --- | --- |
| 10.16 | 清零 | 输入任意数字，按AC | 文本框为空 | True |
| 10.16 | 输入数据 | 输入123 | 文本框显示为123 | True |
| 10.17 | 运算相加 | 输入7，按+，在输入5 | 文本框显示为12 | True |

【注意】该表格可自行增加更多的行

# 第3轮迭代：增量2

## 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：添加鼓励语句。

（2）任务2：添加换颜色功能。

（3）任务3：添加按键声音，并让用户决定是否开启

## 需求分析

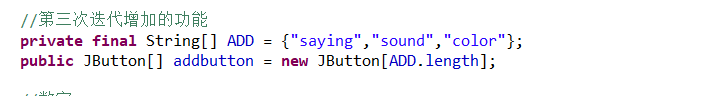
本轮迭代所开发模块的详细需求分析。

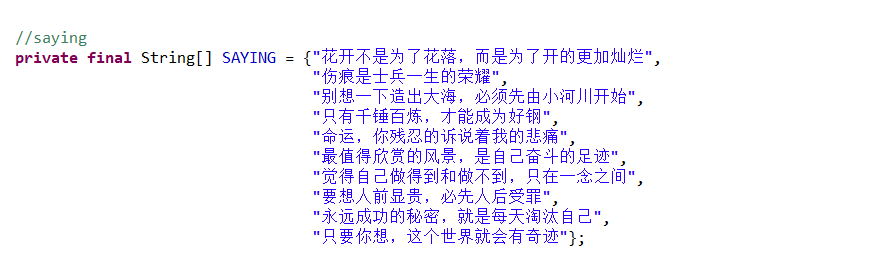
本阶段我们完成的是计算器的辅助功能，例如添加鼓励性话语，按键发生，以及更换颜色等。我们将这些也分成了三个部分。

## 设计方案

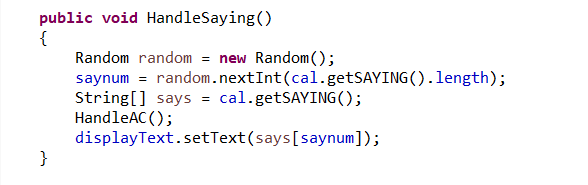
针对本轮迭代的开发内容给出设计方案：

1. 详细设计方案：模块功能结构设计，界面原型设计等。

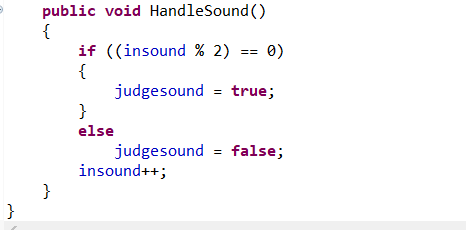




1. 模块相关的设计类（Design Class）：给出每个类的名字、属性、方法。







## 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 |
| --- | --- | --- | --- | --- |
| 10.18 | 8：00-11：30 | 王尧 | 顾海耀 | 完成saying 和 sound部分的代码 |
| 10.18 | 15；45-17；30 | 顾海耀 | 王尧 | 完成color部分的代码 |
| 10.18 | 18：30-20：00 | 顾海耀 | 王尧 | 实现项目的整体运行 |

【注意】该表格可自行增加更多的行。

**（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 |
| --- | --- | --- | --- |
| 10.18 | Color 切换的实现方式：关于是按一定颜色顺序更换还是由用户选择数字确认颜色未达成一致 | 按一定顺序更换颜色 | 我们双方观点不同，但是就按一定顺序更换颜色更加符合简易计算器的功能达成一致 |
| 10.18 | Saying 语句无法一次性完全输出 | 增加断句功能，实现一按一换 | 既然无法一次性输出，那就多次输出 |
| 10.18 | Sound 音频寻找 | 用qq 音乐完成 | 双方通过不同渠道进行寻找，最终确定 |

【注意】该表格可自行增加更多的行

**（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

## 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 |
| --- | --- | --- | --- | --- |
| 10.18 | Handlesaying | 点击saying 按 钮 | 语句按次序输出 | True |
| 10.18 | Handlecolor | 点击color 按 钮 | 颜色依次更换 | True |
| 10.18 | Handllesound | 点击sound 按钮然后点击数字键位 | 按键时有声音 | True |

【注意】该表格可自行增加更多的行

## 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 | 类名：方法名 | 输入数据 | 输出数据 | 结论 |
| --- | --- | --- | --- | --- |
| 10.18 | 鼓励语句输出 | 点击saying 按 钮 | 语句按次序输出 | True |
| 10.18 | 更换颜色 | 点击color 按 钮 | 颜色依次更换 | True |
| 10.18 | 发声 | 点击sound 按钮然后点击数字键位 | 按键时有声音 | True |







# 项目总结

## 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？

两个都要考虑到

从工作量划分来说，我们首先要完成的就是计算器的基本结构框架，然后我们需要完成的便是各种算法的加入还有颜色和声音的添加，根据工作量的划分，我们将算法与声音颜色划分到两个不同的迭代中，这样，我们基本的框架就建立起来了

从用户需求上来说，用户首先要求的便是一个便于理解的计算器界面，这个界面便是我们的第一次迭代的任务，也是我们计算器的基本框架，然后我们其次需要添加的便是计算器的核心功能，也就是运算功能，所以在第二次迭代中，我们需要完成的便是计算器的加减乘除功能。在第三次迭代中，我们需要完成的便是计算器的辅助功能，及声音与颜色功能

* 第1个增量你们团队是如何确定的？

在我们确立完基本框架后，我们发现我们的计算器并没有添加核心算法，

因此，我们决定在增量一中实现计算器的核心功能，即运算功能，在此功能下，我们决定添加+，-，\*，/，%，1/x，sin，cos 等运算功能。

* 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？

增量模型的好处：可以将一个大的任务划分为多个小任务并逐次实现，降低开发难度。

不好之处：

(一) 由于各个构件是逐渐并入已有的软件体系结构中的，所以加入构件必须不破坏已构造好的系统部分，这需要软件具备开放式的体系结构。在开发过程中，需求的变化是不可避免的。

(二) 增量模型的灵活性可以使其适应这种变化的能力大大优于瀑布模型和快速原型模型，但也很容易退化为边做边改模型，从而是软件过程的控制失去整体性。

适用场合：

用户急需部分核心功能，后续延伸功能暂时不急需。总的来说，适合于快速迭代的小项目。

## 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？

确实，在双方的协同下，双方轮流担任领航员和驾驶员，整个项目的整体方向和编程速度都有了保障，这样编程效率和编程的质量有了有效的提高。

* 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？

成功的体验：

1. 双方轮流做领航员和驾驶员对于项目的实现大有裨益
2. 结对编程在某一方陷入僵局时能更容易有所突破，双方的交流为项目的推进打下了基础

* 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：领航员可以帮助驾驶员确定方向，并及时发现程序中的错误然后查阅资料，反馈给驾驶员。
* 不适应：以往的编程实践都是个人独自编程，现在有领航员和驾驶员的身份互换，有些不太适应。
* 适合场合：双方的配合比较默契，适合快速迭代的小项目。