**实 验 报 告**

**课程名称** 移动应用系统

**实验项目**  基本移动程序界面设计

**实验仪器** 台式电脑

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实验报告评分 | | | | | |
| 一（15分） | 二（20分） | 三（30分） | 四（25分） | 参考文献（10分） | 总分 |
|  |  |  |  |  |  |

**专 业** 计算机科学与技术

**班级/学号** 计科1905/2018010636

**学生姓名** 殷绍轩

**实验日期** 2021-10-20

**成 绩**

**指导教师** 陈雷

实验**一** 基本移动程序界面设计

1. **需求分析（15分）**

功能要求： 用户输入一个复合表达式，这个表达式符合数学语法规范，如果用户输入错误要能够提醒用户并且把错误的输入删除。要能够计算基本的加减乘除，求相反数，sin正弦函数,cos余弦函数,sqrt开根号函数，以及复杂表达式的左括号、右括号，清零。

非功能要求：界面按钮交互友好，意图清晰，使用户能够一目了然界面如何使用以及软件的功能。程序具有一定健壮性，要能够处理各种潜在的异常。

设计约束：界面约束，使用相对约束，以此来适配不同型号和屏幕大小的设备。

1. **软件设计（20分）**

说明：请在这里根据需求分析的内容，进行软件设计。需要阐明App采用的架构和实现方法，如所使用的主要安卓组件，功能模块划分，数据库设计等。

1. 用户输入一个长表达式，最后点击“=”后进行计算。长表达式以String类型存储，每一次输入都是字符串的拼接。
2. 计算的时候把表达式从中缀表达式转成后缀表达式(逆波兰表达式)，利用Dijkstra的调度场(Shunting Yard)算法实现。计算的时候区分单目运算符(sin/cos/sqrt函数，求相反数)和双目运算符(加减乘除)。这两类运算符的逻辑是一样的。
3. 括号匹配：用最简单的括号计数器，输入了左括号+1，输入了右括号-1。计数器初始化为0，如果表达式正确，计算时候计数器数字应该为0。
4. 检错机制：用户每次输入都检查该输入是否合法。根据枚举情况，每个位置可以输入的类型是可以定的。
   1. 符号表示，数字：#，加减乘除：+，单目运算符：F，左括号：(，右括号：)，并列情况：/ (例如a/b表示a或b)以下为合法输入情况的列举。

|  |  |  |  |
| --- | --- | --- | --- |
| 输入类别🔘 | 上一次输入 | 当前输入 | 下一次输入 |
| 数字0～9 | #/+/( | # | #/+/) |
| 加减乘除 | #/) | + | #/( |
| 左括号 | (/+ | ( | (/# |
| 右括号 | )/# | ) | )/+ |
| 单目运算符 | +/F | F | (/F |

* 1. 列举出了输入的所有情况后，如果用户输入的情况有误，就把错误的输入从输入缓冲中删去，并且用Toast消息提示用户为什么有错。
  2. 为了方便，每次用户点击sqrt/sin/cos函数后自动添加左括号。并且在没有补齐右括号但是其他地方没有语法错误的时候，用户点击等号“=”自动补齐右括号并且计算。把交互做到最人性化。

1. 字符串匹配使用正则表达式。
2. **核心程序代码及运行结果（30分）**

本次实验的Github项目地址：https://github.com/ShaoxuanYin/Android/tree/Calculator

(文件夹是以submodule形式挂在github上，所以只有克隆后才能看到)

主要的文件MainActivity.java和activity\_main.xml单独列出来以供在线查看。

执行结果如下(照片顺序就是输入顺序)：

Calendar

Description automatically generated Calendar

Description automatically generated Calendar

Description automatically generated with medium confidence Calendar

Description automatically generated

Calendar

Description automatically generated Calendar

Description automatically generated with medium confidence

1. **实验中遇到的问题和解决方法（25分）**

本次实验中，使用了很多安卓特性，例如toast，activity，listener等。本次界面设计使用相对约束，以此来适配不同设备上的UI按钮大小。本次实验总共遇到过几个难题：1.表达式的输入与存储，使用了中间缓存机制。2.正则表达式进行匹配。3.检测错误的情况考虑。非常容易把情况考虑遗漏，以至于产生各种各样有意思的bug。使用了一些标志位来进行逻辑控制。代码写完将近3天，调试bug花费了3天。本次开发使用的是敏捷开发，让各个组件尽可能解耦合，让一个组件出问题不至于所有组件都出问题。非常期待后续的实验。

**参考文献（10分）**

[1] 安卓官网文档Documentation for app developers <https://developer.android.com/docs>

[2] 课本《第一行代码》郭霖