



# 信息与软件工程学院

## 项目报告

课程名称： 程序设计项目实践（PBLF）

学 期： 2023-2024 第 1 学期

项目名称： 计算器的设计与实现

授课教师： 张学

序号	学号	姓名
1（组长）	2023091602005	刘王哲
2	2023091602001	丁晨海
3	2023091602002	陆品安
4	2023091602003	卿胜
5	2023091602004	林嘉豪

## 目录

1 项目简介 .....	3
1.1 考核方式 .....	3
1.2 项目题目及内容简介 .....	3
1.3 项目组成员与分工 .....	3
2 需求分析 .....	4
2.1 选题的依据 .....	4
2.2 功能需求 .....	4
3 系统设计 .....	5
3.1 总体设计 .....	5
3.2 模块设计 .....	5
4.1 主函数 .....	7
4.2 其他函数 .....	7
5 测试 .....	8
6 总结 .....	9

# 1 项目简介

## 1.1 考核方式

总成绩 = 项目和项目文档成绩(40%) + 汇报幻灯片成绩(20%)  
+ 表达能力(20%) + 团队合作(20%)

## 1.2 项目题目及内容简介

用 C 语言实现计算器，主要是采用中缀转后缀的方法实现，并且在完成基本功能后还要增加通用栈，链式栈，批量计算，三角函数和反三角函数以及根号的实现等，最后还有进行计算器对象的实现以及打包

## 1.3 项目组成员与分工

1.组员刘王哲：负责团队协调员与实现计算器对象

确保项目进度按计划进行。协调团队成员的工作，解决可能出现的问题。负责项目文档的管理，包括头文件和源文件、技术文档等。在队友设计好计算器以后对计算器对象进行打包与实现。

2.组员丁晨海：负通用栈的实现

负责设计和实现通用栈的基本功能。设计并实现通用栈的基本函数与操作等功能。确保栈的基本功能是稳定可靠的。

3.组员陆品安：负责链式栈的实现

设计和实现链式栈结构。处理链表的创建、节点的插入、删除、链表的遍历等操作。增加通过链表实现的链式栈函数，与通用栈协调，确保接口一致性。

4.组员卿胜：负责批量计算的模块

实现计算器的批量计算功能，包括运用流指针读取文件，改写文件等。处理输入表达式的解析和计算。确保对于复杂表达式的正确处理。

5.组员林嘉豪：负责工程文件的管理与计算器更多功能的实现

负责项目的构建系统，如源文件头文件的分类与规划等。处理编译、构建和部署流程。增添例如三角函数，反三角函数以及求平方根等功能。

## 2 需求分析

### 2.1 选题的依据

#### 1.综合应用方面：

通过实现一个计算器项目，学生能够综合应用在 C 语言编程中所学到的知识，包括基本语法、数据结构、算法等。

#### 2.算法理解方面：

中缀转后缀是一种经典的算法问题，通过实现这一功能，学生可以深入理解栈的应用，并提高对基本算法的理解和实现能力。

#### 3.数据结构的实际应用方面：

强调通用栈和链式栈的设计和实现，加深学生对于栈这一数据结构的理解和应用。引入批量计算功能，使学生能够处理多个表达式的输入，提高程序的实用性和用户体验。

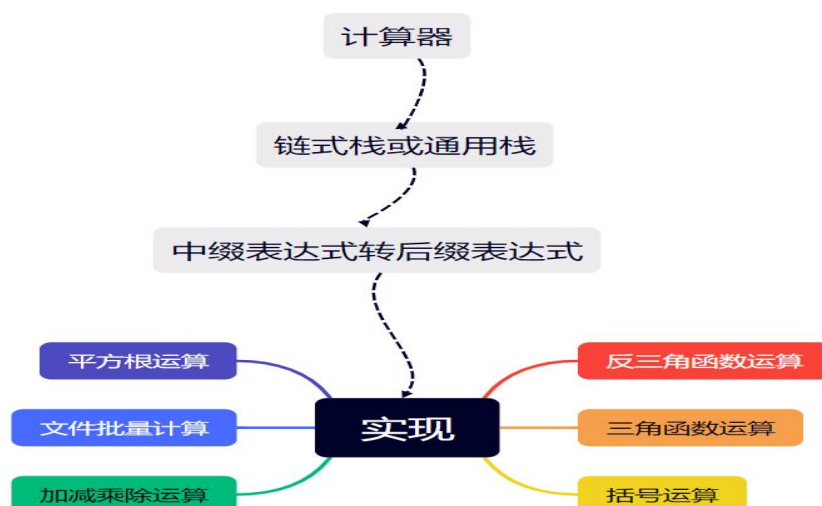
#### 4.数学函数的实现方面：

通过添加三角函数和反三角函数，以及根号的实现，拓展了计算器的功能，同时加深学生对于数学函数在程序中的具体实现方法的了解。

#### 5.实际应用和工程化思维方面：

整个项目涉及多个方面，包括算法、数据结构、功能拓展和项目管理等，使学生在实际应用中能够形成综合性的解决问题的思维模式，培养工程化思维。通过这样一个项目，学生将能够深入了解 C 语言的各个方面，并在实践中掌握综合性的编程能力。此外，项目的可扩展性和模块化设计可以让学生更好地理解软件工程的思想。

### 2.2 功能需求



## 3 系统设计

### 3.1 总体设计

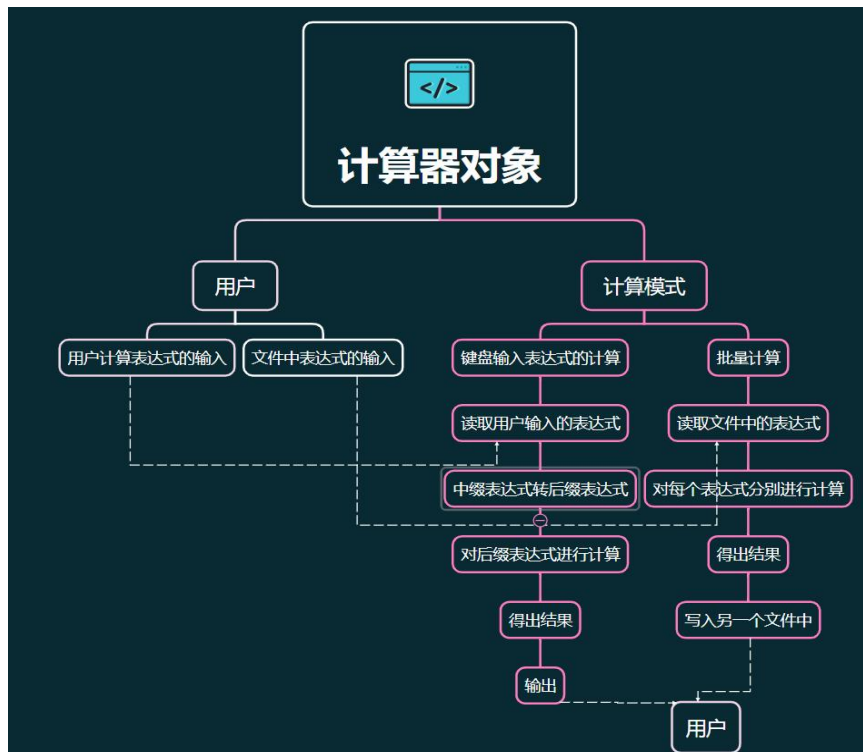


图 3-1

### 3.2 模块设计

文件 1: main.c

执行文件，主函数，通过引用计算器对象的函数实现计算器的运行

文件 2: common.h

头文件，定义 Status 类型和 Bool 类型，并使其他程序能够共享此定义

文件 3: Generic\_Stack.h

头文件，包含宏定义，类型定义，外部变量声明，栈的函数接口与必要的注释

#### 文件 4: Generic\_Stack.c

源文件，对 Generic\_Stack.h 头文件中函数接口进行实现，包括具体的函数实现和必要的文件包含

#### 文件 5: Caculator.h

头文件，包含宏定义，类型定义，外部变量声明，有关计算器对象的函数接口与必要的注释

#### 文件 6: Caculator.c

源文件，对 Caculator.h 头文件中函数接口进行实现，包括具体的计算器对象函数（键盘输入计算与批量计算）实现和必要的文件包含，其中实现计算的函数需要有中缀表达式转后缀表达式的过程，并且需要通过栈的函数进行实现

#### 文件 7: Link.h

头文件，头文件，包含宏定义，类型定义，外部变量声明，有关链表的函数接口与必要的注释

#### 文件 8: Link.c

源文件，源文件，对 Link.h 头文件中函数接口进行实现，包括具体的函数实现和必要的文件包含

#### 文件 9: Linked\_Stack.h

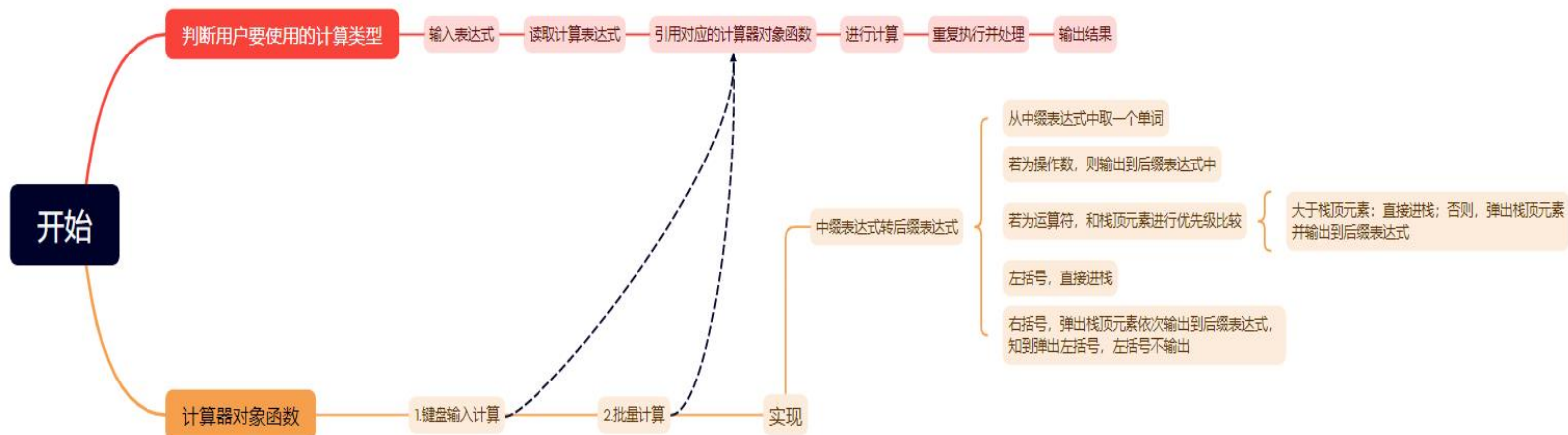
头文件，头文件，包含宏定义，类型定义，外部变量声明，有关链式栈的函数接口与必要的注释

#### 文件 10: Linked\_Stack.c

源文件，通过继承 Link.c 的函数接口，使用链表中的函数来对 Linked\_Stack.h 头文件中链式栈函数接口进行实现，包括具体的函数实现和必要的文件包含

## 4 系统实现

### 4.1 主函数

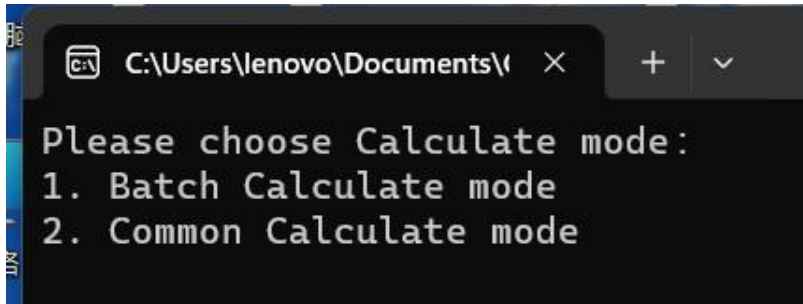


### 4.2 其他函数



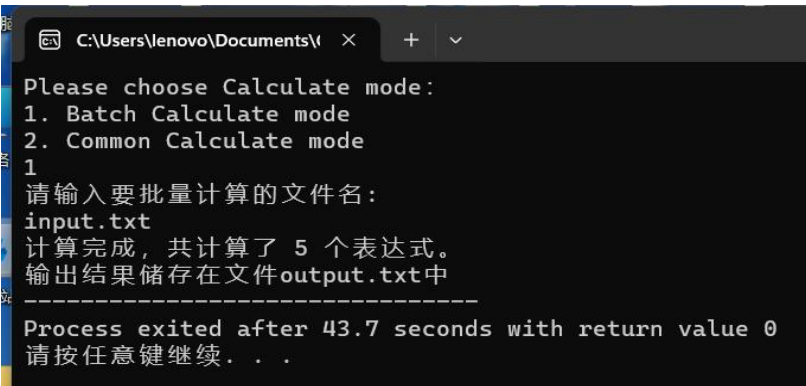
## 5 测试

开始:



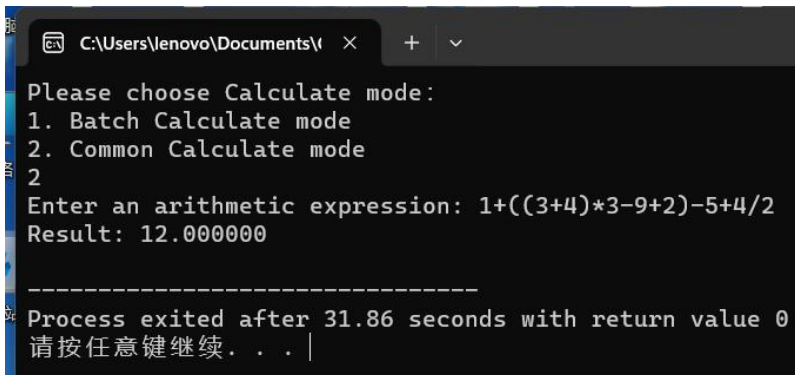
```
C:\Users\lenovo\Documents\l >
Please choose Calculate mode:
1. Batch Calculate mode
2. Common Calculate mode
```

批量计算:



```
C:\Users\lenovo\Documents\l >
Please choose Calculate mode:
1. Batch Calculate mode
2. Common Calculate mode
1
请输入要批量计算的文件名:
input.txt
计算完成, 共计算了 5 个表达式。
输出结果储存在文件output.txt中
-----
Process exited after 43.7 seconds with return value 0
请按任意键继续. . .
```

键盘输入计算:



```
C:\Users\lenovo\Documents\l >
Please choose Calculate mode:
1. Batch Calculate mode
2. Common Calculate mode
2
Enter an arithmetic expression: 1+((3+4)*3-9+2)-5+4/2
Result: 12.000000
-----
Process exited after 31.86 seconds with return value 0
请按任意键继续. . . |
```

批量计算结果:



```
input.txt
文件 编辑 查看
1+((3+5)*7-8)-5+9
sin0.5
arcsin0.999
2^(1/2)
arcsin(2-1)
```



```
output.txt
文件 编辑 查看
53.000000
0.479426
1.526071
1.414214
1.570796
```



## 6 总结

刘王哲-2023091602005

你的总结：

通过此次的 C 语言计算器项目的实现，我学到了很多 C 语言知识，包括堆栈的实现，通用栈，链式栈，链表的实现，以及中缀表达式转后缀表达式，逆波兰式的计算等，在后来的进一步完善中又增添了批量计算，计算器对象的实现等等新的内容。

学到了什么？

首先我熟悉了 C 语言的基本语法和编程结构，其次我掌握了程序设计中与用户交互的基本原理，而且我也学会了处理数学运算和逻辑操作，最后我也理解了一些程序中的错误处理和异常情况的处理方法。

痛点和难点：

首先是用户输入处理，关于如何有效地处理用户输入，确保输入的合法性和正确性，我思考了很多，也遇到了很关于读取，输入等问题；其次就是数学运算的实现，我通过自己查阅资料，解决如何正确执行各种数学运算，包括加法、减法、乘法和除法，以及后来的三角函数，平方根，反三角函数等；最后就是关于错误处理的技巧，我学会了处理用户可能输入的错误情况，如除以零、非法字符等。

自己的贡献：

我通过自己的发掘思索，独立完成了计算器项目的设计和实现，优化了代码结构，提高了程序的可读性和可维护性，不仅如此，我还添加了额外的功能，如批量计算，计算器对象的实现等，最后，我解决了一些关键性的 bug，确保程序的稳定性。

如何与他人合作？

首先我掌握了一些沟通技巧，我学到了与团队成员有效沟通的重要性，包括分享进展、寻求帮助和提供反馈等，其次我学会了团队协作，学习如何在团队中分工合作，充分发挥每个成员的专长，最后我也学会了接受反馈，我能够接受并理解他人的建议和反馈，以改进代码和项目。

## 丁晨海-2023091602001

学到了什么？

在编写计算器程序的过程中，我学到了许多关于编程的重要概念和技能。首先，我加深了对 C 语言的理解，并学会了如何使用不同的数据类型、循环和条件语句来构建一个功能完善的程序。我还学会了如何处理用户输入、执行基本的数学运算，并设计用户友好的界面来提高程序的易用性。

痛点和难点：

在编写计算器程序时，我遇到了一些挑战和困难。其中一个痛点是如何有效地处理用户输入错误和异常情况。确保程序能够稳健地处理各种可能出现的错误是一个具有挑战性的任务。另一个难点是设计程序的逻辑，使其在处理不同类型的运算时能够正确执行，并保持代码的简洁性和可读性。

自己的贡献：

在这个项目中，我贡献了程序的整体架构设计、基本运算功能的实现以及用户界面的开发。我花了大量的时间来优化代码，确保其效率和稳定性。同时，我也专注于处理输入异常和错误情况，以提高程序的健壮性和用户体验。

如何与他人合作？

在开发过程中，我与团队成员进行了有效的沟通和合作。我们定期举行会议，讨论程序设计的细节并共享想法。我乐于倾听他人的建议，并乐意与团队协作解决问题。通过分工合作，我负责了程序的核心功能，同时也与其他成员合作，确保整个项目的顺利进行。

## 陆品安-2023091602002

学到了什么？

经过这次计算器实践，我学到了合作编码大型程序的技巧、C 语言编写的规范和一些 C 语言编码的技巧。

痛点和难点：

链式栈、通用栈的设计和运用；计算器计算思路。比如：刚开始实现计算器时，即使知道需要使用中缀转后缀的方法，我们依旧无法熟练编写计算程序、没有快速完成中缀转后缀的接口处理。

自己的贡献：

设计和实现链式栈结构。处理链表的创建、节点的插入、删除、链表的遍历等操作。增加通过链表实现的链式栈函数，与通用栈协调，确保接口一致性。

如何与他人合作？

与组员有效沟通、积极交流、分析自己的思路、接受他人建议。

## 卿胜-2023091602003

学到了什么？

我学会运用 C 语言编写计算器，同时也增强团队协作能力和与他人交流的能力。同时对 C 语言的基本知识有了一定的掌握。

痛点和难点：

程序编好并未报错，但无法运行，更改许多次才能运行，但计算结果仍有问题，最后与同学讨论才解决。

自己的贡献：

通过与同学之间的探讨，使计算器结构更加符合规定，能让人更好读懂程序。

如何与他人合作？

和他人分享计算机的设计思路，在写程序出现问题时与同学积极探讨并进行改错。

## 林嘉豪-2023091602004

学到了什么？

首先，我明白了栈是一种特殊的数据结构，遵循后进先出的原则。理解了栈的工作原理对于理解程序如何处理数据和执行操作至关重要。计算器程序需要使用各种算法来处理输入、执行计算和输出结果。这可能包括基本的算术运算、优先级处理、括号处理等算法。通过编写栈计算器程序，我更深入地理解 C 语言的特性，包括变量、数据类型、函数、指针等。

痛点和难点：

在这样一个程序的编写中，我遇到的痛点和难点包括栈的编写，栈功能的实现，运算符的优先级处理和括号的匹配。更包括非法字符的输入处理，反三角函数运算的识别。

自己的贡献：

项目的构建系统，如源文件头文件的分类与规划等。处理编译、构建和部署流程。增添例如三角函数，反三角函数以及求平方根等功能。

如何与他人合作？

如何与他人合作：我们组内分工明确，各司其职，组长分工后每个人都尽力去实现自己的功能，最后调试代码，将预期的效果实现。