

## 目录

一、课程目标：	.....
二、小组分工：	.....
三、项目介绍	.....
1. 项目简介	.....
1.1 背景：	.....
1.2 题目描述：	.....
1.3 理论研究基础	.....
1.4 技术层面的支持	.....
2. 需求获取	.....
2.1 功能需求与非功能需求	.....
3. 可行性分析	.....
3.1 社会角度	.....
3.2 经济角度	.....
3.3 技术角度	.....
四、系统分析	.....
1. 系统功能图	.....
2. 类图	.....
3. 系统流程图	.....
五、系统设计	.....
1. 类说明；	.....
2. 关键类及其函数设计	.....
六、系统实现	.....
七、结论	.....
1. 个人感受：	.....
2. 存在问题及改进方法：	.....

3. 可以改进的地方 .....	
八、参考文献 .....	

## 一、课程目标：

- 1、能够针对设计题目，通过调研或者查找资料，进行初步需求分析和系统总体设计的能力。
- 2、能够利用面向对象程序设计的思想完成系统的设计并编程实现。
- 3、能够根据设计工作量和人员特点进行合理分工。小组人员配合默契，能够互相交流，能够完成个人承担的任务，并且整个项目能够按时完成和保证质量。
- 4、能够对设计的内容进行表述，并回答老师和同学提出的问题。
- 5、能够绘制类图和用例图，具有按照软件工程思路撰写完整的课程设计报告。
- 6、能够对设计过程进行总结，能够意识到不足，并具有改进的意识。

## 二、小组分工：

组员	任务
组长及组员：	<p>1、设计整体框架及界面，设计并编写所需要的所有功能程序，调试程序并完善程序。</p> <p>2、设计并编写编辑功能程序，调试程序，制作汇报 ppt，制作报告，报告的功能图、类图和流程图等绘制。</p>

## 三、项目介绍

### 1. 项目简介

#### 1.1 背景：

随着人们物质生活水平的日益提高，人们对高质量的计算提出了更高的要求。当今世界，是云计算的生活和物联网发展和使用阶段，计算性能的高低和计算的效率直接影响到了人们的生活。对此，使用 C++ 语言开发出一个为人们生活带来便利的分数计算器，主要完成对分数的高效计算。该系统实现了分数与整数、分数与分数的加、减、乘、除的四则运算及分数化简、取倒数和比较大小的功能。本系统已经通过了全面的系统测试，能够很好的运行，达到了预期的效果。

## 1.2 题目描述:

定义一个整数类。

定义一个分数类，由整数类派生。能对分数进行各种计算和输入/输出。

## 1.3 理论研究基础

- 1 综合 C++编程知识，结合实际加以应用。
- 2 结合 VS 2019 进行程序的调试和修改。

## 1.4 技术层面的支持

VS 2019

坚果云绘图

## 2. 需求获取

### 2.1 功能需求与非功能需求

**功能需求:**

(1) 定义整数类和分数类。其中，包括构造函数、析构函数、显示函数等。

(2) 输入/输出：对流提取和流插入运算符进行重载。

(3) 计算功能：可进行分数的加、减、乘和除法运算。

(4) 化简功能：将分数化简为最简分数。

(5) 异常处理功能：分数中分母不能为零。

(6) 菜单功能：每种功能的操作都是在菜单中进行相应选择。

**非功能需求:**

功能性：适合性（每个子功能都要符合有关的软件属性）；

准确性（需要有准确的操作）；

易用性：需要易理解易使用的分数计算器系统。

## 3. 可行性分析

### 3.1 社会角度

随着电子信息的高速发展，人们的生活也越来越便捷，越来越有效。人们对便捷生活的需求也越来越明显。随着人们物质生活水平的日益提高，人们对高质量的计算提出了更高的要求。当今世界，是云计算的生活和物联网发展和使用阶段，计算性能的高低和计算的效率直接影响到了人们的生活。对此，使用 C++语言开发出一个为人们生活带来便利的分数计算器。该软件的开发不会侵犯国家、集体和他人的利益。

### 3.2 经济角度

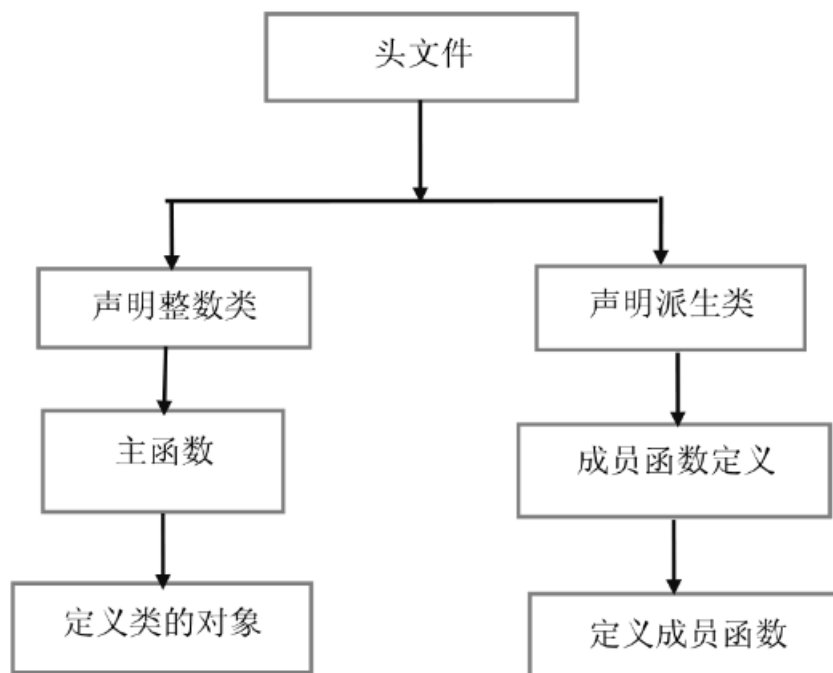
该软件的开发，对于系统的制作， 所需的经济代价很低。后期系统的维护需要少许的费用。总体来说，从经济角度来说，该系统的开发可行。

### 3.3 技术角度

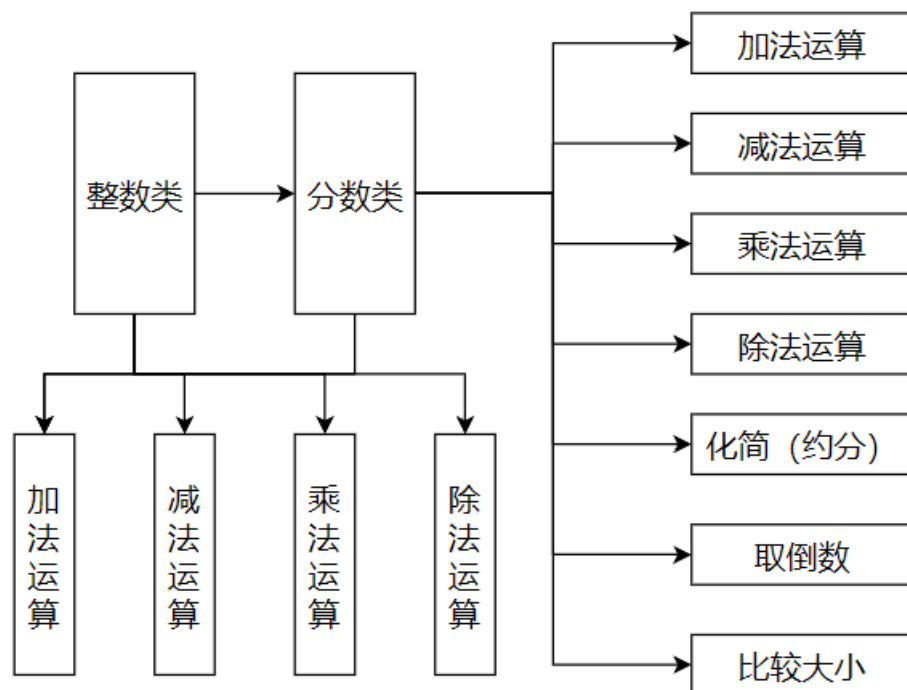
现代技术的高速发展，对开发一个简单的分数计算器系统根本不是问题，相反，我们是要更好的利用技术去渗透我们的生活，改善我们的生活。

## 四、系统分析

### 1. 系统功能图：



## 2. 类图:



## 3. 系统流程图:

进入程序→选择菜单功能→进行计算。

## 五、系统设计

在接下来的介绍中将详细的介绍每个关键的类和类中重要的函数起到的作用，整个功能运行的流程，并且有详略的介绍重点函数是如何实现的。

### 1. 类说明

```
class Integer          //整数类

class Fraction :public Integer    //分数类，由整数类派生

    friend istream & operator>>(istream &, Fraction &);          //
重载输入流

    friend ostream & operator<<(ostream &, Fraction &);          //
重载输出流
```

## 2. 关键类及其函数设计

```
Integer(int a = 0, int b = 1) :fenzi(a), fenmu(b) {};    //构造函数
~Integer() {};                                           //析构函数
void display();                                          //显示函数

Fraction(int a=0,int b=1):Integer(a,b) {}              //分数类构造函数

Fraction operator+(const Fraction& c);                  //重载+运算符（分数与分数）

Fraction operator+(int n);                             //重载+运算符（整数与分数）

Fraction operator-(const Fraction& c);                  //重载-运算符（分数与分数）

Fraction operator-(int n);                             //重载-运算符（整数与分数）

friend Fraction operator-(int n, const Fraction& c);    //重载-运算符（分数与整数）

Fraction operator*(const Fraction& c);                  //重载*运算符（分数与分数）

Fraction operator*(int n);                             //重载*运算符（整数与分数）

Fraction operator/(const Fraction& c);                  //重载/运算符（分数与分数）

Fraction operator/(int n);                             //重载/运算符（整数与分数）
```

```
friend Fraction operator/(int n, const Fraction& c);    //
重载/运算符（分数与整数）
```

```
Fraction operator~();    //取倒数一目运算符
```

```
//判断大小
```

```
bool operator>(const Fraction& c);
```

```
bool operator<(const Fraction& c);
```

```
bool operator==(const Fraction& c);
```

```
bool operator!=(const Fraction& c);
```

```
bool operator>=(const Fraction& c);
```

```
bool operator<=(const Fraction& c);
```

```
Fraction reduce();    //约分函数
```

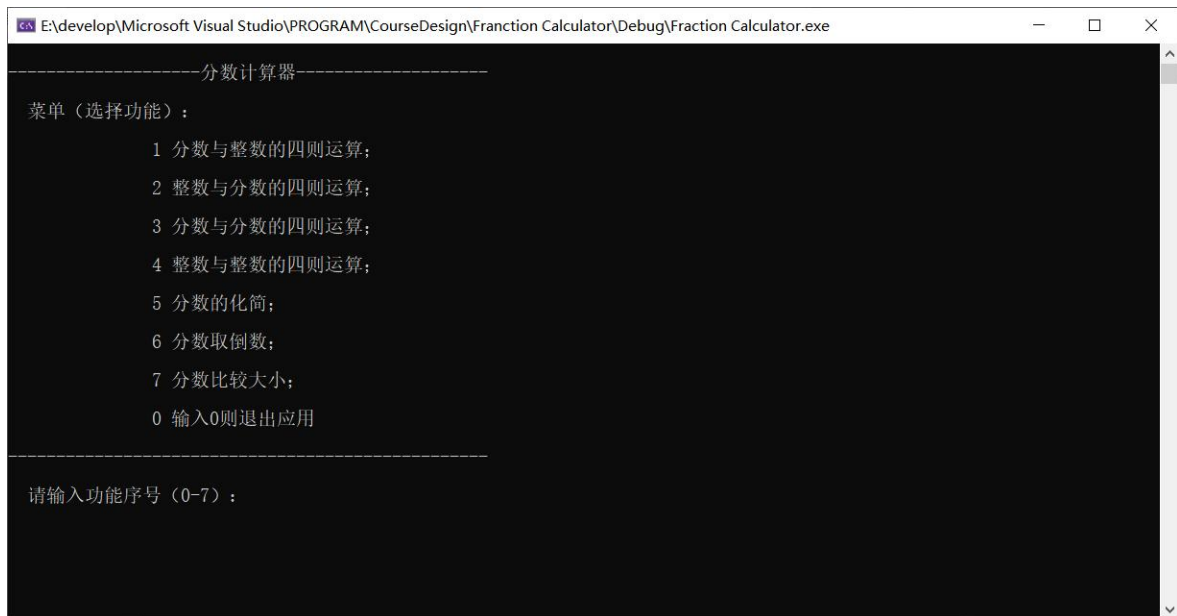
```
~Fraction() {}    //析构函数
```

```
void display();    //开头界面显示函数
```

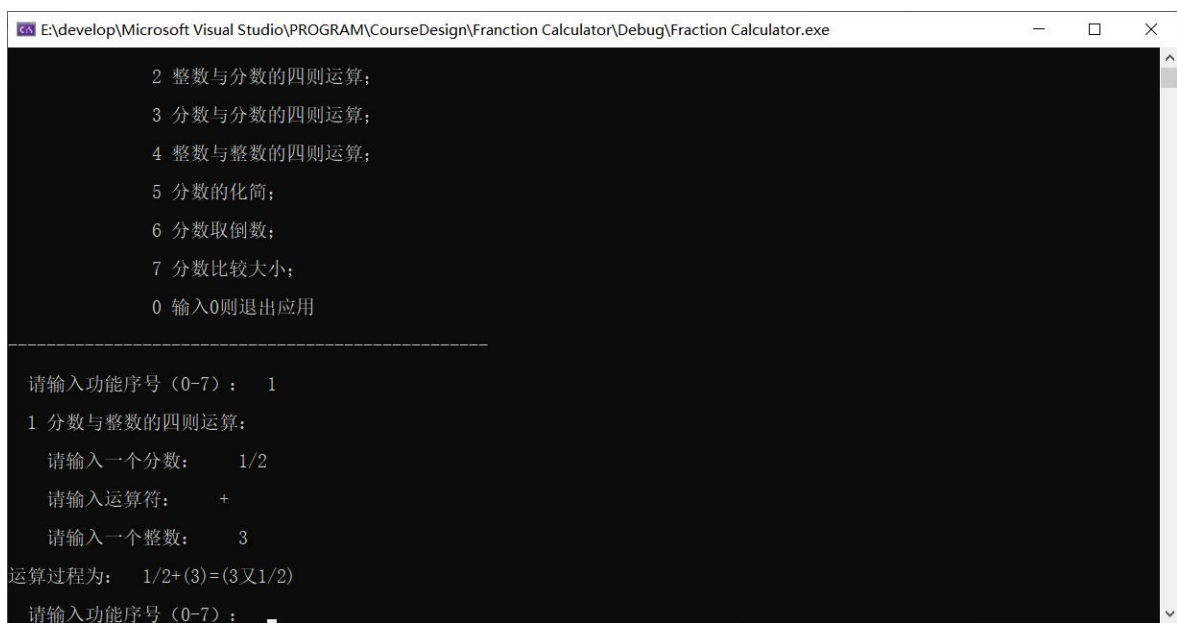
## 六、系统实现

Ps:以运行过程为主线，逐渐显示功能的实现





进入开始菜单界面



分数与整数的加法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

0 输入0则退出应用

-----

请输入功能序号 (0-7): 1
1 分数与整数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: +
  请输入一个整数: 3
运算过程为: 1/2+(3)=(3又1/2)
请输入功能序号 (0-7): 1
1 分数与整数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: -
  请输入一个整数: 3
运算过程为: 1/2-(3)=(-2又1/2)
请输入功能序号 (0-7):
```

分数与整数的减法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

  请输入一个整数: 3
运算过程为: 1/2+(3)=(3又1/2)
请输入功能序号 (0-7): 1
1 分数与整数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: -
  请输入一个整数: 3
运算过程为: 1/2-(3)=(-2又1/2)
请输入功能序号 (0-7): 1
1 分数与整数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: *
  请输入一个整数: 3
运算过程为: 1/2*(3)=(1又1/2)
请输入功能序号 (0-7):
```

分数与整数的乘法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

2 整数与分数的四则运算;
3 分数与分数的四则运算;
4 整数与整数的四则运算;
5 分数的化简;
6 分数取倒数;
7 分数比较大小;
0 输入0则退出应用

-----

请输入功能序号 (0-7): 1
1 分数与整数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: /
  请输入一个整数: 3
运算过程为: 1/2/(3)=1/6
请输入功能序号 (0-7):
```

分数与整数的除法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

0 输入0则退出应用

-----

请输入功能序号 (0-7): 1
1 分数与整数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: /
  请输入一个整数: 3
运算过程为: 1/2/(3)=1/6
请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
  请输入一个整数: 1
  请输入运算符: +
  请输入一个分数: 1/2
运算过程为: 1+1/2=(1又1/2)
请输入功能序号 (0-7):
```

整数与分数的加法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

请输入一个整数: 3
运算过程为: 1/2/(3)=1/6
请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
请输入一个整数: 1
请输入运算符: +
请输入一个分数: 1/2
运算过程为: 1+1/2=(1又1/2)
请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
请输入一个整数: 2
请输入运算符: -
请输入一个分数: 1/2
运算过程为: 2-1/2=(1又1/2)
请输入功能序号 (0-7):
```

### 整数与分数的减法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

请输入一个分数: 1/2
运算过程为: 1+1/2=(1又1/2)
请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
请输入一个整数: 2
请输入运算符: -
请输入一个分数: 1/2
运算过程为: 2-1/2=(1又1/2)
请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
请输入一个整数: 2
请输入运算符: *
请输入一个分数: 1/2
运算过程为: 2*1/2=1
请输入功能序号 (0-7):
```

### 整数与分数的乘法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

2 整数与分数的四则运算;
3 分数与分数的四则运算;
4 整数与整数的四则运算;
5 分数的化简;
6 分数取倒数;
7 分数比较大小;
0 输入0则退出应用

-----

请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
  请输入一个整数: 2
  请输入运算符: /
  请输入一个分数: 1/2
运算过程为: 2/1/2=4
请输入功能序号 (0-7):
```

整数与分数的除法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

0 输入0则退出应用

-----

请输入功能序号 (0-7): 2
2 整数与分数的四则运算:
  请输入一个整数: 2
  请输入运算符: /
  请输入一个分数: 1/2
运算过程为: 2/1/2=4
请输入功能序号 (0-7): 3
3 分数与分数的四则运算:
  请输入一个分数: 1/2
  请输入运算符: +
  请输入一个分数: 1/2
运算过程: 1/2+1/2=1
请输入功能序号 (0-7):
```

分数与分数的加法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe
请输入一个分数: 1/2
运算过程: 2/1/2=4
请输入功能序号 (0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: +
请输入一个分数: 1/2
运算过程: 1/2+1/2=1
请输入功能序号 (0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: -
请输入一个分数: 1/2
运算过程: 1/2-1/2=0
请输入功能序号 (0-7):
```

### 分数与分数的减法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe
请输入一个分数: 1/2
运算过程: 1/2-1/2=0
请输入功能序号 (0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: *
请输入一个分数: 3
/2
运算过程: 1/2*(1又1/2)=3/4
请输入功能序号 (0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: *
请输入一个分数: 2/3
运算过程: 1/2*2/3=1/3
请输入功能序号 (0-7):
```

### 分数与分数的乘法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe
请输入一个分数: 3
/2
运算过程: 1/2*(1又1/2)=3/4
请输入功能序号(0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: *
请输入一个分数: 2/3
运算过程: 1/2*2/3=1/3
请输入功能序号(0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: /
请输入一个分数: 3/2
运算过程: 1/2/(1又1/2)=1/3
请输入功能序号(0-7):
```

## 分数与分数的除法运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe
请输入一个分数: 3/2
运算过程: 1/2/(1又1/2)=1/3
请输入功能序号(0-7): 4
4 整数与整数的四则运算:
请输入一个整数: 1
请输入运算符: +
请输入一个整数: 2
运算过程: 1+2=3
请输入功能序号(0-7): 4
4 整数与整数的四则运算:
请输入一个整数: 1
请输入运算符: -
请输入一个整数: 2
运算过程: 1-2=-1
请输入功能序号(0-7):
```

## 整数与整数的四则运算调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

请输入一个整数:    2
运算过程:    1-2=-1
请输入功能序号 (0-7):    5
5 分数的化简:
    请输入一个分数:    4/6
化简结果:    2/3
    请输入功能序号 (0-7):    _
```

分数化简（约分）调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

菜单（选择功能）：
    1 分数与整数的四则运算；
    2 整数与分数的四则运算；
    3 分数与分数的四则运算；
    4 整数与整数的四则运算；
    5 分数的化简；
    6 分数取倒数；
    7 分数比较大小；
    0 输入0则退出应用
-----
请输入功能序号 (0-7):    6
6 分数取倒数:
    请输入一个分数:    1/2
取倒结果:    2
    请输入功能序号 (0-7):    _
```

分数取倒数调试



```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

5 分数的化简;
6 分数取倒数;
7 分数比较大小;
0 输入0则退出应用

-----

请输入功能序号 (0-7): 6
6 分数取倒数;
  请输入一个分数: 1/2
取倒结果: 2
请输入功能序号 (0-7): 7
7 分数比较大小;
  请输入一个分数: 1/2
  请输入另一个分数: 3/4
1/2前者小于后者3/4
请输入功能序号 (0-7):
```

## 分数比较大小调试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe

取倒结果: 2
请输入功能序号 (0-7): 7
7 分数比较大小;
  请输入一个分数: 1/2
  请输入另一个分数: 3/4
1/2前者小于后者3/4
请输入功能序号 (0-7): 3
3 分数与分数的四则运算;
  请输入一个分数: 1/0
  分母为0, 请重新输入!
1/2
  请输入运算符: +
  请输入一个分数: 1&2
  错误! 输入格式错误 (请输入格式形如 m/n) !
1/2
运算过程: 1/2+1/2=1
请输入功能序号 (0-7):
```

## 分数分母为 0 或分号/符号错误测试

```
E:\develop\Microsoft Visual Studio\PROGRAM\CourseDesign\Fraction Calculator\Debug\Fraction Calculator.exe
请输入运算符: +
请输入一个分数: 1/2
运算过程: 1/2+1/2=1
请输入功能序号(0-7): 3
3 分数与分数的四则运算:
请输入一个分数: 1/2
请输入运算符: /
请输入一个分数: 0/1
错误!! 除数不能为0! 结果出错!
运算过程: 1/2/00=1
请输入功能序号(0-7):
```

除数为 0 错误测试

## 七、结论

### 1. 个人感受

通过此次的课程设计,我们将本学期由课本上所学习的有关内容应用到实践当中,刚开始感觉一头雾水,无从下手。但后来理清了程序设计的一条条思路后,画了设计层次图,然后一步一步深入,把程序完成了。其中也遇到很多麻烦,在网上搜索解决了问题。这一次课程设计也使我在应用过程中更加清楚的了解并掌握 C++ 语言的各部分的内容,进一步加深了对 C++ 语言的认识与理解,同时提高了个人程序设计思想,及时发现一些错误,并在调试过程中更加深刻的认识到 C++ 语言程序的严谨性。

总之,通过此次课程设计,我们学到了许多我们在书本上学不到的知识,受益匪浅。通过实践,我们能更好地掌握 C++ 语言,并运用它编写应用程序。我们对它的功能和实用性、严密性,和对程序的编写都有了比较深刻的体会。毫无疑问,编程过程中我们遇到了许多问题,但通过我们不懈的努力,团结奋斗的精神,我们终究将问题一一解除,最终实现了程序的各项功能。虽然我们的程序还不够完善,加以改进以实现更多的功能。但通过这次的课程设计我们不仅学到了许多知识,学会如何设计一个应用程序,包括程序设计方法,调试程序,还学会如何解决问题,学会将所学知识运用到实际情况中。

## 2. 存在问题及改进方法

在分数加法的时候，例如“ $1/2+1/2$ ”的情况返回的结果出现了“11”的情况，经过检查调试发现问题是出在判断重载输出流中：

```
if (fm == 1)           //当分母等于 1 时，输出分子
cout << fz;
if (fz == fm)          //当分子分母相等时，输出 1
cout << "1";
```

此处的 if 判断条件不够完善从而导致输出了两次 1

应改进为：

```
if (fm == 1)           //当分母等于 1 时，输出分子
cout << fz;
if (fz == fm && fm != 1) //当分子分母相等时，输出 1
cout << "1";
```

## 3. 可以改进的地方

可以考虑使用 QT Creator 进行图形化界面的设计并完善程序。

## 八、参考文献

- 1 沈显君《C++语言程序设计教程》北京：清华大学出版社，2015. 5