**《程序设计基础》Project报告**

**（CST11103）**

|  |  |  |  |
| --- | --- | --- | --- |
| **姓 名** | **文红兵** | **学 号** | **20214590** |
| **年级、专业、班级** | **大一本科生院电气信息类19班** | | |
| **上课时间** | **2021-2022学年第1学期** | | |
| **Project成绩** | **代码成绩**  （70%） | **报告成绩**  （30%） | **综合成绩** |
|  |  |  |
| **评语** |  | | |
| **评价教师** |  | | |

**《程序设计基础》Project任务书**

|  |  |
| --- | --- |
| 名 称 | 分数计算器的设计与实现 |
| 类 型 | □验证性 □设计性 ■综合性 |
| 内 容 | 运用面向对象程序设计思想，基于命令行界面设计并实现一个分数计算器。 |
| 要 求 | （1）按照分数类Franction的声明，实现其所有成员函数、友元及运算符重载。类的声明如下：  class Fraction  {  friend Fraction operator+(const Fraction& frac1, const Fraction& frac2); //重载+运算符  friend Fraction operator-(const Fraction& frac1, const Fraction& frac2); //重载-运算符  friend Fraction operator\*(const Fraction& frac1, const Fraction& frac2); //重载\*运算符  friend Fraction operator/(const Fraction& frac1, const Fraction& frac2); //重载/运算符  friend bool operator==(Fraction frac1, Fraction frac2); //重载==运算符  friend bool operator>(const Fraction& frac1, const Fraction& frac2); //重载>运算符  friend bool operator<(const Fraction& frac1, const Fraction& frac2); //重载<运算符  friend ostream& operator<<(ostream& out, const Fraction& frac); //重载<<运算符  friend istream& operator>>(istream& in, Fraction& frac); //重载>>运算符  friend void sortFraction(参数列表自行设计);//对分数数组排序  public:  Fraction(); //无参造函数  Fraction(int n, int d); //带参造函数  Fraction(const Fraction& f); //复制造函数  void setFraction(int n, int d); //设置分数的分子和分母  int getNumer(); //获取分数的分子  int getDeno(); //获取分数的分母  void RdcFrc(); //当前分数约分  private:  int numer; //分子  int deno; //分母  };   1. 分数计算器的主要功能为：   ① 程序开始时选择想要使用的功能，如图1所示  请选择功能（键入1或者2）：   1. 分数计算 2. 分数排序   —  图1 分数计算器开始界面  ② 如果选择“1.分数计算”则进入计算界面，如图2所示  请输入分数计算式（如1/2+1/3回车），输入#号键返回上一层目录：  —  图2分数计算初始界面  如果输入正确分数计算式给出计算结果，并等待下一次用户输入，如图3所示  请输入分数计算式（如1/2+1/3回车），输入#号键返回上一层目录：  1/4+1/4  =1/2  请输入分数计算式（如1/2+1/3回车），输入#号键返回上一层目录：  —  图3分数计算界面  如果输入错误分数计算式，提示输入错误并等待用户重新输入，如图4所示  请输入分数计算式（如1/2+1/3回车），输入#号键返回上一层目录：  1/4？1/a  分数算式输入错误！  请输入分数计算式（如1/2+1/3回车），输入#号键返回上一层目录：  —  图4分数计算输入错误界面  ③ 如果选择“2.分数排序”则进入分数排序界面，如图5所示  请输入一组分数，用逗号隔开，如需由小到大排序用符号<结尾，由大到小排序用符号>结尾（如1/2,1/4,3/5<回车），输入#号键返回上一层目录：  —  图5 分数排序初始界面  注意：这里分数个数无上限限制，应考虑动态地创建分数的数组序列。  当输入一组正确分数后，输出排序结果，并等待下一次用户输入，如图6所示：  请输入一组分数，用逗号隔开，如需由小到大排序用符号<结尾，由大到小排序用符号>结尾（如1/2,1/4,3/5<回车），输入#号键返回上一层目录：  1/2,1/5,3/7<  1/5 3/7 1/2  请输入一组分数，用逗号隔开，如需由小到大排序用符号<结尾，由大到小排序用符号>结尾（如1/2,1/4,3/5<回车），输入#号键返回上一层目录：  —  图6 分数排序界面  当输入的分数不正确，则提示输入错误，并等待用户重新输入，如图7所示：  请输入一组分数，用逗号隔开，如需由小到大排序用符号<结尾，由大到小排序用符号>结尾（如1/2,1/4,3/5<回车），输入#号键返回上一层目录：  1/2,1/a3/7  输入错误！  请输入一组分数，用逗号隔开，如需由小到大排序用符号<结尾，由大到小排序用符号>结尾（如1/2,1/4,3/5<回车），输入#号键返回上一层目录：  —  图7 分数排序输入错误界面 |
| 任务时间 | 2021年12月6日至2021年12月31日 |

**代码评分标准（满分10分）**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **学习态度** | **程序功能** | **程序Bug** | **程序界面** | **异常处理** | **得分** |
| 抄袭、被抄袭 | -- | -- | -- | -- | **0** |
| 自己完成编码  无创新意识 | 没有完成实验要求的所有功能 | -- | -- | -- | **5** |
| 自己完成编码  有一定的创新意识 | 基本完成实验要求的所有功能 | 很多 | 一般 | 无 | **6** |
| 自己完成编码  有一定的创新意识 | 完成实验要求的所有功能 | 较多 | 一般 | 无 | **7** |
| 自己完成编码  有较强的创新意识 | 完成实验要求的所有功能，并有所扩展 | 较少 | 良好 | 有 | **8** |
| 自己完成编码  有较强的创新意识 | 完成实验要求的所有功能，并有所扩展 | 极少 | 友好 | 有 | **10** |

**报告评分标准（满分10分）**

|  |  |  |  |
| --- | --- | --- | --- |
| **文档规范性** | **文档条理性** | **表述准确性** | **得分** |
| 不规范，需求分析、系统设计、系统实现部分不完整。 | 描述缺少逻辑性，没有条理。 | 文字描述不准确。 | **3** |
| 较规范，有需求分析、系统设计、系统实现部分，但实质性描述较少。 | 描述具有一定的逻辑性，条理较清晰。 | 文字描述较准确，但缺少必要的类图、程序流程图等图表说明。 | **5** |
| 规范，有需求分析、系统设计、系统实现部分，描述较充分。 | 描述具有较好的逻辑性，条理清晰 | 文字描述较准确，所描述的类图、程序流程图基本准确、无大错。 | **7** |
| 规范，有需求分析、系统设计、系统实现部分，描述充分。 | 描述具有很好的逻辑性，条理清晰 | 文字描述准确，所描述的类图、程序流程图准确。 | **10** |

1. **需要解决的问题**

**1、实现数学的基本运算**

设计一个分数计算器来实现计算机对分数的基本运算，包括加法运算，减法运算，乘法运算，除法运算。

1. **实现数学方面的大小排序**

设计分数计算器实现对传入的数据的大小排序，包括由大到小和由小到大的排序。

1. **软件的需求**

**1、数据处理**

需要设计一个分数的计算器，实现用户输入分数和对分数的处理，对用户输入的两个分数进行加减乘除，以及对用户传入的不确定个数的分数进行大小的排序。

**2、数据存储**

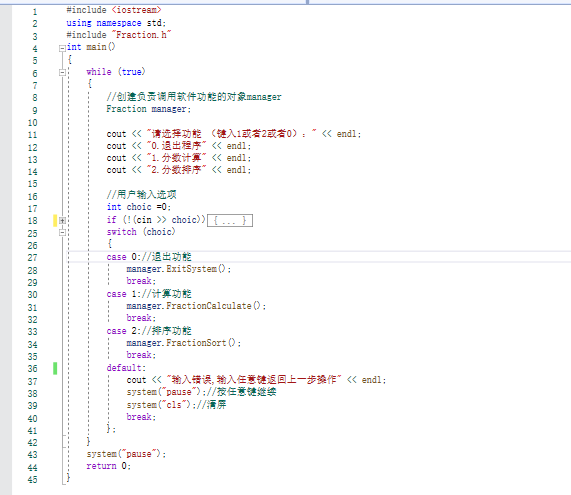
由于该软件仅仅是一个功能软件，负责进行计算和排序，因此该软件不需要对输入的数据和产生的数据结果进行存储。

**3、功能方面**

该软件主要需要两个功能，第一个功能就是计算，实现在数学范围定义的对数字的基本运算，并且在用户的屏幕上输出计算的结果，第二个功能就是排序，实现对用户输入的分数进行由大到小或是由小到大的排序，并在屏幕上输出排序后的结果。额外需要一个退出功能，也就是用户用完该程序后退出该程序。

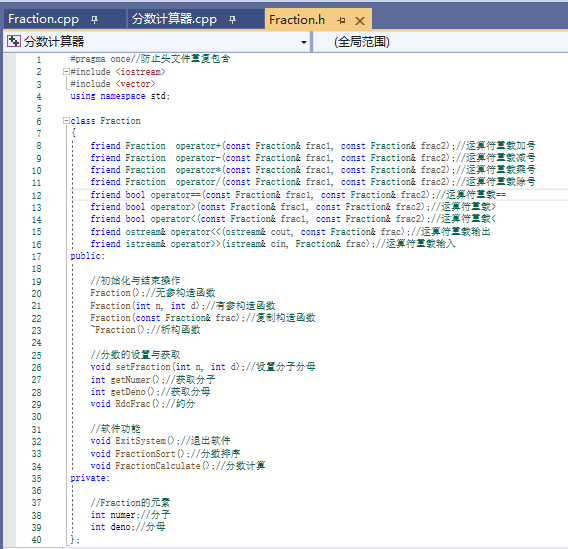
**三、软件模块的设计思路**

第一步是创建一个软件的主体也就是“分数计算器”，来实现该程序主体的部分，统筹盖软件的功能，实现不同功能之间的操作与联系。



该流程图简单易懂，在循环下的用户输入数字调用三个不同功能。三个功能分别用不同的成员函数来实现。

第二步是创建一个“分数”的类的声明，在这个类的里面写好需要的实现功能以及不同的类的成员，成员函数，对不同的运算符进行重载。下面图片的注释写明了该类图需要实现的功能，在原有的要求之下，我新增加了软件的退出功能和分数计算功能，方便“分数计算器.cpp”的统筹调用。以及实现类的封装性。



第三步实现分数类Fraction里面的声明，这个也是最重要的，写好各个功能实现的代码，做好注释，如果有需要，方便后期的维护和增删。

`}4WB5_XV$$}%U3SXYU_%FN

**四、这样设计的目的和好处**

**1、这样设计目的**

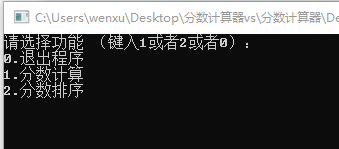
最主要的目的是让设计有条理，代码清晰易读懂，实现软件的不同功能，也同时能够方便后期的修改。

1. **这样设计的好处**

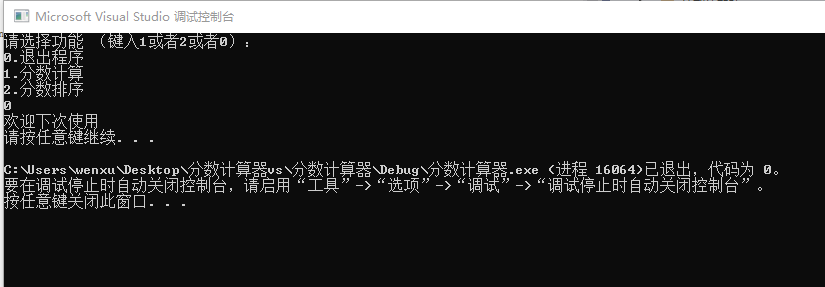
当然对我而言最大的好处就是很容易看懂代码，能够随时进行编程，很轻易就能发现上次写到哪里了，写了什么功能，接下来要写什么功能，非常有条理性。在编写代码的时候及时发现错误，哪个地方出问题了就修改哪一个地方，而不是牵一发而动全身。对于别人阅读我的代码而言，因为我的注释写的很详细，，也能够很轻易的看懂每个部分的功能和实现的方法。

1. **软件的实现效果以及如何使用该软件**
2. **软件的实现效果**

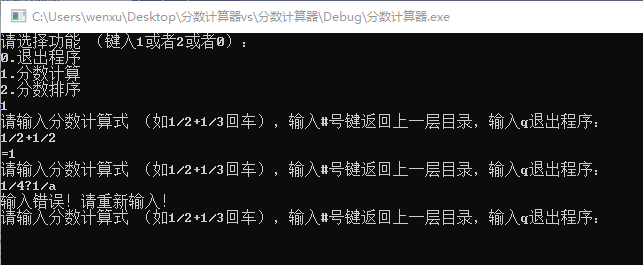
该软件的主界面



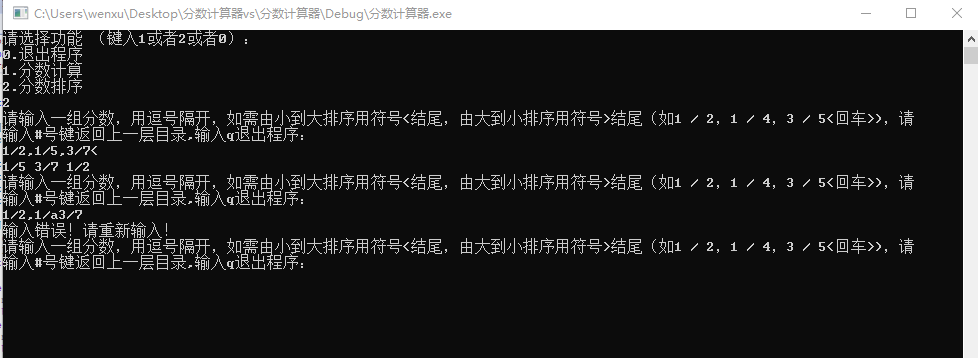
软件退出功能界面



软件计算功能界面

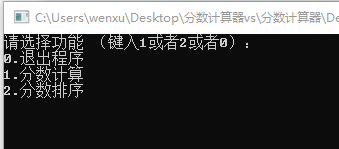


软件排序功能方面界面

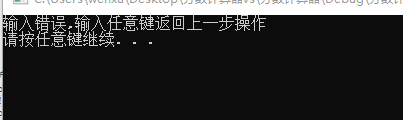
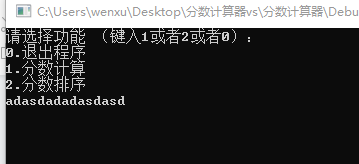


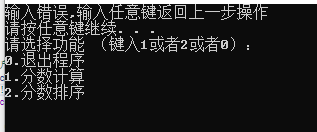
1. **使用方法**

在主界面用户可以选择三个功能，即“退出程序”“分数计算”“分数排序”，分别从键盘读入0，1，2即可



如果输入错误会清空屏幕同时提醒“输入错误,输入任意键返回上一步”。





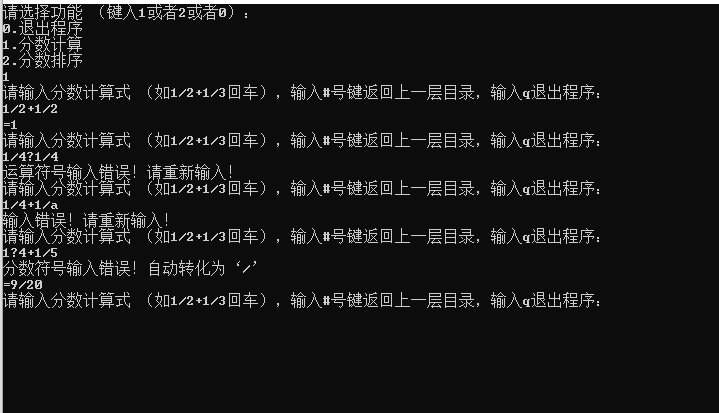
**如果用户输入0，也就是退出该程序  
如果用户输入1，就选择了分数计算的功能**

进入分数计算功能后会有相应的提示话语帮助用户输入

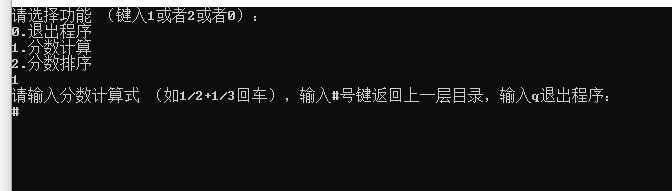
同时分数计算功能里面也有三个子功能

第一个是计算功能

如下图，当然用户输入错误会有相应的不同提醒

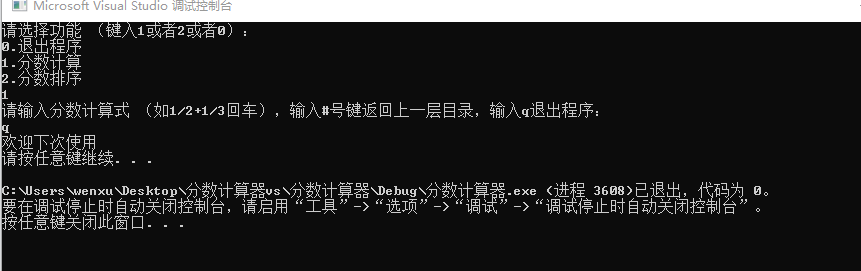


第二个功能是输入#号键返回上一目录并清空屏幕，如下图



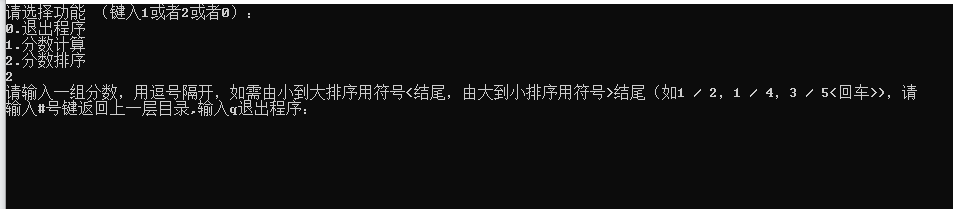


第三个功能就是输入q号键退出该程序功能，如下图



**如果用户输入2，也就是选择了分数排序的功能，该功能下有三个子功能**

第一个功能排序功能，如下图



第二个功能和第三个功能和上面分数计算里面的相同。

**六、总结**

这个项目算是花了较多心思和时间来完成，其中绝大部分都是自己根据以前的完成作业积累下来的经验和课外拓展的知识独立完成，有两处在完成这个项目时上网查询了一下，会在下面参考文献中写出。这个也算是我学习C++的处女作品了，总体而言还是非常满意自己的，在这学期的学习当中也发现了一些自己学习的不足，在开始时由于没有相关的学习经验，完完全全就是一个小白，导致在上课时完全听不懂讲课，不过还好及时调整，积极上网寻找资源，寻找网课跟上老师的节奏，已经养成了先预习在听课的习惯，也不再像刚开始那般无奈了。有时候写一个小程序想了一天也想不出来，也不知道有相关的函数简化步骤，导致学习很累，别人的代码也看不懂，不过好在能够坚持下来，现在也算是对编程语言有了基本的认识了。

**七、参考文献**

1、https://zhidao.baidu.com/question/2143493839235084628.html

C++怎么编写出提示 能显示你输入的变量不符合要求

2、https://blog.csdn.net/lesslie1992/article/details/14057417

C++编程对缓冲区的理解