

C++程序设计课程设计

**2018/2019(2)**



实验题目 合同管理系统

学生姓名 王程飞

学生学号 20186061219

学生班级 软件工程05

任课教师 毛国红

提交日期 2019.5.23

**计算机科学与技术学院**

**图书管理系统 实验报告**

1. **大型实验的内容**

合同管理系统（CMS：Contract Management System）用于合同管理，包括生成合同，修改合同，删除合同，按要求查询合同，文件读取等功能。使用学习过的C/C++程序设计的知识完成合同管理系统的设计与实现。

1. **运行环境**

合同管理系统（CMS）在Visual Studio 平台下开发，操作系统：Windows 10。

硬件环境：

处理器：Intel(R) Core (TM) i7-8750H CPU @ 2.20GHz

内存：8.00GB

系统类型：64位操作系统

1. **实验课题分析**

**3.1 图书管理系统的主要功能**

**合同管理系统（CMS）主要功能为：**合同添加，修改，删除，保存读取，查找等操作。详细的系统功能结构为图1所示。

合同管理系统（CMS）

增删模块

查询模块

修改模块

文件读写模块

图1 系统结构图

**系统各模块的功能具体描述为：**

1. **文件读写模块**

打开文件读取合同信息，并将改动保存到文件

1. **增删模块**

新增合同和删除合同

1. **查询模块**

按日期查询合同，显示所有合同，查询某一天的合同，到期合同功能

1. **修改模块**

修改指定合同的ID, 甲方, 乙方, 时间, 金额, 内容等.

**3.2 系统分析及设计**

**系统涉及对象有两个基本类：**时间类和合同类。其中合同类涉及的功能操作归纳为如下表1所示：

表1 合同类的操作

|  |  |
| --- | --- |
| **对象** | **涉及的对象操作** |
| 日期  Date | 输出时间 |
| 输入时间 |
| 验证输入合法性 |
| 比较大小 |
| 合同类  Contract  ContractUI | 合同开始时间 合同结束时间 合同签订 |
| 合同ID |
| 甲方 乙方 |
| 合同金额 |
| 合同内容 |
| 输入合同，输出合同 |

采用面向对象的方式实现合同管理系统。系统的主要的类结构如图2所示。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Date | 年 月 日 | | | | | | | |
| Contract | ID | 甲方 | 乙方 | 金额 | 内容 | 签订时间 | 开始时间 | 结束时间 |
| ContractUI | 继承自Contract类，主要用于人性化显示输入输出交互逻辑 | | | | | | | |

图2 系统主要类结构图

用文本文件进行数据的保存，需要保存的数据主要包括上述Contract类成员，在main.cpp中以函数的方式实现所有的文本操作相关的功能。

**3.3系统的实现**

**（1）类的编写**

系统工程名为：CMS。包含了

Date类（日期操作类）

Contract类（合同数据类）

ContractUI类（合同交互类）

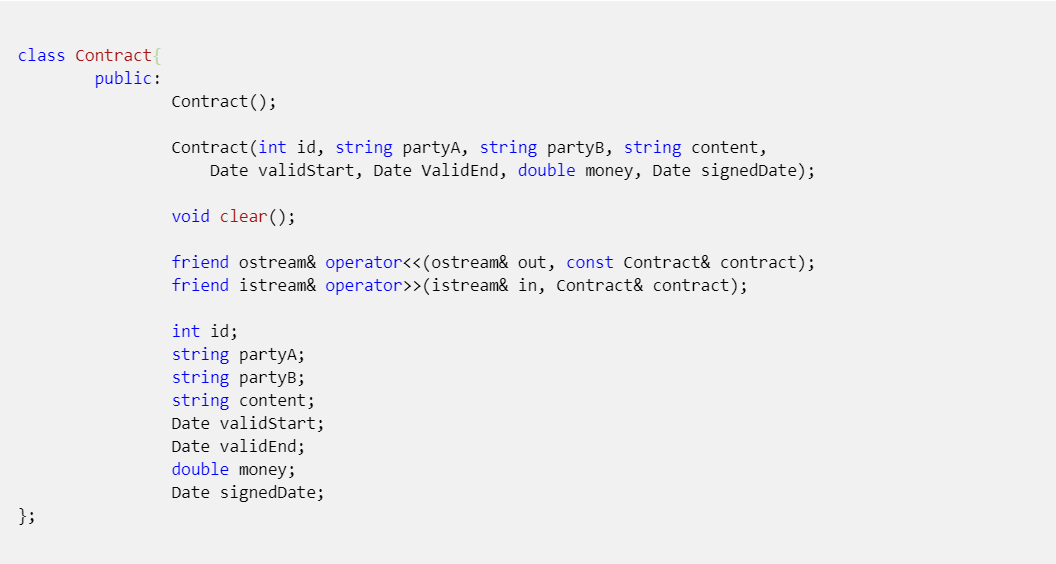
三个基本类，其中ContractUI继承自Contract类，增加了人性化提升用户输入输出过程的功能。

具体类结构声明如下：

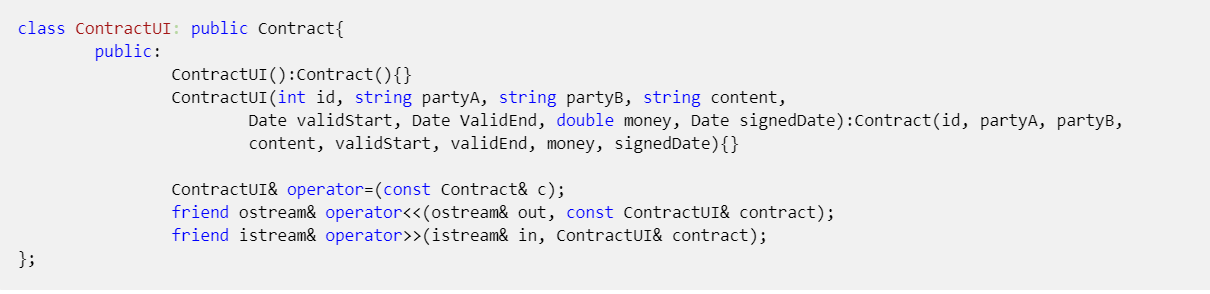
* **Date类**



* **Contract类**

****

* **ContractUI类**

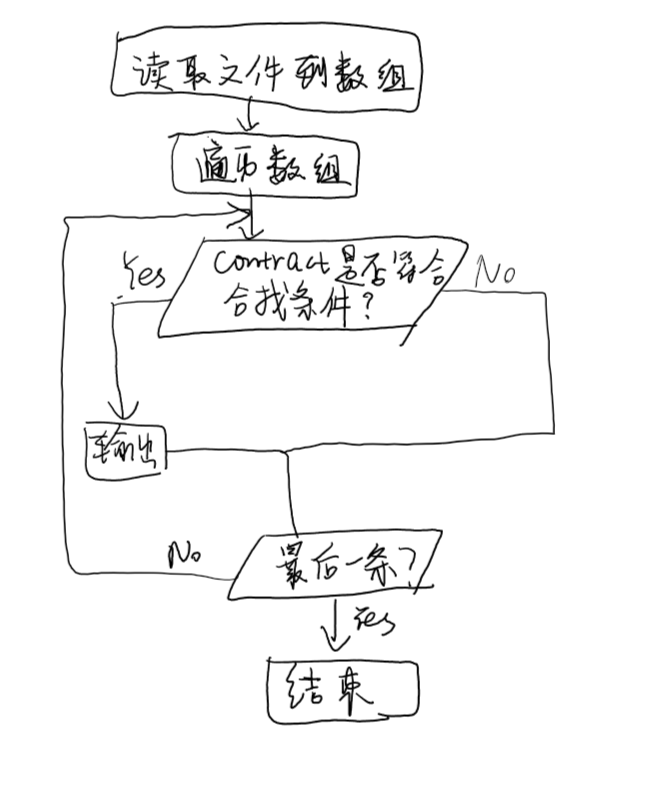
****

**（2）数组的使用**

系统实现采用文件的输入输出流对文本数据进行读取与写入，但是由于合同信息是一个数据的集合，于是对数据的存储使用了一维数组。

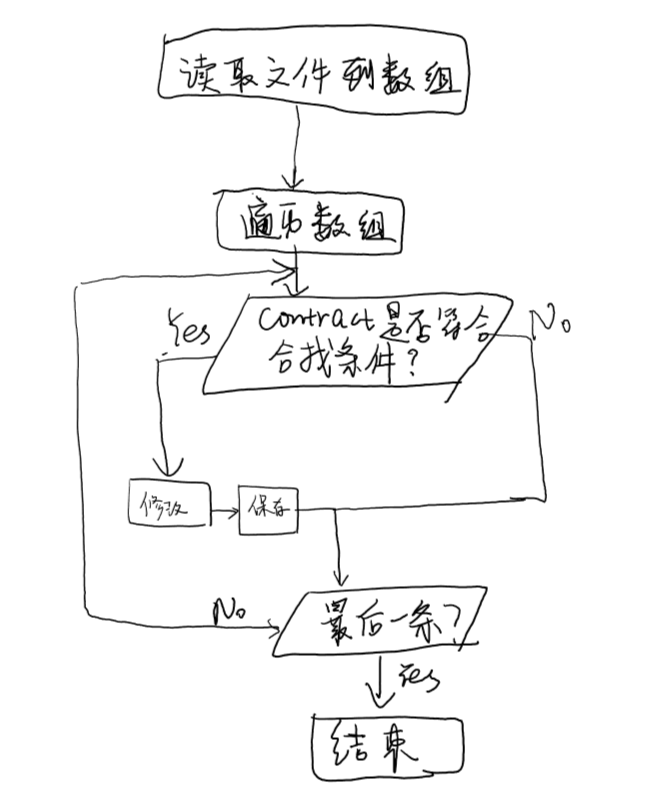
* **合同的查找：**

对数组进行遍历，当发现某项与条件符合后输出



* **合同的修改：**

对数组进行遍历，以ID为KEY进行查找，当某项ID与KEY\_ID符合后对Contract对象修改，最后保存到文件

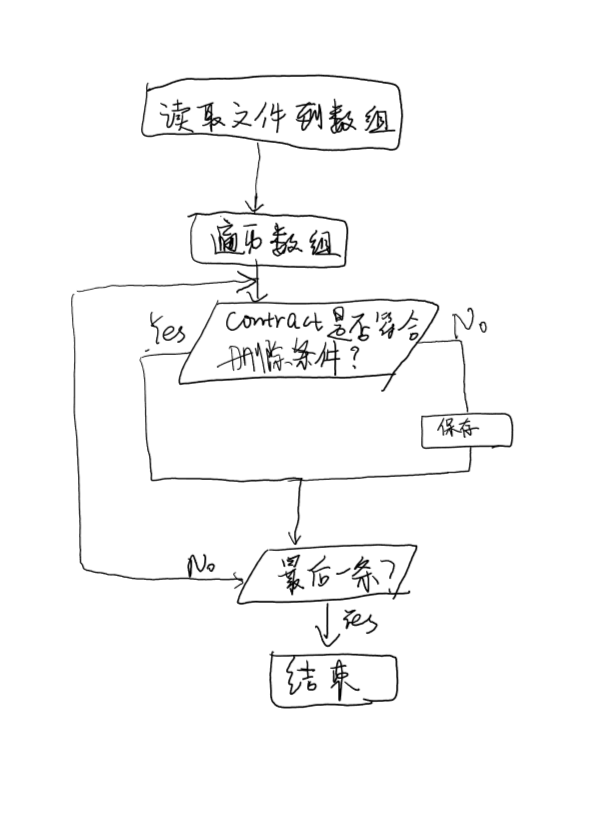
****

* **合同的添加：**

先完成对新合同的创建，然后追加文件结尾

* **合同的删除：**

对数组进行遍历，以ID为KEY进行查找，当某项ID与KEY\_ID符合后删除该项并保存到文件

****

**（3）交互界面以及功能菜单的实现**

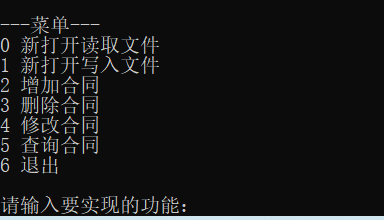
系统运行开始的界面如图5所示  
 

图5 开始登录界面

主要通过选择结构和循环结构实现界面的前进和后退。例如，主菜单的选择项为：

1. 新打开读取文件
2. 新打开写入文件
3. 增加合同
4. 删除合同
5. 修改合同
6. 查询合同
7. 退出

其中，菜单5查找合同的子项有：

0．输出全部合同

1．输出某一天的

2．输出到今天为止到期的

3．输出这个月签订的合同

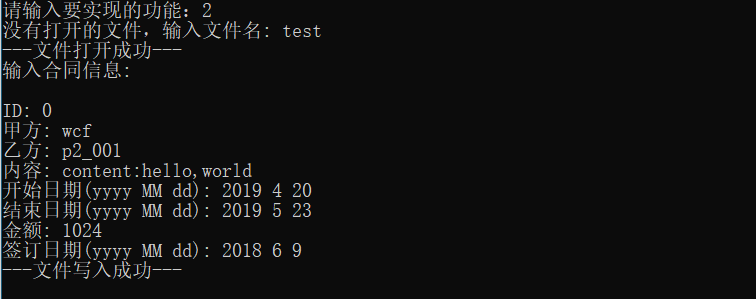
4．根据甲方或乙方查询合同

1. **实验调试、测试、运行记录及分析**

系统在调试测试过程中遇到若干问题，不过经过仔细反复的检查已经消除各种bug

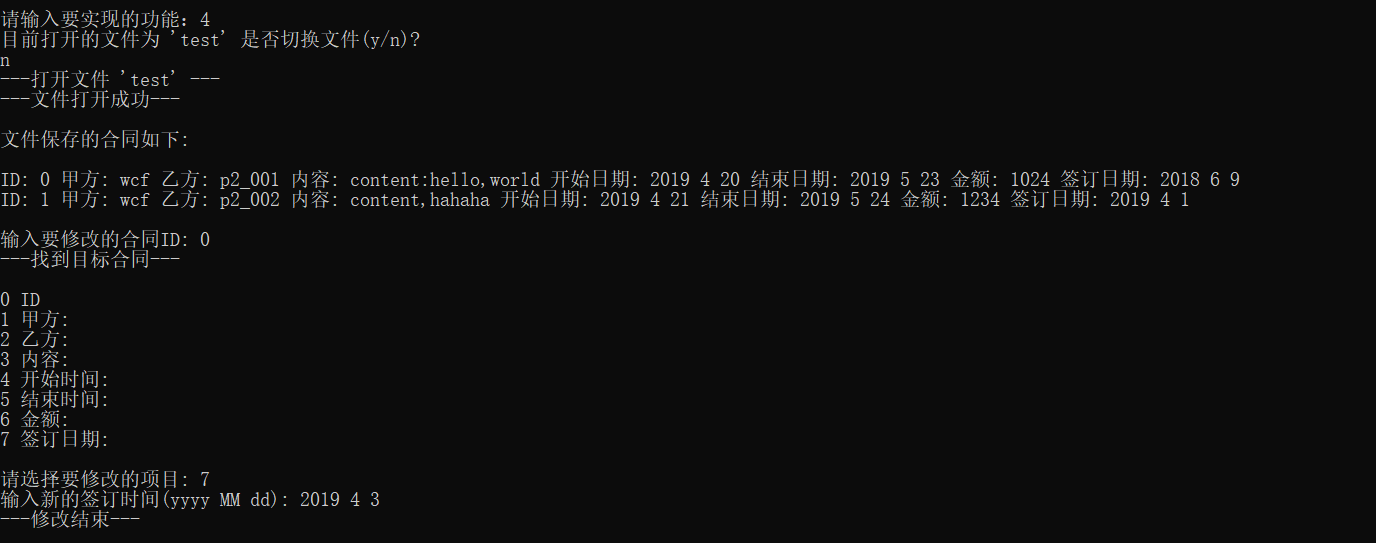
主要的测试经过如下:

1. **新增合同**

****

选择菜单项 ‘2’，程序提示是否打开现在的文件或打开新的文件，然后按提示输入各项数据，最后保存到文件

1. **修改合同**

****

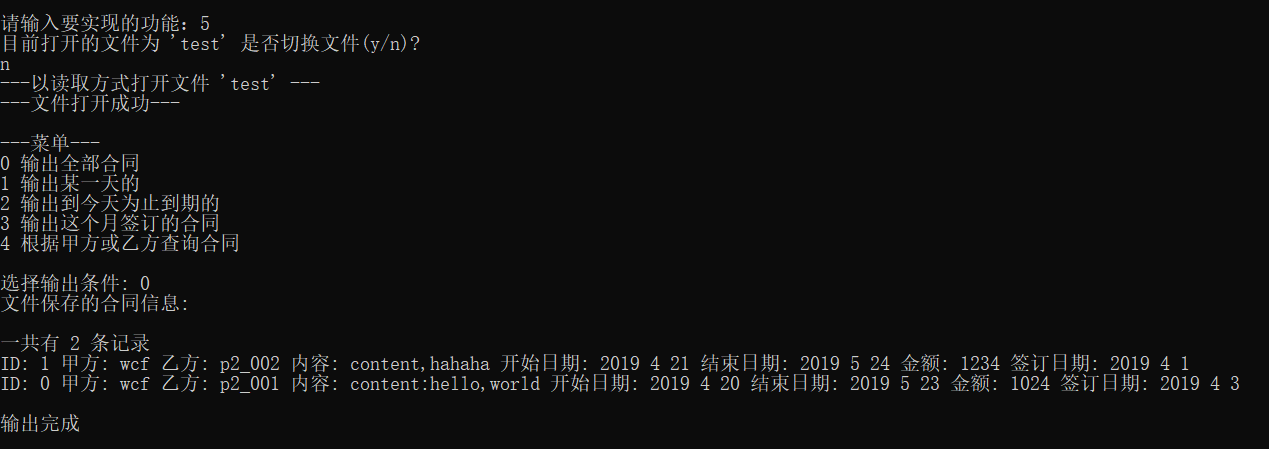
选择菜单项 ‘4’, 程序将显示文件中保存的所有合同，然后要求用户输入要修改的合同ID，随后输入要修改的项的编号，最后按提示输入数据并保存

1. **查找合同**

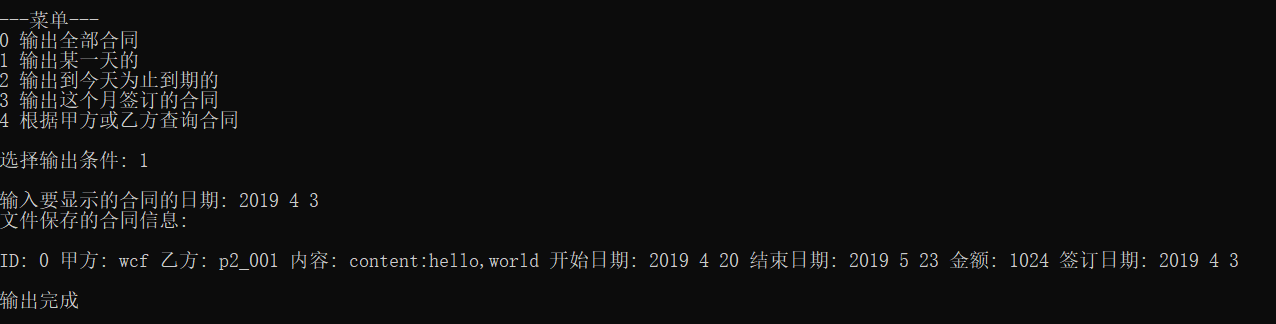
选择菜单项 ‘5’, 程序进入查找子菜单

菜单项有：

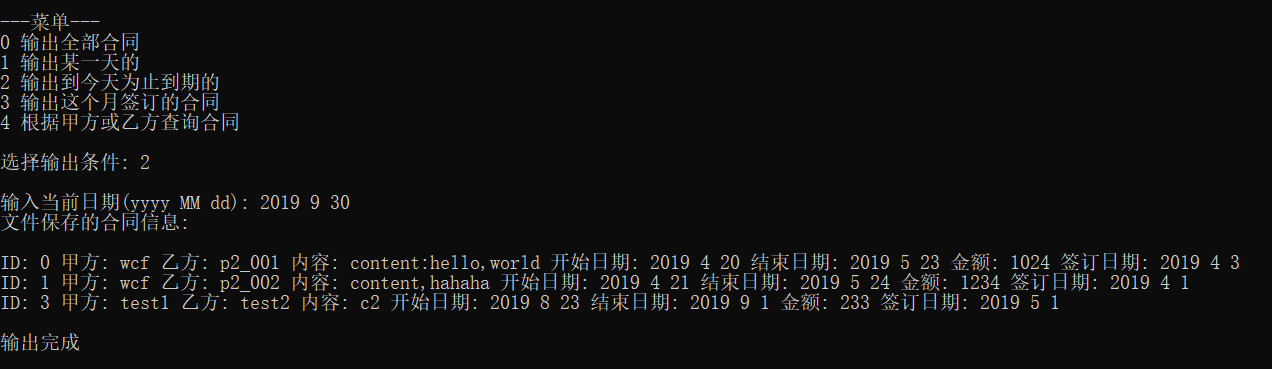
1. 显示所有合同



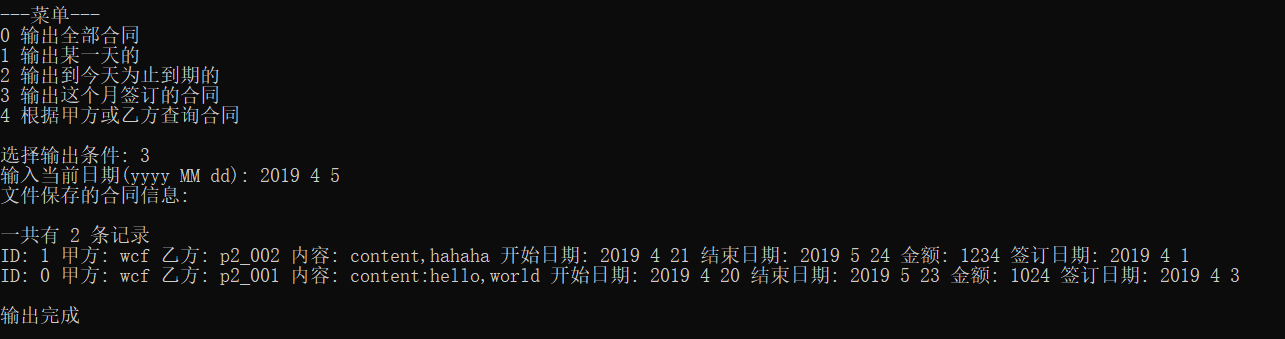
1. 查找某一天的合同



1. 显示今天为止到期的合同



1. 显示这个月签订的合同



1. 根据甲方或乙方查询合同



选择1时将按时间排序输出所有合同

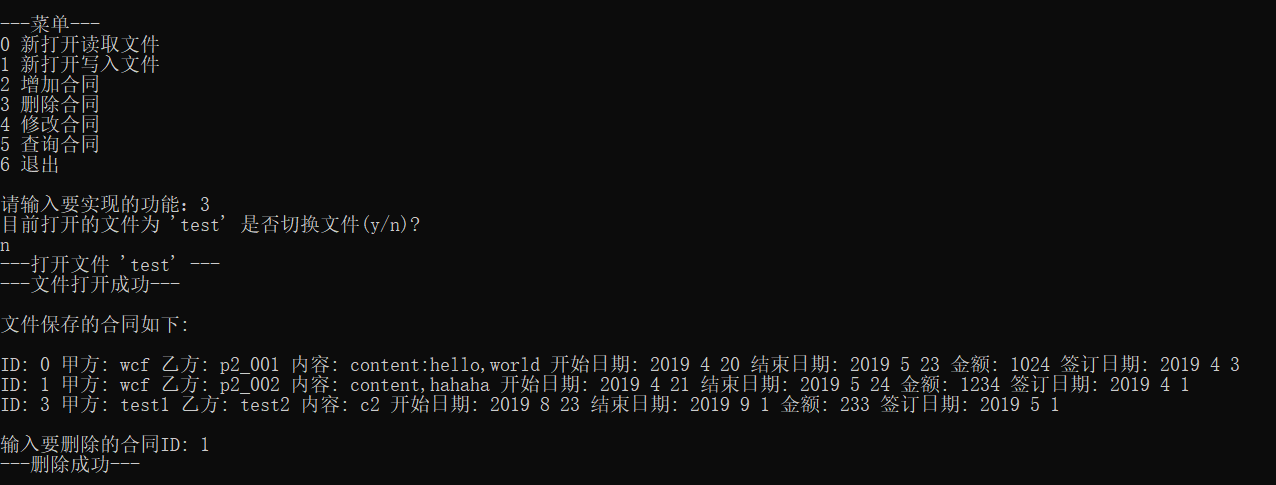
选择2时将要求数日指定日期，并输出当天的合同

选择3时将要求输入今天的日期，输出到今天为止到期的合同

选择4时输入日期，输出日期所对应月份的合同

选择5时输入要查找的甲方或乙方，输出符合条件的合同

1. **删除合同**

****

选择菜单项 ‘5’, 程序将显示文件中保存的所有合同，然后要求用户输入要删除的合同ID，删除完成后保存到文件

**遇到的问题及解决方法如下：**

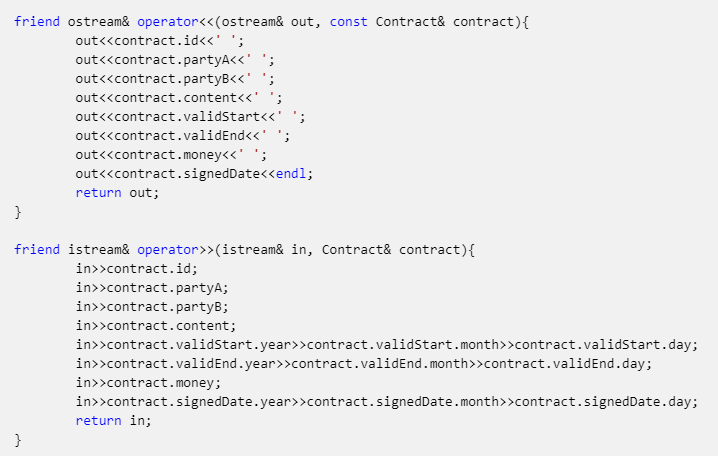
* + **问题1：**

发现对合同数组按日期排序并不能简单的用sort直接排序,所以在主函数中自定义比较器bool compare(const ContractUI& a, const ContractUI& b)按照自定义逻辑比较日期

* + **问题2：**

在合同类输入输出时，为求代码复用和可读性，将流操作符<<, >>重载方便输入输出，但文件读写和用户交互输入输出有所不同，文件读写不需要额外提示数据类型和格式，而在用户交互时时必须的，故在Contract类的基础上派生出ContractUI类，再次重载运算符<<, >>可以达到功能要求，再加入额外的构造方法和拷贝方法使得两者可以相互转换。

**Contract类**：



**ContractUI类：**

****

1. **实验总结**

合同管理系统基本满足任务书的功能要求，类的结构和关系清晰，功能完善，支持自定义文件名进行保存和读取，支持多种条件的数据查找，删除、修改的功能实现的也不错。

存在的缺点主要是在数据存储使用简单的数组，而没有使用链表，用户交互没有很人性化，代码冗余比较多，没有专门创建一个文件IO对象读写，逻辑比较分散，相信在今后的实践中会有所提高。

1. **附录：源代码**