A1分数统计

设计一个分数统计程序。包括学生信息的输入输出以及排序。通过该课题全面熟悉数组、字符串、文件的使用，掌握程序设计的基本方法及友好界面的设计。

流程图：

开始

读入学生人数

读入学生成绩（格式： 姓名 成绩）

按成绩排序

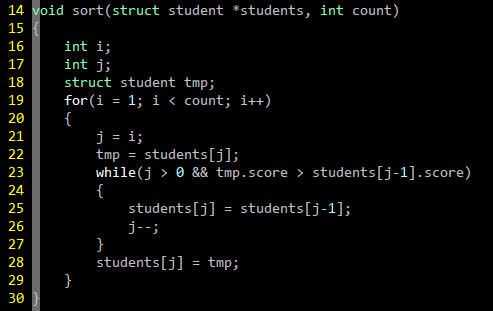
输出排序后结果

结束

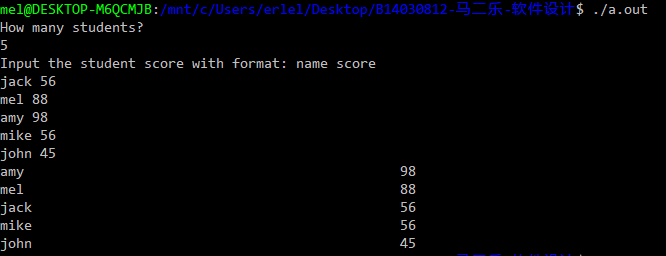
核心代码：

按成绩从大到小排序(亦可采用库函数<stdlib.h>中的qsort函数)

此处采用了插入排序



运行图：



A2 打字程序

设计一个打字程序。包括随机产生字符串，以及字符串比较和统计。通过此课题，熟练掌握数组、格式输出、字符串处理等。

流程图：

开始

生成随机字符串str

输入字符串in

比较str和in

Y

再来一次

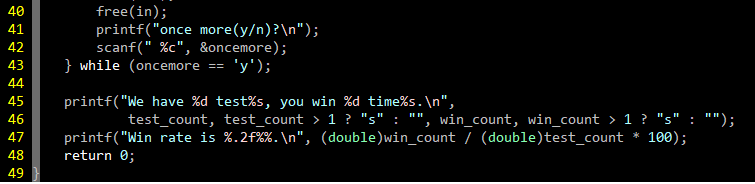
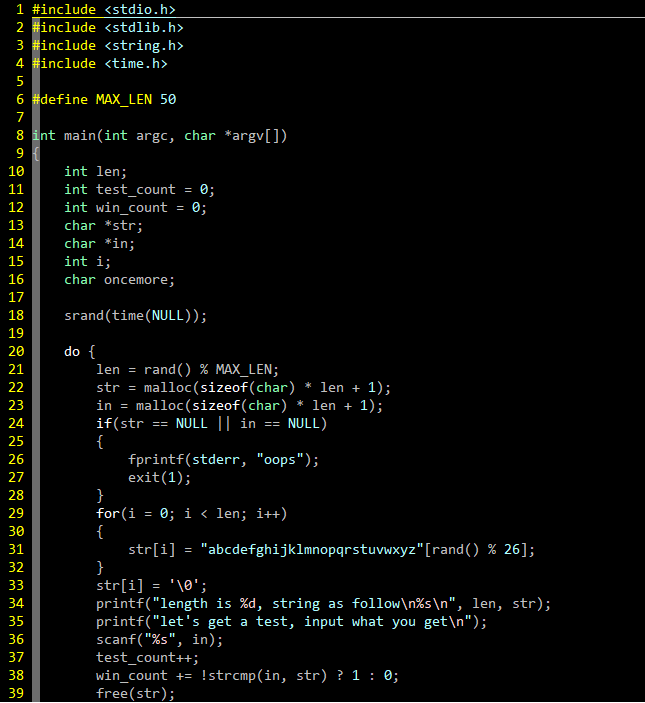
N

输出正确率

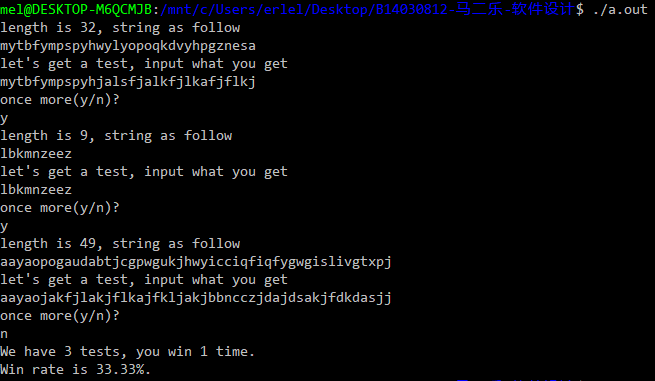
结束

核心代码:

每次随机出的字符串内容和长度都是随机的，最大长度使用了宏定义，正确率以条数计算，结束时输出输赢次数及正确率。



运行图：



A3 文本编辑器

设计一个简单的文本编辑器，该系统要求对一个文本文件中的内容进行各种常规操作，如：插入、删除、查找、替换等功能。通过此课题，熟练掌握文本文件的操作及用字符数组或字符指针实现字符串操作的功能。

流程图：

开始

读入文件名称

显示功能模块

显示文件内容

插入

删除

查找

替换

保存

退出

Y

是否再次操作

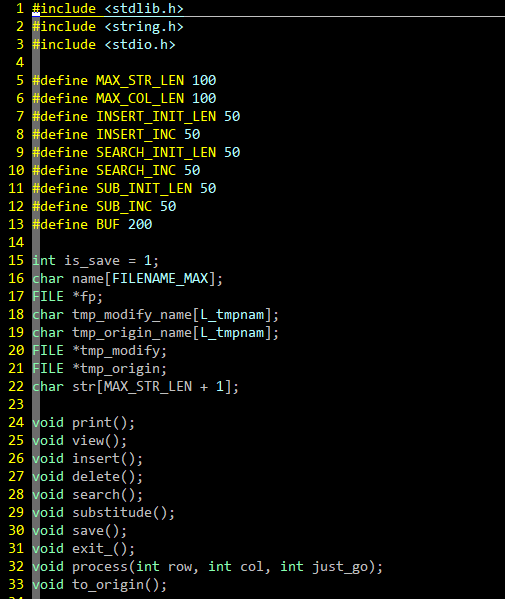
N

结束

核心代码：

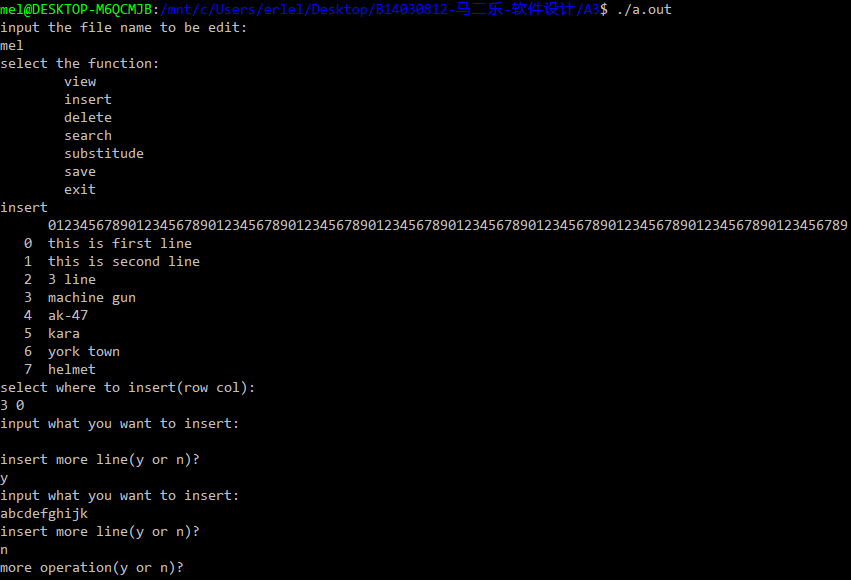
核心思想是创建两个流，一个存放最新的操作结果，一个用作各项操作的临时流。

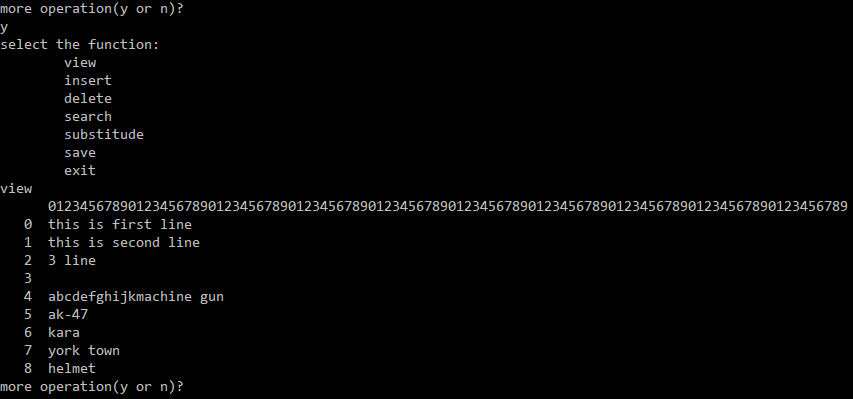
如下为全局变量和各宏及函数声明，每个功能单独一个函数进行执行。

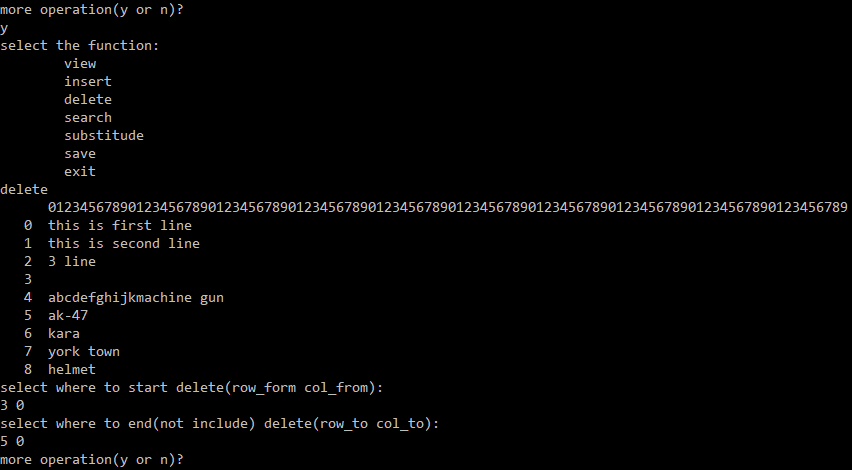
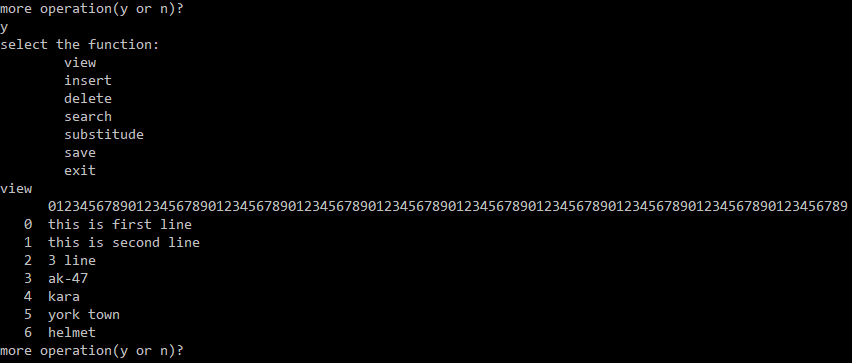
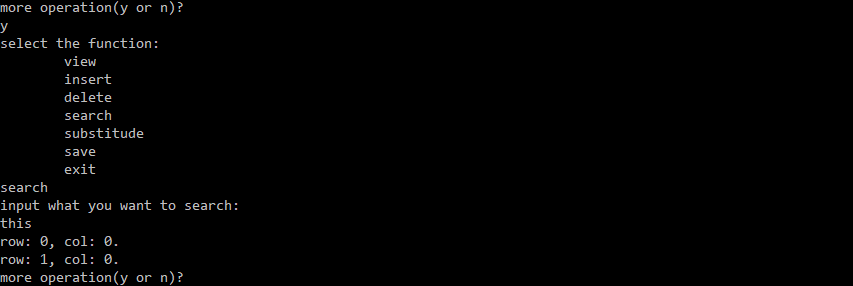
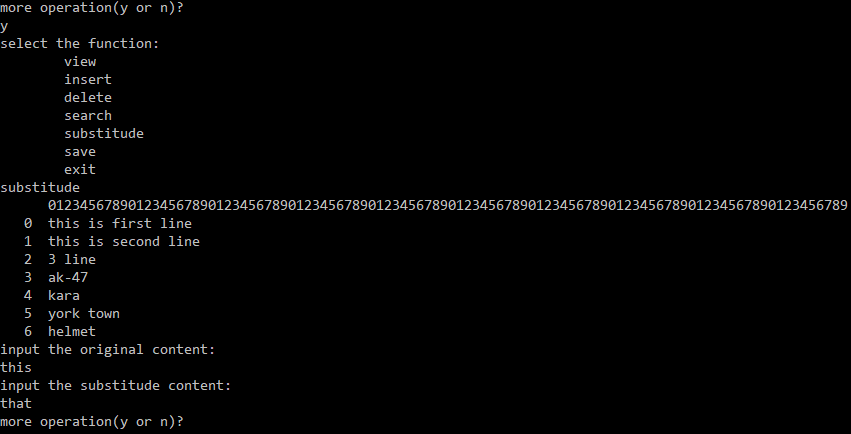
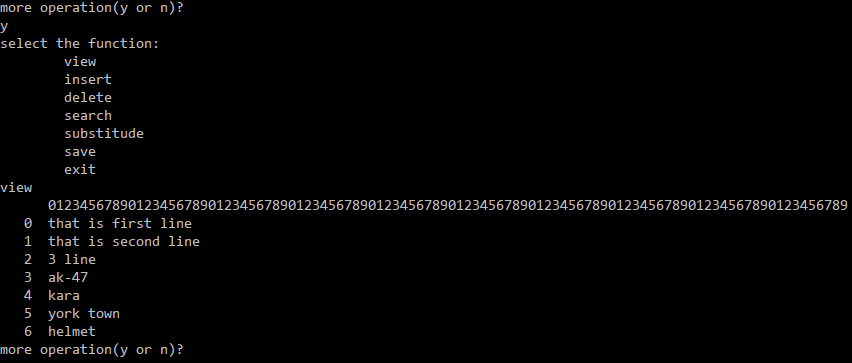
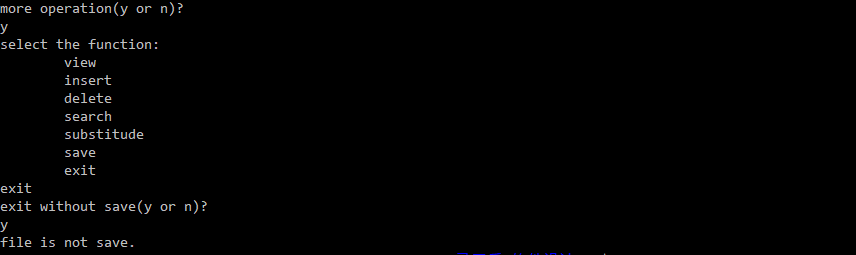


各个功能的函数实现请参考附带文件中的代码，只要记住tmp\_origin和tmp\_modify这两个流，tmp\_origin用于在每次操作之后存放结果，tmp\_modify用于进行每次操作，基本就能体会代码。每个功能的实现都异曲同工。

运行图：





A4 加密

设计一个加密程序。包括明文与密钥的转换。通过此课题，熟练掌握数组、格式输出、字符串处理、类型转换等。

流程图：

开始

显示功能模块

加密

解密

退出

Y

是否再次操作

N

结束

核心代码：

使用加密方式如下，从输入得到key（1-26），key为26个字母的偏移量，当key为5，示例如下

abcdefghijklmnopqrstuvwxyz

^

fghijklmnopqrstuvwxyzabcde

^

即将26个字母表对应为偏移key后的字母表。

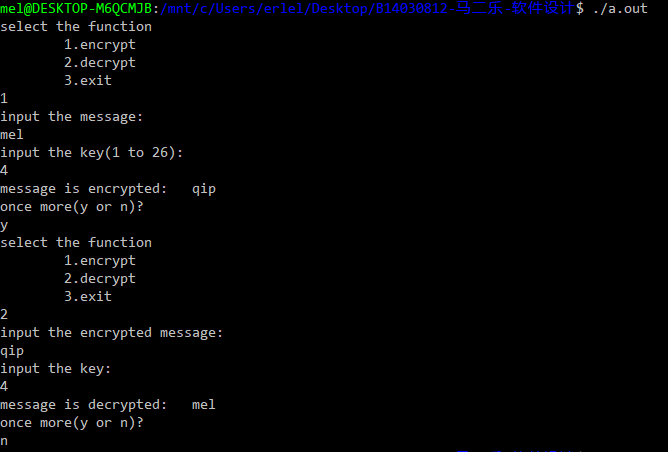
加密代码如下：



解密代码如下：



运行图：



A5 进制转换器

设计一个进制转换器程序。包括二进制、八进制、十进制、十六进制数互相转换。通过此课题，熟练掌握字符串、格式输出、进制换算的各种操作。

流程图：

开始

读取进制

输入相应进制数字

显示该数字2/8/10/16

进制下表示方式

Y

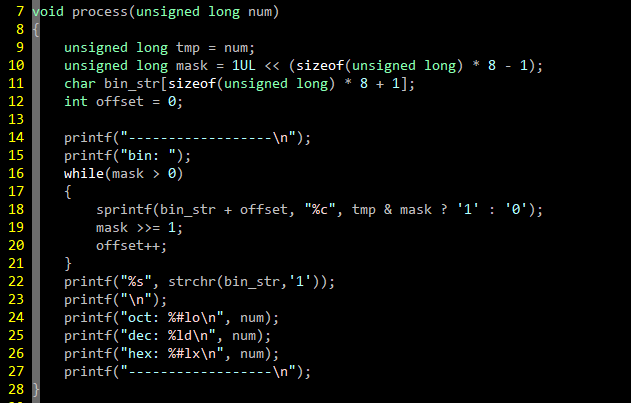
是否再次操作

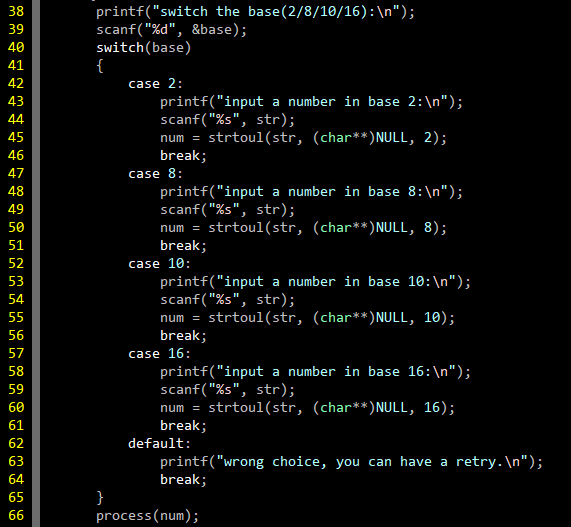
N

结束

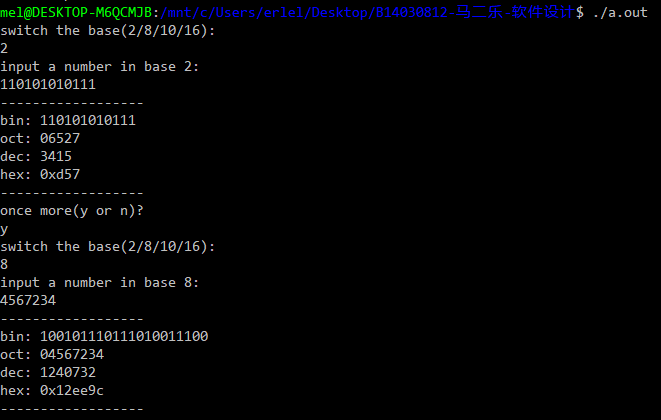
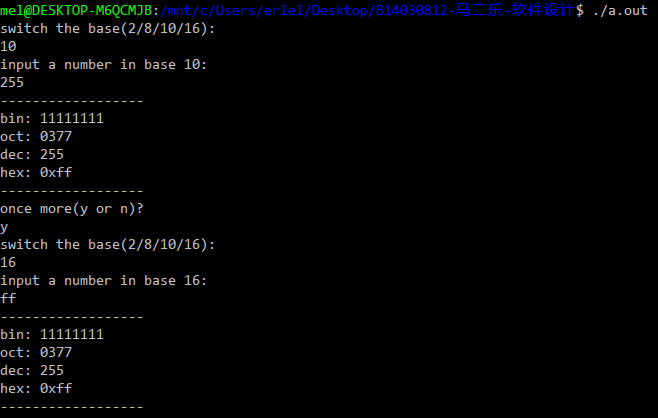
核心代码：

读入时统一用标准库函数strtoul将各进制数转换为unsigned long类型的整数，之后用process函数统一处理。





运行图：

B6 学生成绩核算系统的设计与实现

课题要求：

（1）按班级按课程从文件中读入相应的平时成绩、期中考试成绩和期末考试成绩。

（2）三个成绩对总评成绩的百分比被定义为常数，各占总成绩的30%、30%和40%。

（3）计算每位学生的总评成绩。

（4）计算该班级本课程的总平均成绩。

（5）计算处于优、良、中、及格、不及格的学生人数以及占总人数的百分比。其中100-90为优，89-80为良，79-70为中，69-60为及格，60分以下为不及格。

（6）按要求输出成绩在优、良、中、及格、不及格各区间的学生学号、成绩。

流程图：

开始

显示功能模块

显示学生成绩条目

统计班级平均成绩

成绩等级统计

特定等级成绩统计

是否再次操作

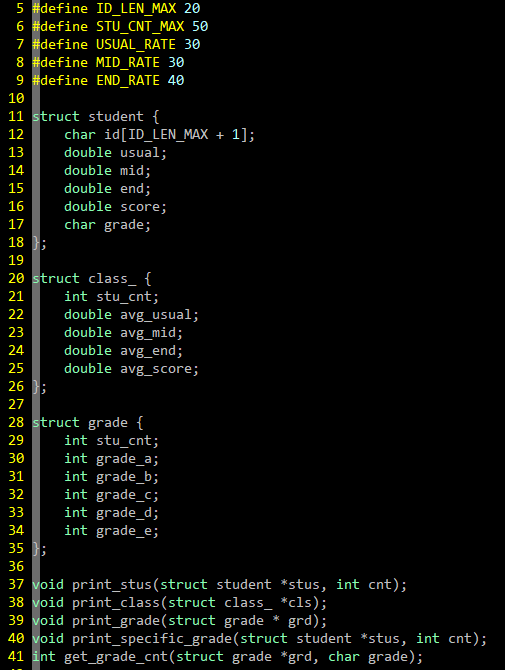
Y

N

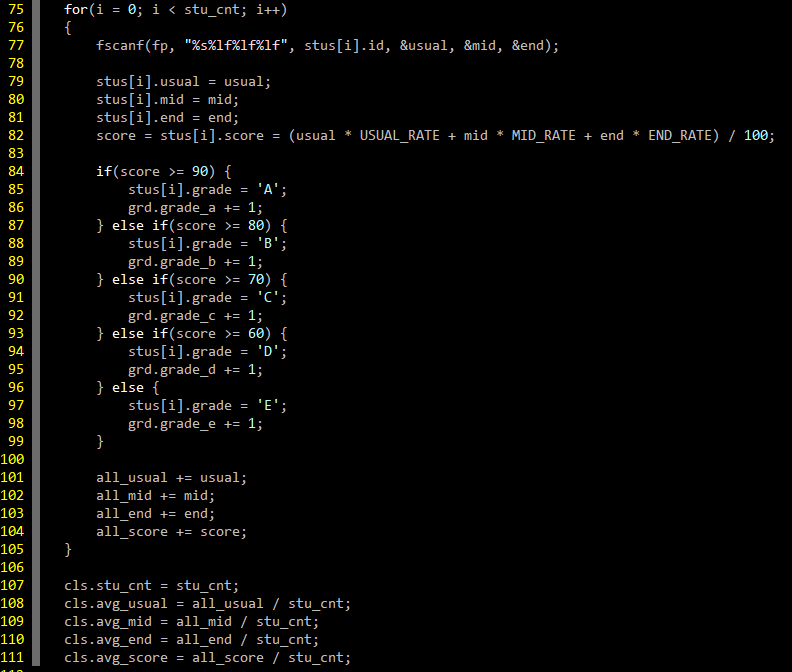
结束

核心代码：

如下为各宏，结构体定义，函数声明，这个函数都为辅助函数，这里不具体列出，详见附带的文件。三个结构体分别代表学生，班级，等级。

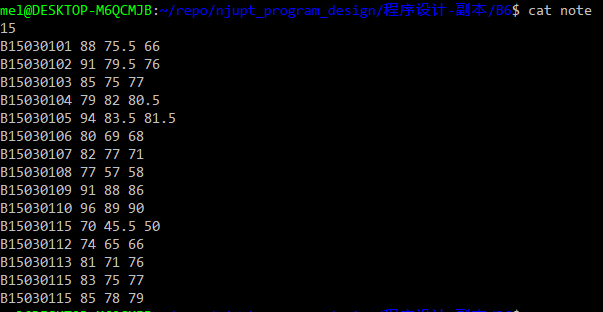


核心的处理在主函数进行，计算了期末成绩以及相应等级，之后的操作都依赖此处的结果。

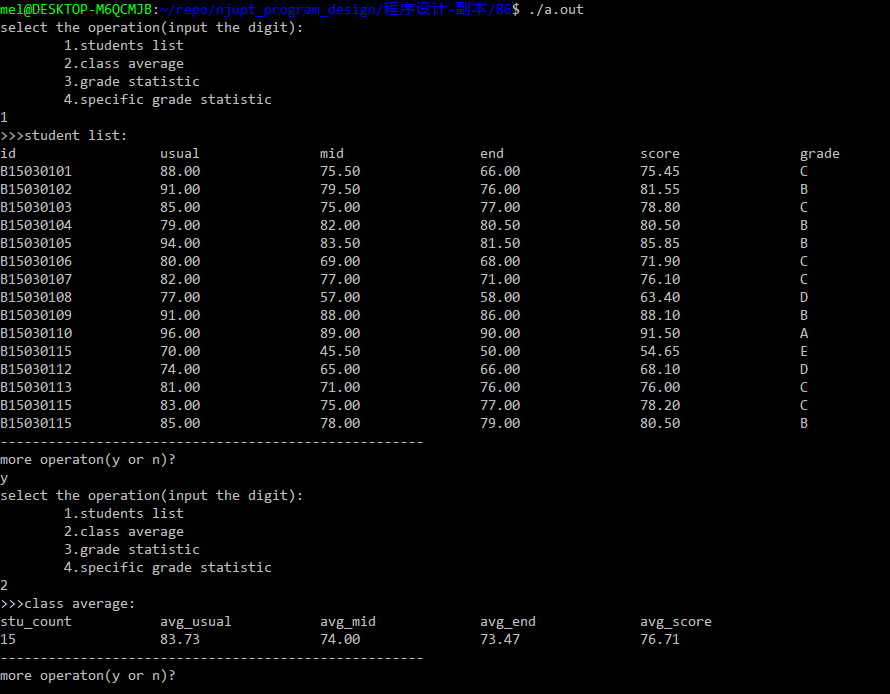


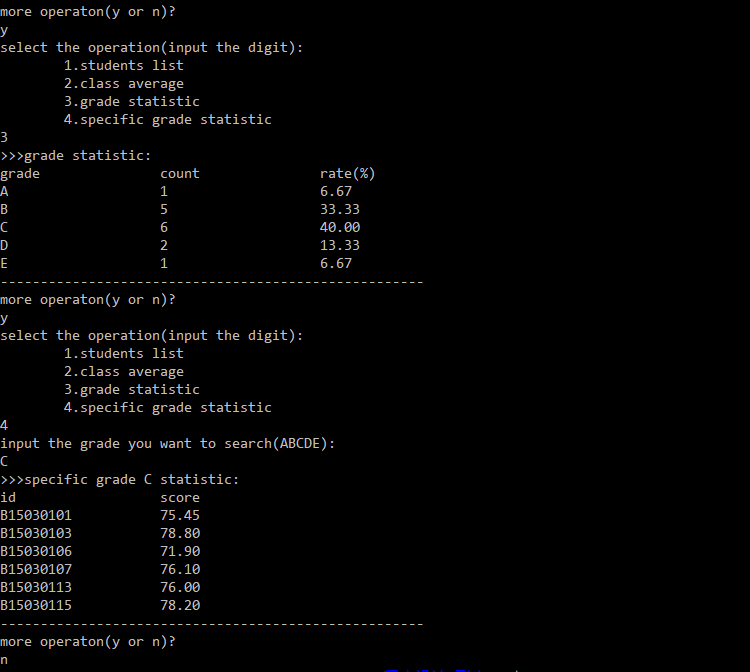
数据文件：

学生成绩文件 note



运行图：





B7 模拟电信计费系统的设计与实现

**课题要求：**

（1） 计费功能。根据存放在源数据文件中的通话记录和长途费率文件对每一条通话记录计算其通话费用，并将结果保存在费用文件中。其中：

通话费的计算方法如下：

通话费＝长途电话费＋本地电话费

长途电话费＝费率（元/分钟）×通话时长（分钟）

(通话时长不满1分钟的按1分钟计算)

本地电话费为：3分钟以内0.3元，以后每1分钟递增0.2元。

（2） 话费查询。输入一个电话号码，从费用文件中统计该电话号码的所有本地话费、长途话费，并从用户文件中查找其用户名，最后在屏幕上显示：

用户名 电话号码 本地话费 长途话费 话费总计

（3） 话单查询。输入一个电话号码，查询并在屏幕显示该用户的所有通话记录，格式为：

用户名 主叫电话号码 被叫电话号码 通话时长

流程图：

开始

显示功能模块

计算通话费用单

查询个人通话费用

查询个人通话记录

退出

是否再次操作

Y

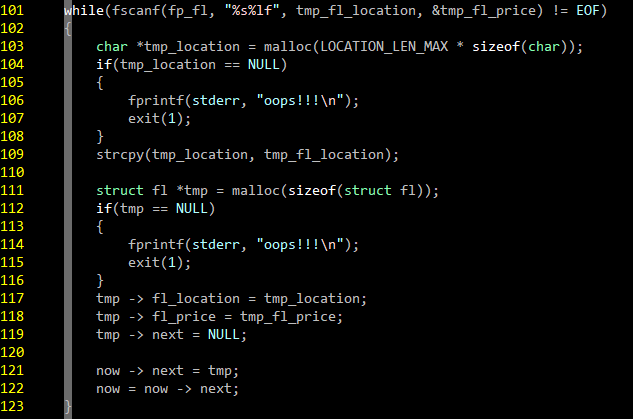
N

结束

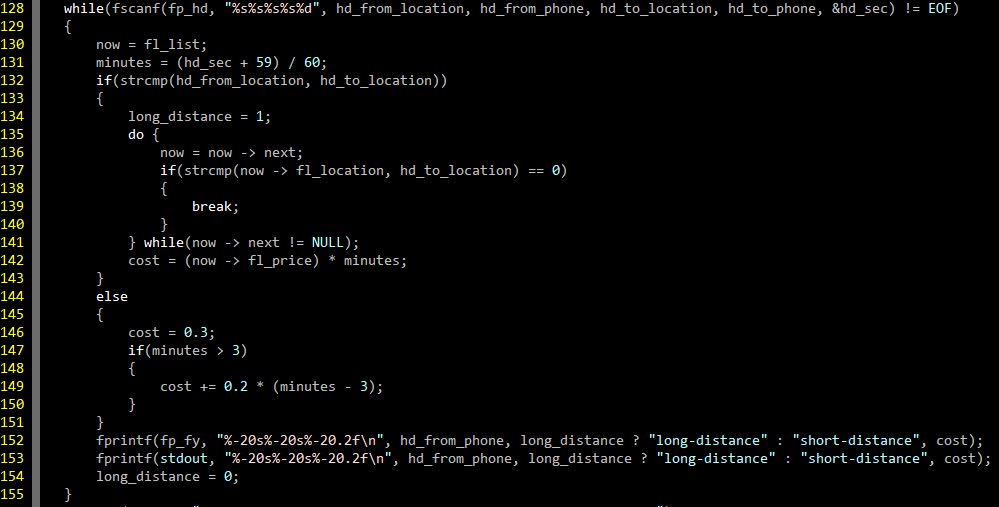
核心代码：

如下为计算通话费单的核心代码段：

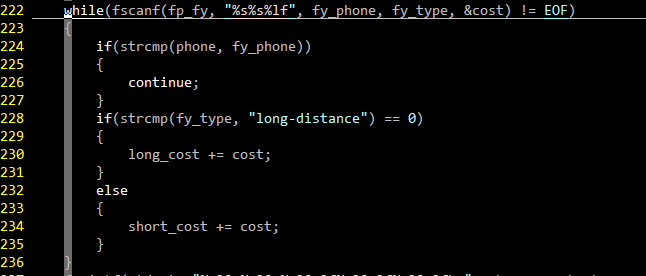
如下代码将各区费率以链表形式存储。



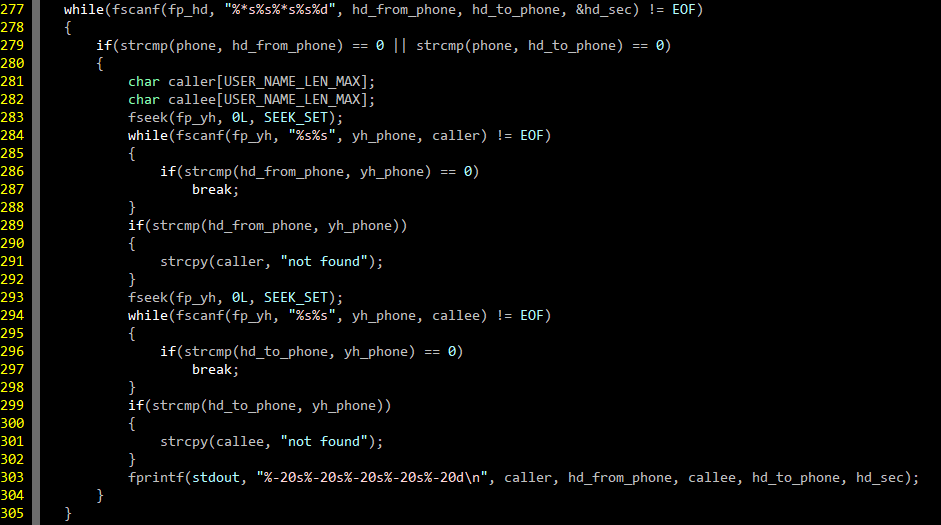
如下代码根据费率及区号计算话费。



如下为用户通话费用查询的核心代码段：

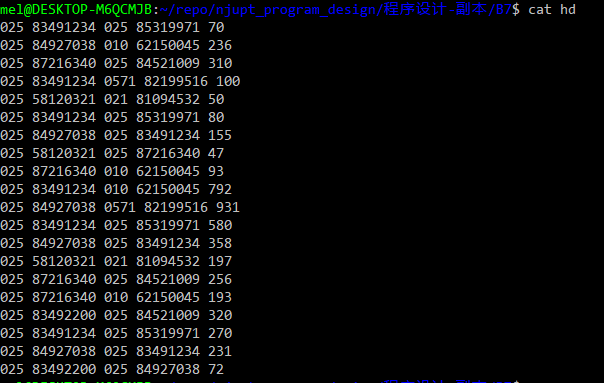


如下为用户通话记录查询的核心代码段



数据文件：

话单文件 hd



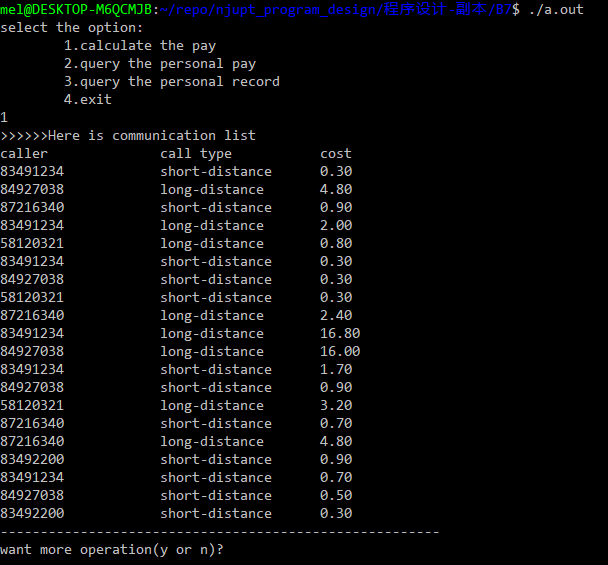
费率文件 fl

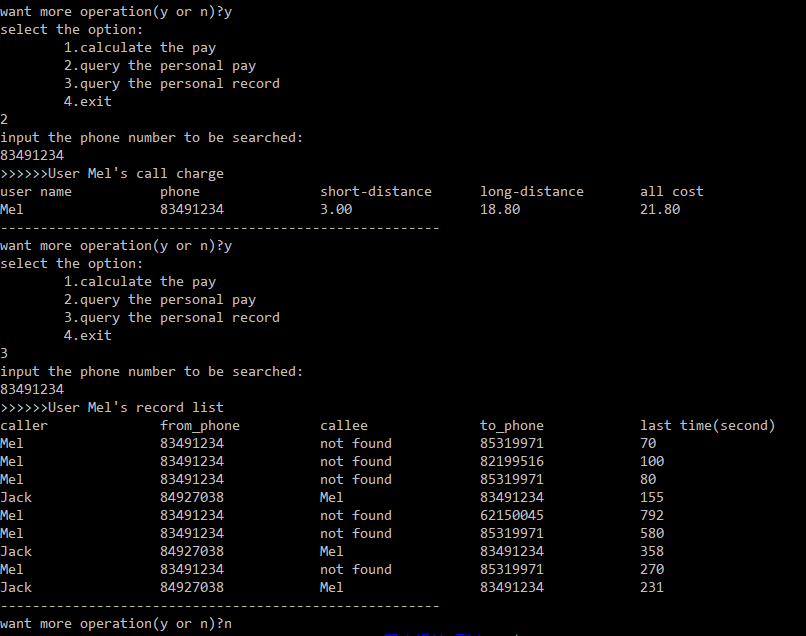


用户文件 yh



运行图：





软件设计小结

写程序，感觉还是自上而下的方式写比较快速且不会出错。提前确定需要的函数，需要定义的宏，以及全局变量。虽然不可能刚开始就把程序需要的一切都考虑到，但可以在已经确定的初始框架的基础上，写的过程中再慢慢完善。不过如果程序开写之后发现对整体框架需要改动的地方太多，就应该思考是不是一开始就规划错了。

当然要尽量考虑程序的通用性，比如使用动态分配内存，而不是使用数组，那样的程序失去了源数据的延展性。对于C语言，我可以自信地说是很熟悉了，C语言中纯语言的几乎所有的知识点，我基本都有过了解，并参阅了各种博客以及书籍，偏僻如位域以及内存对齐等知识点也曾做过深入了解。所以写的过程中并没有遇到什么语法方面的问题，不过由于代码量太少，对库函数的函数头不是很熟悉，所以写的过程中需要查看库函数声明。我是比较推崇写代码多用库函数的，比如qsort，strtoul这些，因为使用这些现成的“万无一失”的代码，能让我们更多的关心程序的核心实现。

这7个程序，只有第三个（实现编辑器）有一些难度，因为很少写关于流的代码，但静下心来，确定了实现的基本准则之后，发现所有功能的实现都是一个套路，颇有一番“云开雾散”的感觉。编程给人的乐趣应该就来源于此吧。

希望自己今后也可以从事编程这件快乐的事情吧。