第十章 C语言课程设计

引言：

学完本章内容以后，你将能够

设计实现学生信息管理系统

学习项目的设计、管理、实现的基本流程

提高代码的实现和调试能力

通过实践进一步加强对C语言理论知识的理解。

10.1 学生管理系统

10.1.1 平台开和发环境简介

本项目使用的平台和开发环境为：Linux系统 + Gcc + Gdb 。

10.1.2 功能描述

使用终端命令界面，作为系统的输入输出界面。信息运行是把信息保存在内存中，系统退出后信息保存至文件。

学生信息：

基本数据信息：

学号： int

姓名： char 128

密码： char 32

年龄： int

班级： int

成绩信息：

数学成绩： int

C语言成绩：int

语文成绩： int

名次： int

1. 实现对学生信息基本信息的录入，其中名次根据成绩自动计算（添加）。学号不能重复。
2. 删除学生信息
3. 修改学生信息
4. 查找学生信息
   1. 根据学号查找
   2. 根据姓名查找
   3. 根据班级群体查找，把属于这个班的学生全部显示
5. 显示所有学生信息列表
   1. 按学号顺序显示
   2. 按名次显示
   3. 按某学科（可选）成绩顺序显示，并在最后显示出此科平均分。
6. 保存学生信息

修改后的信息保存到文件中。当程序启动后从文件中读取信息，至内存。

在做添加删除修改时信息保存至内才中。(在内存中使用链表或数组…保存)。

1. 退出系统
   1. 如果有修改操作要提示是否保存。

教师信息：

教员序号：int

教员姓名：char 128

教员密码：char 32

所任班级： int

1、对老师信息的添加、删除、修改、显示、保存。教员序号不能重复。

登陆功能：

**1、**系统启动后提示登陆界面。使用用户名（学号）和密码登陆。学生和老师都可以登陆。

2、用户名和密码正确登陆成功。显示欢迎界面:”欢迎XX同学（或老师）登陆”

显示操作界面。

3、如果输入用户名和密码错误接续提示，超过3次程序自动退出。

4、除老师和学生外，系统有一个特殊用户超级管理员用户。用户名固定（admin），

也可以固定。

5、超级管理员、老师和学生有不同的操作界面。

管理权限：

超级管理员 ：可以对老师学生做所有操作。

老师 ：可以查看自己的信息，和修改自己的密码。

可以查看所有学生的信息，修改学生的成绩信息。

学生 ：可以查看自己的信息，和修改自己的基本信息。

注意事项：输入姓名、密码、分数、注意不能超出存储长度。如果用户超出后应给于相应的提示。

文件保存的格式：1，张三，\*\*\*\*\*，16…尽量不要存储结构体。

10.1.3 项目实施计划

项目分组：两人一组，相互讨论，独立完成。

项目计划：

分以下几步完成。

第一步：完成学生信息管理的部分。

第二步：完成老师信息管理、和登陆管理。

第三步：完成权限管理。

10.1.4 项目要求

1、程序书写格式，按照高质量编程规范完成。

2、程序的注释不得少于代码的40%。

3、使用makefile对项目进行编译。编译选项加-Wall。

4、代码的不同的功能放在不同的文件中。

参考目录：

/doc/

/src/

main.c

meanu.c

student.c

student.h

teacher.c

teacher.h

tooler.c

tooler.h

common.h

Makefile

可以参考此目录文件也可以根据自己的设计组织自己的目录文件结构。

10.2火车订票系统

10.2.1 平台开和发环境简介

本项目使用的平台和开发环境为：Linux系统 + Gcc + Gdb 。

10.2.2 需求设计

1.每条线路所涉及的信息有：起点、终点、站名、车次、票价、时间、座位号、所经站点(多个)。

2.作为示意系统，全部数据保存至文件中。程序启动后从文件中读取加载至内存中。

3.系统能实现的功能和操作如下：

①.查询路线：根据旅客提出的终点站名输入下列信息：车次、车站名。

②.承办订票业务：根据客户提出的要求查询该车次票额的情况，若尚有余票，则为客户办理订票手续，输出座位号；若已满员或余票额少于订票额，则需重新查询客户要求，若需要可登记排队候补。

③.承办退票业务：根据客户提供的情况（车次、时间、座位号）为客户办理退票手续，然后查询该车次是否有人排队候补，首先询问排在第一的客户，若所退票额能满足他的要求，则为他办理订票手续，否则依次询问其他排队候补的客户。

④登记旅客情况：包括旅客姓名，性别，年龄，家庭住址，联系方式等。

⑤统计功能：将每次车的订票，退票结果统计出来。

⑥管理功能:列车管理员可以通过调用函数来查看车票极其用户情况.可以增加、删除线路信息。

⑦.查询功能:用户可以查询自己需要的车辆信息.

10.2.3 总体设计

 1．总体设计说明：  
①.数据结构设计:  
程序=数据结构+算法，一个好的程序必定有一个好的数据结构。本设计主要考虑车票信息和用户信息的数据结构。  
车票信息采用半十字链表。.横向链表中的每一个结点包含以下内容：车次、起始站、发车时间、指向下一个结点的指针、指向中途站的指针。纵向链表中每一个结点包含以下内容：从始发站开始的依次到站、票价、到达时间，是否已被购买标识，才用mark标识,当mark为1时车票已售出，当mark为0时车票还未售出.以及指向下一个结点的指针。纵向链表采用循环链表，尾指针指向首指针。  
未购票用户信息使用队列来保存，因为考虑到用户会预订票，所以把用户以来订票的时间早晚排在队中。先来先买，后来后买。队列中的每个元素包含以下内容:用户姓名，SFZ号码，想要购票的车次，起始站，要到的站，时间。  
已经购票用户信息使用一个单链表来保存，链表中的每个结点包含以下内容：用户姓名，SFZ号码，已购车票的车次，出发时间，到达时间。

车票信息链表示意图:

车次、 起始站、终点站、发车时间、 到站时间、 票价、 座位号

未购票用户信息队列:

用户姓名  
 购票的车次  
 起始站  
 要到的站  
 达到时间  
用户姓名  
 购票的车次  
 起始站  
 要到的站  
 达到时间

注:座位号包含在车次号中

已购票用户信息链表示意图:

用户姓名  
 SFZ号码  
 已购车票的车次号  
 出发时间  
 到达时间  
注:座位号包含在车次号中

10.2.4 界面设计

1、程序开始运行提示：

请选择用户操作：

1、登录

2、新用户注册

2、旅客用户登录后界面：

1、用户基本信息维护

2、查询操作

子菜单如下：

a、按车次，车次信息。

b、按车次、日期，查询座位信息。

a、按始发站、终点站、日期查询座位信息。

3、订票操作

4、退票操作

5、注销

3、管理员用户界面：

1、管理旅客用户信息

子菜单如下：

a、旅客用户查询

b、旅客用户信息列表显示

c、旅客用户增加

d、旅客用户删除

e、旅客用户修改

2、管理车次信息

子菜单如下：

a、车次查询

b、车次列表信息显示

c、车次增加

d、车次删除

e、车次修改

3、统计

4、用户信息管理

5、注销

10.2.5 项目实施计划

项目分组：两人一组，相互讨论，独立完成。

10.3 护理诊疗系统

10.3.1 平台开发和环境简介

本项目使用的平台和开发环境为：Linux系统 + Gcc + Gdb

10.3.2 功能描述

本系统主要功能包括患者的挂号，医生的诊疗，医嘱的制定及护士医嘱的执行3个部分。其数据流向见下图：



护理诊疗系统功能描述图

医生信息、护士信息是诊疗系统中管理员根据在职护理人员已录入的信息。患者要想等到护理诊疗首先需要挂号，将自己的患者信息添加到诊疗系统中，并选择主治医生及护士。当医生对该患者进行诊疗时，得出诊疗结果并制定医嘱信息，医嘱分为四类，分别是包括药品医嘱，治疗，输液及其它，制定好的医嘱信息属于待执行医嘱。护士的任务就是完成医生制定的医嘱信息，护士查看自己所有的待执行医嘱，并依次执行医嘱，医嘱执行完成后，医嘱的状态就变成已执行医嘱。

本系统的界面终端字符命令界面作为系统的输入输出界面。系统运行时所有的数据信息保存在内存中，系统退出后数据信息保存在已经配置好的数据文件中。

10.3.3 需求详细描述

**10.3.3.1 人员信息注册**

1．患者挂号功能：完成患者基本信息的录入

完成患者ID, 姓名，密码，性别，年龄，主治医生，护理护士，入院时间，初步诊断等基本信息的登记。注：患者ID最好可以自动生成，主治医生，和护理护士由挂号时选择。密码初始化为患者ID。

2. 医生信息的系统录入

完成医生ID, 姓名，密码，性别，科室，主治项目，患者个数的信息记录。密码初始化为医生ID。

3. 护士信息的系统录入

完成护士ID,姓名，性别，科室，护理级别，护理人数，待执行医嘱个数的信息记录。密码初始化为护士ID。

**10.3.3.2 人员信息管理**

1.完成对患者信息的显示，查询，删除，密码初始化功能。

信息查询包括患者基本信息，诊疗信息。

2完成对医生信息的显示，查询，修改，删除，密码初始化功能。

信息查询包括查询医生的基本信息，患者诊断信息； 按科室查询医生信息。

3完成对护士信息的显示，查询，修改，删除，密码初始化功能

信息查询包括查询护士的基本信息，患者护理信息；按护理等级查询护士信息。

**10.3.3.2 医生诊断系统**

完成对已挂该医生的患者做出诊断结果并开出医嘱信息。医嘱包括药品医嘱，治疗，输液及其它。

完成对待执行的医嘱信息的修改，删除功能，已执行的医嘱不允许做任何修改及删除。

查看某一个特定患者的所有医嘱信息列表

查看所有已挂号患者列表，并统计患者个数

查看个人信息

修改个人密码

**10.3.3.3 护士护理系统**

查看所有护理的患者列表；并统计护理患者个数

查看所有患者的待执行医嘱列表；并统计工作量

查看某一个特定患者的待执行医嘱

执行医嘱

查看个人信息

修改个人密码

**10.3.3.4 患者信息管理**

查看个人信息

修改个人密码

查看已执行医嘱信息

查看未执行医嘱信息

10.3.4 总体设计

1. 数据结构设计：一个功能完整，健壮的系统都需要提前设计良好的数据存储方式，也就是需要确定数据存储的数据结构，基于本系统的特点及4中物理存取结构的优缺点，本系统采取链式存储结构进行系统数据存储设计。

患者信息采用具有多个指针域的链表进行存储；横向链表表示患者信息，纵向链表可以将已执行医嘱和未执行医嘱保存于不同的链表上。也可以存储于一条链表中，用标记来标识医嘱是否已经执行。当未执行医嘱被护理工作人员执行以后，医嘱的属性要发生相应的改变。

患者结构信息

患者的基本信息，主治医生，护理护士，初诊结果等

医生结构信息

医生的基本信息，科室，主治项目，患者个数

护士结构信息

护士的基本信息，科室，护理等级，患者个数

医嘱结构信息

医嘱ID 医嘱类型 医嘱医生 执行护士

2. 文件读取：系统启动时，按用户选择读取相应的用户信息，核对用户输入，登陆成功则用户信息从硬盘中的文件中读入内存。

（1）管理员登陆成功，将患者信息，医生信息，护士信息读入内存，以链表的形式存储

（2）医务人员登陆成功，将患者信息，相应医务人员信息读入内存，以链表的形式存储

（3）患者登陆成功，将患者信息读入内存，以链表的形式存储

3. 文件写入：系统退出时 ，或者用户信息有修改主动选择保存的时候，数据写入文件；系统退出时内存链表释放空间。

10.3.5 界面设计

一、程序开始，运行提示：

欢迎登陆护理诊疗系统

请选择用户登陆

* 1. 患者登陆
  2. 医务人员登陆
  3. 管理员登陆
  4. 退出

二、患者登陆后界面：

1．个人信息查看

2. 个人信息修改

3. 查看所有医嘱，分为两部分已执行医嘱和待执行医嘱

4. 返回上一层

三、医务人员登陆

医生账号登陆进入医生界面

1. 个人信息维护
2. 查看患者列表
3. 查询患者医嘱列表
4. 医嘱更改
5. 诊断开出医嘱
6. 返回上一层

护士账号登陆进入护理界面

1. 个人信息维护

2．查看患者列表

3. 查看所有待执行医嘱列表

4. 查看某患者待执行医嘱列表

5. 执行医嘱

6. 返回上一层

三、管理员登陆

1. 个人信息维护

2．患者挂号

3. 患者信息维护

4. 医生信息维护

5. 护士信息维护

6. 返回上一层

10.3.6 项目实施计划

项目分组，几人一组相互讨论独立完成。

项目计划：

1. 基本界面设计
2. 管理员功能完成，患者链表，医生链表，护士链表的操作。
3. 医生功能完成，诊疗并出具医嘱。
4. 护士功能完成，执行医嘱。
5. 患者功能完成，查询个人医嘱执行情况。

10.3.7 项目要求

1. 编写代码格式，按照高质量编程规范完成。

2. 程序的注释不得少于代码的40%。特别是全局变量的注释，全局函数功能的注释，结构体及成员的注释，宏定义的注释，程序分支处的注释。

3.使用makefile对项目进行编译。编译选项加-Wall, 尽量去除所有的编译警告。

4.代码的不同功能放在不同的文件中。

参考目录：

/doc/ 系统文档目录

/src/----🡪/inc/ 头文件目录

----🡪/src/ .c源文件目录

----🡪/data/ 数据文件目录

10.4博弈论之博弈策略论证

10.4.1 平台开和发环境简介

本项目使用的平台和开发环境为：Linux系统 + Gcc + Gdb 。

10.4.2 项目描述

一、博弈游戏描述：N个人分成一组，组中每人分别跟同组的其他所有人一一对决M（M不小于5）次，把每次对决的分数进行累加，累加和除以对决次数就是该人的得分。组中得分最多的人为该组的获胜者。两人对决规则如下：每次对决每人可以出“合作”或者出“不合作”。对决的两人如果都出“合作”，则各得到3分。如果两人都出不合作各得到1分。如果一个人出“不合作”另一个人出“合作”，则出“不合作”的得5分，出“合作”的得0分。

二、策略程序：每人完成一个程序，参与和其他人写的程序的对决。策略程序每次根据对手姓名及之前和此人的对决记录，决定每次对决应该出“不合作”还是“合作”。对决后，把跟此人的对决记录保存，以便于以后分析。累计自己的得分，最后和裁判判决结果对照看是否正确。

三、裁判程序：每一组的人共同设计完成一个裁判程序，负责调度其他的人程序，完成轮流对决，并且记录对决分数并把分数和对手名称告之正在参与对决的程序。所有对决完成之后，计算出最后的获胜者。由裁判程序随机决定每个人的同其他人的对决次数M。

10.4.3 程序分析

所有的策略程序和裁判程序进行交互，所以要留下预先规定好的统一的接口。这样每个学员完成的策略程序才能统一和裁判程序进行交互。

交互操作：

1、裁判通知程序准备博弈，并告知博弈对手，策略程序计算后得出自己的决策结果，返回给裁判程序。

2、裁判把博弈结果告知决策程序（策略程序自己跟其他的博弈结果或者其他的两个策略程序博弈的结果）。

3、通知策略程序，所有的博弈对决结束并通知策略程序最终的博弈结果。策略程序审查最终结果是否正确并返回给裁判程序。

裁判程序可以使用system（）函数对策略程序进行调用。裁判程序可以把操作和要内容作为参数传递给策略程序。策略程序通过返回值（main函数的返回值，或者程序exit的值）把结果返回给裁判程序。

传递参数及返回值：

第一个参数：做何种操作，以上3中操作，分别使用“1”、“2”、“3”来表示以上对应的4中操作。

“1”操作：第二个参数为博弈对手的名称。返回值0：合作 1：不合作 <0：错误。

“2”操作：第二个参数：第一博弈方名称。第三个参数：第二博弈方名称。第四个参数为博弈后的得分。“0”：0分 “1”：1分 “3”：3分 “5”：5分。

“3”操作：第二个参数：博弈方名称。第三个参数：博弈方得分。

10.4.4 项目实施计划

项目分组：5人一组，相互讨论共同完成。

10.5 常用函数

1、字符测试函数：

isalnum(测试字符是否为英文或数字）

isalpha（测试字符是否为英文字母）

isascii（测试字符是否ascii码字符）

isblank（测试字符是否为空格符）

isdigit（测试字符是否为阿拉伯数字）

islower（测试字符是否为小写英文字母）

issupper（测试字符是否为大写英文字母）

2、数据转换函数：

atof(将字符转换为浮点型数）

atoi（将字符转换成整形）

atol（将字符串转换成长整型数）

tolower（将大写字母转换成小写字母）

toupper（将小写字母转换成大写字母）

3、内存配置函数：

malloc（配置内存空间）

alloca（配置内存空间）

calloc（配置内存空间）

realloc（配置内存空间）

free（释放内存空间）

4、时间函数

time（取的当前时间）

asctime（将时间和日期转换成字符串）

clock（取进程占用的cpu的大约时间）

ctime（将时间和日期以字符串格式表示）

difftime（计算时间差）

ftime（取得目前的时间和日期）

gettimeofday（取得目前的日期）

localtime（取当地目前的时间和日期）

mktime（将时间数据结构转换成经过的秒数）

5、字符串处理函数

bzero(将一段内存内容全部清空）

memcpy(内存拷贝函数）

memcmp（比较内存内容）

memmove（拷贝内存内容）

memset（将一段内存填入某值）

strchr（查找字符串中第一个出现指定的字符）

strcmp（比较字符串）

strncmp（比较字符串）

strcpy（拷贝字符串）

strdup（复制字符串）

strlen（返回字符串长度）

strstr（在一字符串中查找连续部含指定字符串）

strtok（分割字符串）

6、格式化输出输入函数

fprintf（格式化输出数据至文件）

fscanf（格式换字符串输入）

printf（格式化输出数据）

scanf（格式化输入字符串）

snprintf（格式化字符串输出）

sprintf（格式化字符串输出）

7、随机数函数：

rand（产生随机数）

srand（设置随机数种子）

8、错误处理函数

ferror（检查文件流是否有错误发生）

perror（打印错误圆心信息字符串）

strerror（返回错误原因的描述字符串）