《数据结构》实验报告

班号：

姓名：

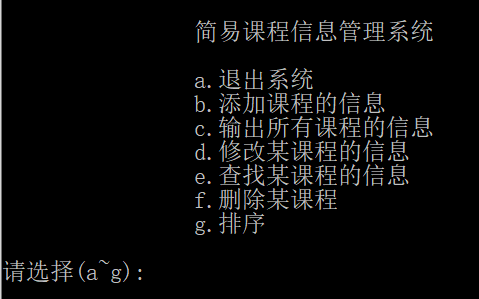
学号：

设计日期：2019.12.18

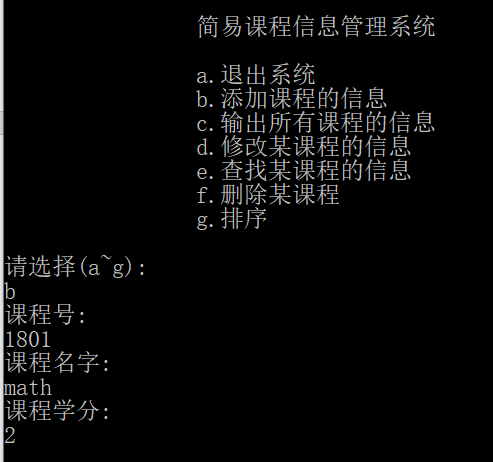
上机环境：Windows 10 + visual studio c++ 6.0

1. **实验结果**

1.选择a，正常退出



**2.选择b，并输入数据1801 math 2**

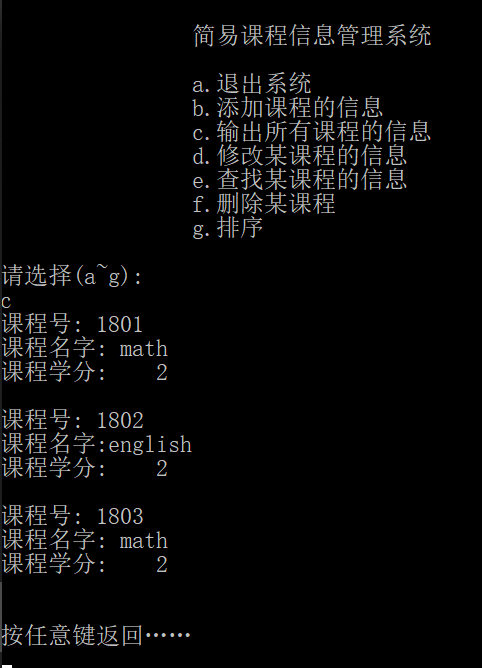


继续选择b输入以下两组数据：

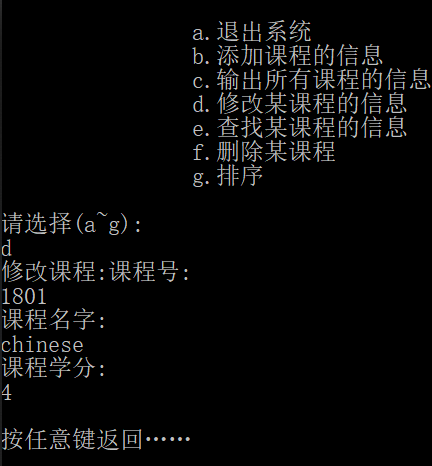
1802 english 2

1803 math 2

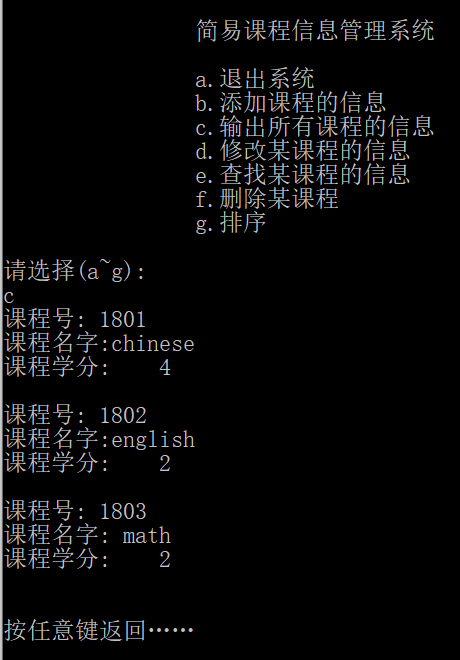
3.选择输出所有课程的信息



4选择d，修改课程号1801的课程名字为chinese，课程学分为4

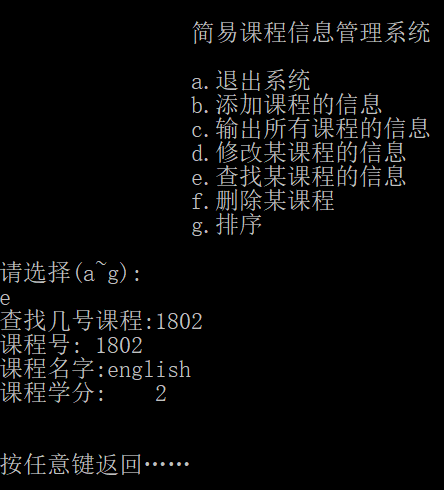


输出所有表

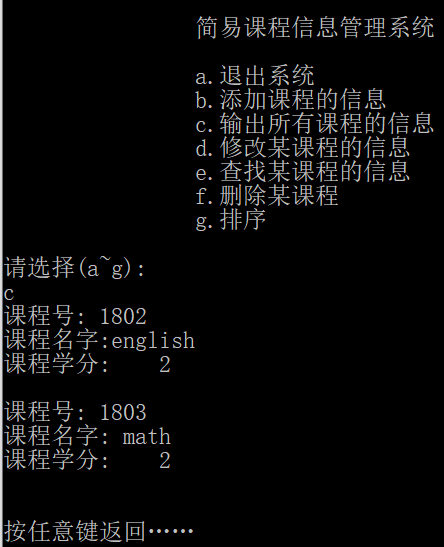


成功修改

查找1802

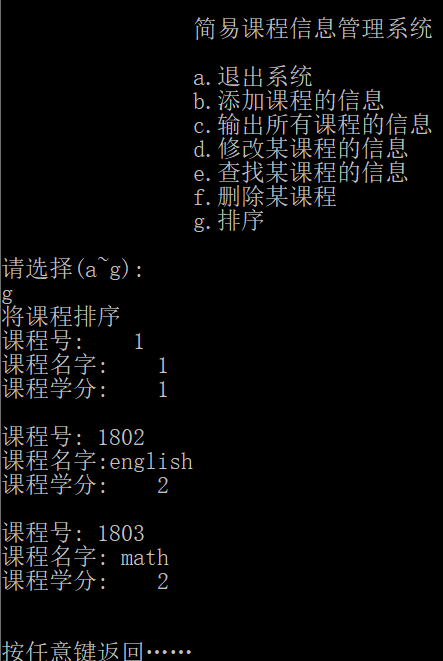


删除1801

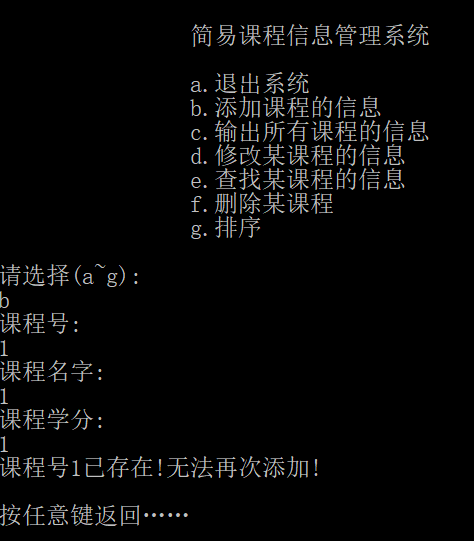


插入一个课程号1的数据

然后排序



1. 结果分析
2. 在添加新的记录之前，要判断是否已经存在，在修改记录之前也要判断是否存在



1. 表中还可以添加其他数据，比如任教老师或者选课的人数。
2. 课程个数也可以随意添加，只要主关键字lessionId不同就视为不同的课程。
3. 在关键字的基础上，排序以及查重都是从关键字出发

**三、程序清单：**

//综合以前实验的功能，设计简易课程信息管理系统，课程信息包括课程号（关键字）、课程名、学分等信息，设计功能主要包括:界面菜单，实现记录的插入、修改、删除、查询、排序等功能。

#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<conio.h>

#define MAXSIZE 20

//课程信息 ,还可添加其他的信息,例如任教老师

typedef struct {

int lessionId;//课程号

char lessionName[50];//课程名字

int credit;//学分

}Information;

//顺序表:由MAXSIZE个Information类型以及一个用来记录位置于长度的length组成

typedef struct{

Information I[MAXSIZE];

int length;//记录长度

}sqlist;

//获取表的当前长度

int len(sqlist head){

return head.length;

}

//初始化表,令长度为0

int create(sqlist &head){

return head.length = 0;

}

//判断表是否为空

bool is\_empty(sqlist head){

if(head.length==0)

return true;

else

return false;

}

//按课程id(关键字)查找,并返回对应的下标length

int search(sqlist head,int lessionId){

int i;

for(i = 0; i<head.length; i++){

if(head.I[i].lessionId == lessionId){

//把此时的i返回,i就是他的下标(类似于数组下标)

return i+1;

}

}//没有找到返回-1

return -1;

}

//用于输入信息 ,输入的信息需要与调用他的函数的信息同步,所以用引用符 或者 使用指针

void input(int &lessionId,char lessionName[],int &credit){

printf("课程号:\n");

scanf("%d",&lessionId);

printf("课程名字:\n");

scanf("%s",lessionName);

printf("课程学分:\n");

scanf("%d",&credit);

}

//添加课程信息

bool addInformation(sqlist &head,int lessionId,char lessionName[],int credit){

int len = head.length;

//调用input,返回时改变了lessionid,lessionname,credit

input(lessionId,lessionName,credit);

//如果被添加的课程号(关键字)已经存在,返回假

if( ( search(head,lessionId) ) != -1)

{

printf("课程号%d已存在!无法再次添加!\n",lessionId);

return false;

}

else{

head.I[len].lessionId = lessionId;

strcpy(head.I[len].lessionName,lessionName);

head.I[len].credit = credit;

head.length++;

return true;

}

}

//修改课程信息 ,直接覆盖关键字位置的对应的其他信息值

void modifyInformation(sqlist &head,int lessionId,char lessionName[],int credit){

int len = lessionId;//关键字

input(lessionId,lessionName,credit);

//如果被添加的课程号(关键字)不存在,执行

if(search(head,lessionId) != -1)

{

head.I[len].lessionId = lessionId;

strcpy(head.I[len].lessionName,lessionName);

head.I[len].credit = credit;

}

else printf("不存在");

}

//删除课程信息,覆盖掉

int dele(sqlist &head,int lessionId){

int i,j;

//遍历整个顺序表

for(i = 0; i<head.length; i++)

{

//如果查找到

if(head.I[i].lessionId == lessionId){

//从 这个查找点开始 ,所有数据往前移动

for(j = i; j<head.length;j++)

{

head.I[j]=head.I[j+1];

}

//删除完毕长度减少

head.length = head.length--;

}

}

return 0;

}

//

void DisNplayOne(sqlist head,int lessionId){

//找到lessionId的下标

int location = search(head,lessionId);

if(location!=-1){

printf("课程号:%5d\n",head.I[location-1].lessionId);

printf("课程名字:%5s\n",head.I[location-1].lessionName);

printf("课程学分:%5d\n",head.I[location-1].credit);

printf("\n");

}

else

printf("该课程id不存在!\n");

}

void DisplayAll(sqlist head){

int i;

if(is\_empty(head))

printf("空\n");

for(i=0; i<head.length; i++)

{

printf("课程号:%5d\n",head.I[i].lessionId);

printf("课程名字:%5s\n",head.I[i].lessionName);

printf("课程学分:%5d\n",head.I[i].credit);

printf("\n");

}

}

void bobblesort(sqlist head)

{

int i,j;

Information temp;

int exchange;

for(i = 0;i<head.length;i++)

{

exchange = 0;

for(j = head.length-1;j>i;j--)

{

if(head.I[j].lessionId<head.I[j-1].lessionId)

{

temp = head.I[j];

head.I[j] = head.I[j-1];

head.I[j-1] = temp;

exchange = 1;

}

if(!exchange)

break;

}

}

DisplayAll(head);

}

void go\_menu()

{

printf("\n按任意键返回……\n");

//system("pause");

getch();

}

char menu()

{

//由于课程关键id是数字,所有选择用字母来选择,避免混在一起

char code;

system("cls");

printf(" \n\t\t简易课程信息管理系统\n\n");

printf("\t\ta.退出系统\n");

printf("\t\tb.添加课程的信息\n");

printf("\t\tc.输出所有课程的信息\n");

printf("\t\td.修改某课程的信息\n");

printf("\t\te.查找某课程的信息\n");

printf("\t\tf.删除某课程\n");

printf("\t\tg.排序\n");

printf("\t\n");

printf("请选择(a~g):\n");

scanf("%c",&code);

return code;

}

int action(){

int code;

char lessionName[50]={0};

int lessionId = 0;

int credit = 0;

sqlist head;

create(head);

printf("建立成功！") ;

while(1){

code = menu();

switch(code)

{

case 'a':

exit (1) ;

go\_menu();

break;

break;

case 'b':

addInformation(head,lessionId,lessionName,credit);

go\_menu();

break;

case 'c':

DisplayAll(head);

go\_menu();

break;

case 'd':

printf("修改课程:");

modifyInformation(head,lessionId,lessionName,credit);

go\_menu();

break;

case 'e':

printf("查找几号课程:");

scanf("%d",&lessionId);

//查找并输出

DisNplayOne(head,lessionId);

go\_menu();

break;

case 'f':

printf("删除几号课程:");

scanf("%d",&lessionId);

//查找并输出

dele(head,lessionId);

go\_menu();

break;

case 'g':

printf("将课程排序\n");

bobblesort(head);

go\_menu();

break;

}

}

}

int main(){

action();

return 0;

}