#include<stdio.h>

#include<stdlib.h>

#define MAXSIZE 100

/\*

5 yue 28

By:gao

学生管理系统

\*/

//学生信息结构体

typedef struct student{

long long stu\_number; //学号

char stu\_name[12]; //姓名

int stu\_age; //年龄

char stu\_college[18]; //学院

char stu\_class[18]; //班级

}stu;

//顺序表的构建

typedef struct LNode \*PtrToAdjVNode;

struct LNode{

int last; //指向最后一个元素下标

stu \*array; //学生指针数组

};

typedef PtrToAdjVNode List;

stu temp; //全局变量

List MakeEmpty(); //顺序表的初始化

void Insert(List L,stu value); //顺序表的按序号插入

void Delete(List L); //顺序表的按序号删除

void Find(List L); //顺序表的按序号查询

void Show(List L); //展示信息

void After(List L,stu value); //顺序表的按序号修改

void login(); //界面信息

void input(); //输入学生信息

void fileWrite(List L); //写入文件

void fileRead(List L); //读取文件

FILE \*fp; //文件变量

int main(){

List L=MakeEmpty(); //顺序表的初始化

int option;

while(1){

login(); //显示界面

scanf("%d",&option);

system("cls"); //清屏函数

switch(option)

{

case 1:

Show(L);

break;

case 2:

input(); //输出信息

Insert(L,temp);

break;

case 3:

input();

After(L,temp);

break;

case 4:

Find(L);

break;

case 5:

Delete(L);

break;

case 6:

fileWrite(L); //文件写入

break;

case 7:

fileRead(L); //文件读取

break;

case 0:

exit(0);

break;

default:

printf("输入格式错误！！！\n按任意键继续...\n");

getchar();

break;

}

system("cls");

}

return 0;

}

List MakeEmpty(){

List L;

L=(List)malloc(sizeof(LNode)); //给指针分配内存

L->array=(stu\*)malloc(sizeof(stu)\*MAXSIZE); //给学生数组分配空间

L->last=-1; //此时还没有学生,即初始值指向-1

return L;

}

//顺序表的按序号插入

void Insert(List L,stu value){

int index;

printf("请输入插入序号:\n");

scanf("%d",&index);

//判断是否可以插入

if(index>L->last+1||L->last+1>=MAXSIZE){

printf("插入失败!!");

getchar();getchar();

return;

}

for(int i=L->last;i>index;i--)

L->array[i+1]=L->array[i]; //从后往前依次挪动 空出一个位置出来

L->array[index]=value; //学生数组的插入

L->last++; //长度增加+ 1

printf("插入成功!!");

getchar();getchar();

}

//顺序表的删除

void Delete(List L){

int index;

printf("请输入你要删除学生的编号:\n");

scanf("%d",&index);

if(index>=0&&index<=L->last){ //判断删除条件

for(int i=index;i<L->last;i++)

L->array[i]=L->array[i+1]; //挪动覆盖掉就行了

L->last--; //长度减小

printf("删除成功!!");

}

else{

printf("删除失败!!");

getchar();getchar();

return;

}

getchar();getchar();

}

//顺序表的按序号查询

void Find(List L){

//我只是写按序号查询，当然你们可以按名字 学号等查询

int index;

printf("请输入你要查询学生的编号:\n");

scanf("%d",&index);

if(index>=0&&index<=L->last){

printf("学号:%d 姓名:%s 年龄:%d 学院:%s 班级:%s\n",

L->array[index].stu\_number,

L->array[index].stu\_name,

L->array[index].stu\_age,

L->array[index].stu\_college,

L->array[index].stu\_class);

}

else{

printf("查询失败");

getchar();getchar();

return;

}

getchar();getchar();

}

//顺序表的显示

void Show(List L)

{

for(int i=0;i<=L->last;i++){

printf("学号:%d 姓名:%s 年龄:%d 学院:%s 班级:%s\n",

L->array[i].stu\_number,

L->array[i].stu\_name,

L->array[i].stu\_age,

L->array[i].stu\_college,

L->array[i].stu\_class);

}

getchar();getchar();

}

//顺序表的修改

void After(List L,stu value){

//我只是写按序号修改，当然你们可以按名字 学号等查询

int index;

printf("请输入你要修改学生的编号:");

scanf("%d",&index);

if(index>=0&&index<=L->last){

L->array[index]=value;

printf("修改成功！！");

}

else{

printf("修改失败");

return;

}

getchar();getchar();

}

//学生管理系统显示界面

void login()

{

printf("\n-------欢迎来到xxx学生信息管理系统---------\n");

printf("------- ---------\n");

printf("------- 1、显示学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 2、添加学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 3、修改学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 4、查询学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 5、删除学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 6、存储学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 7、读取学生信息 ---------\n");

printf("------- ---------\n");

printf("------- 0、退出本系统 ---------\n");

printf("------- ---------\n");

printf("-------------------------------------------by gao---\n");

printf("\n\n请输入你的操作：");

}

//输入信息

void input()

{

printf("请输入学号：");

scanf("%lld",&temp.stu\_number);

printf("请输入姓名：");

scanf("%s",&temp.stu\_name);

printf("请输入年龄：");

scanf("%d",&temp.stu\_age);

printf("请输入学院：");

scanf("%s",&temp.stu\_college);

printf("请输入班级：");

scanf("%s",&temp.stu\_class);

}

void fileWrite(List L){

/\*写入文件过程\*/

if(L->last==-1){ //说明此时没有学生数据

printf("存储学生信息失败");

getchar();getchar();

return;

}

fp = fopen("text.txt", "w"); //打开文件，并且把它设置为可写状态

for(int i=0;i<=L->last;i++){ //依次把所有学生信息存入text.txt

fprintf(fp,"%lld %s %d %s %s\n",

L->array[i].stu\_number,

L->array[i].stu\_name,

L->array[i].stu\_age,

L->array[i].stu\_college,

L->array[i].stu\_class);

}

fclose(fp); //关闭文件

printf("存储学生信息成功!");

getchar();getchar();

}

void fileRead(List L){

/\*读取文件过程\*/

fp = fopen("text.txt", "r"); //打开文件，并且把它设置为可写状态

if(fp==NULL){ //判断文件是否为空

printf("读取学生信息失败!");

getchar();getchar();

return;

}

/\*读取条件必须按照 scanf 里内容读取 且格式 内容必须依依对应，并且不能超过数组最大容量\*/

while(!feof(fp)&&L->last<MAXSIZE){ //feof() 函数的意思 就是 一直读取到文件 最后一个'/0'

++L->last; //添加数据 此时数组下标+1

fscanf(fp,"%d %s %d %s %s\n", //逐渐的读取数据

&L->array[L->last].stu\_number,

L->array[L->last].stu\_name,

&L->array[L->last].stu\_age,

L->array[L->last].stu\_college,

L->array[L->last].stu\_class);

}

fclose(fp); //关闭文件

printf("读取学生信息成功!");

getchar();getchar();

}