**软 件 与 微 电 子 学 院**

数据结构大作业报告

**题 目： 车辆信息管理系统**

**班 级： 软件三班**

**姓 名： 刘昕宇**

**学 号： 1814010319**

目录

1. **需求分析与功能描述**
2. 系统的详细设计与说明
3. 核心算法

**二．数据结构的选择与设计**

**三．算法的选择与设计**

**四．主要代码**

**五．测试结果**

**六．分析与总结**

1. **需求分析与功能描述**
2. **系统的详细设计与说明**

1.Car类：定义一个类包含所有函数。

2.排序函数：将所有输入的信息输出以便用者观看所有信息。

3.添加信息函数：实现了增添一个新的车辆信息的功能，为主要功能函数。

4.修改信息函数：实现了对车辆信息的修改的功能，为主要功能函数。

5.查询信息函数：通过车辆编号，车名，车类型，车级别，出厂时间查找找到所需的车辆信息，为主要功能函数。

6.输出信息函数：输出所有存储的信息，从浏览信息函数里实现。

7.删除信息函数：通过车辆编号删除车辆信息。

**（2）核心算法**

/\*修改车辆信息\*/

void Revise(struct car \*head)

{

int a,b;

char c;

struct car \*p;

cout<<"请输入要修改的汽车编号:";

cin>>a;

p=head;

while(p!=NULL){

if(p->num==a)

break;

p=p->next;

}

if(p==NULL){

cout<<"没有找到该编号的汽车!"<<endl;

getchar();

}

/\*查询车辆\*/

//按编号查询汽车信息

void Num\_Search(struct car \*head)

{

int a;

struct car \*p;

cout<<"请选择您要查询的汽车编号:";

cin>>a;

getchar();

p=head;

while(p!=NULL){

if(p->num==a)

break;

p=p->next;

}

if(p==NULL){

cout<<"没有找到该编号的汽车!"<<endl;

}

else{

cout<<"你所查询的汽车信息如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

cout<<p->num<<"\t"<<p->cname<<"\t\t"<<p->carclass<<"\t"<<p->press<<"\t"<<p->sort<<"\t"<<p->time<<"\t"<<p->price<<endl;

}

}

/\*排序车辆\*/

void Num\_Sort(struct car \*head)

{

struct car \*a[1000],\*p,\*p1,\*temp;

int i,k,index,n=0;

char b;

p1=head;

for(p=head;p;p=p->next)

n++;

for(i=0;i<n;i++){

a[i]=p1;

p1=p1->next;

}

for(k=0;k<n-1;k++)

{

index=k;

for(i=k+1;i<n;i++)

{

if(a[i]->num<a[index]->num)

index=i;

}

temp=a[index];

a[index]=a[k];

a[k]=temp;

}

cout<<"排序成功!"<<endl;

cout<<"是否显示排序结果?(y/n)"<<endl;

cin>>b;

getchar();

switch(b){

case 'n':

break;

case 'y':

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

for(i=0;i<n;i++){

cout<<a[i]->num<<"\t"<<a[i]->cname<<"\t\t"<<a[i]->carclass<<"\t"<<a[i]->press<<"\t"<<a[i]->sort<<"\t"<<a[i]->time<<"\t"<<a[i]->price<<endl;

}

break;

default:

cout<<"您的输入有误!"<<endl;

break;

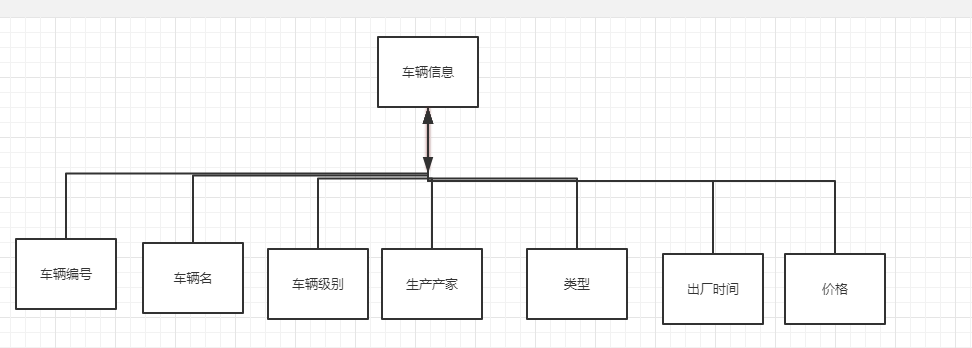
}

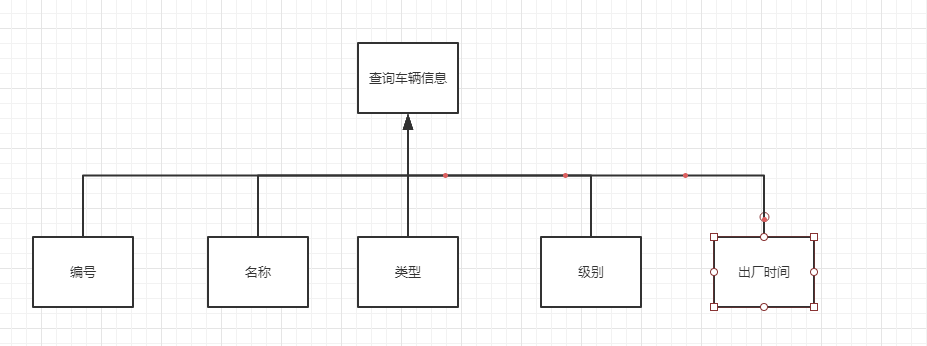
}

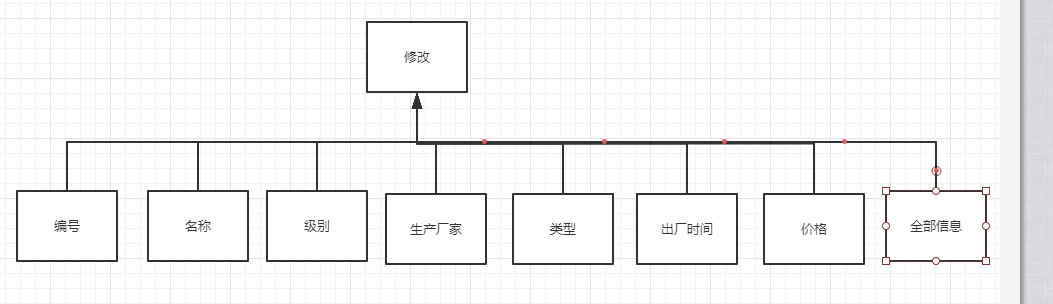
1. **数据结构的选择与设计**

车辆的信息包括：汽车编号，汽车名，汽车级别，生产厂家，类型，出厂时间和价格等，通过链表来存储每一辆汽车的上述信息，并在内存中分配相应的空间。

1. **算法的选择与设计**







1. **主要代码**

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

#include<time.h>

#include <windows.h>

#define D for(j=0;j<i;j++) printf(" ");

using namespace std;

//定义结构体

struct car

{

int num;

char cname[50];

char carclass[20];

char press[50];

char sort[50];

char time[20];

float price;

struct car \*next;

};

struct car \*creatcar(); //创建链表

struct car \*addcar(struct car \*head); //添加汽车

int is\_exist(struct car \*head,int m); //验证新添加的汽车编码是否已存在

void deletecar(struct car \*head); //删除汽车

void fprint(struct car \*head); //将链表写入文件

struct car \*load(); //从文件中读取信息并建成链表

void print\_car(struct car \*head); //将链表信息输出

void Search(struct car \*head); //查询汽车信息

void Num\_Search(struct car \*head); //按汽车编号查询汽车

void Carclass\_Search(struct car \*head); //按汽车级别查询汽车

void Class\_Search(struct car \*head); //按类型查询汽车

void Time\_Search(struct car \*head); //按出厂时间查询汽车

void Cname\_Search(struct car \*head); //按汽车名查询汽车

void Revise(struct car \*head); //修改汽车信息

void Sort\_Car(struct car \*head); //对汽车进行排序

void Num\_Sort(struct car \*head); //按汽车编号排序

void Time\_Sort(struct car \*head); //按汽车出厂时间排序

void Price\_Sort(struct car \*head); //按汽车价格排序

void Cname\_Sort(struct car \*head); //按汽车名排序

void Carclass\_Sort(struct car \*head); //按汽车级别排序

int main()

{

int choice,n;

char a,d;

struct car \*head=NULL;

system("cls");

cout<<"汽车信息管理系统"<<endl;

cout<<"1-添加汽车 2-删除汽车"<<endl;

cout<<"3-汽车列表 4-汽车排序"<<endl;

cout<<"5-查询汽车 6-修改汽车"<<endl;

cout<<"7-录入数据 0-退出系统"<<endl;

cout<<"请输入所选择的序号:";

cin>>choice;

getchar();

system("cls");

switch(choice)

{

case 0:

break;

case 1:

head=load();

if(head==NULL){

cout<<"\t\t文件为空"<<endl;

getchar();

break;

}

else{

head=addcar(head);

cout<<"添加成功!"<<endl;

cout<<"是否将新信息保存到文件?(y/n)"<<endl;

cin>>a;

getchar();

switch(a){

case 'n':

break;

case 'y':

fprint(head);

cout<<"保存成功!"<<endl;

getchar();

break;

}

break;

}

case 2:

head=load();

if(head==NULL){

cout<<"\t\t文件为空"<<endl;

getchar();

break;

}

else{

deletecar(head);

getchar();

break;

}

break;

case 3:

head=load();

if(head==NULL){

cout<<"\t\t文件为空"<<endl;

getchar();

break;

}

else{

print\_car(head);

getchar();

break;

}

case 4:

head=load();

if(head==NULL){

cout<<"\t\t文件为空"<<endl;

getchar();

break;

}

else{

Sort\_Car(head);

getchar();

}

break;

case 5:

head=load();

if(head==NULL){

cout<<"\t\t文件为空"<<endl;

getchar();

break;

}

else{

Search(head);

getchar();

}

break;

case 6:

head=load();

if(head==NULL){

cout<<"\t\t文件为空"<<endl;

getchar();

break;

}

else{

Revise(head);

getchar();

break;

}

break;

case 7:

cout<<"注意:输入汽车编码为0时结束!"<<endl;

head=creatcar();

cout<<"是否将输入的信息保存到文件以覆盖文件中已存在的信息?(y/n)"<<endl;

getchar();

cin>>a;

getchar();

switch(a){

case 'n':

break;

case 'y':

fprint(head);

cout<<"保存成功!"<<endl;

getchar();

break;

}

break;

default:

cout<<"您的输入有误,请重新输入!"<<endl;

getchar();

break;

}

}

//录入数据并形成链表

struct car \*creatcar()

{

struct car \*head,\*tail,\*p;

int num,n;char time[20];

char cname[50],carclass[20],press[50],sort[50];

float price;

int size=sizeof(struct car);

head=tail=NULL;

cout<<"请输入汽车编号:";

cin>>num;

cout<<"请输入汽车名:";

cin>>cname;

getchar();

cout<<"请输入汽车级别:";

cin>>carclass;

getchar();

cout<<"请输入生产厂家:";

cin>>press;

getchar();

cout<<"请输入类型:";

cin>>sort;

getchar();

cout<<"请输入出厂时间:";

cin>>time;

getchar();

cout<<"请输入价格:";

cin>>price;

getchar();

while(1){

p=(struct car \*)malloc(size);

p->num=num;

strcpy(p->cname,cname);

strcpy(p->carclass,carclass);

strcpy(p->press,press);

strcpy(p->sort,sort);

strcpy(p->time,time);

p->price=price;

p->next=NULL;

if(head==NULL)

head=p;

else

tail->next=p;

tail=p;

do{

cout<<"请输入汽车编号:";

cin>>num;

n=is\_exist(head,num);

if(n==0)

break;

else

cout<<"您输入的编号已存在,请重新输入!"<<endl;

}while(1);

if(num==0)

break;

else

{

cout<<"请输入汽车名:";

cin>>cname;

getchar();

cout<<"请输入汽车级别:";

cin>>carclass;

getchar();

cout<<"请输入生产厂家:";

cin>>press;

getchar();

cout<<"请输入类型:";

cin>>sort;

getchar();

cout<<"请输入出厂时间:";

cin>>time;

getchar();

cout<<"请输入价格:";

cin>>price;

getchar();

}

}

return head;

}

//插入结点，并且插入后仍按一定顺序

struct car \*addcar(struct car \*head)

{

struct car \*ptr,\*p1,\*p2,\*p;

char cname[50],carclass[20],press[50],sort[50];

int size=sizeof(struct car);

int num,n=1;char time[20];

float price;

do{

cout<<"请输入汽车编号:";

cin>>num;

n=is\_exist(head,num);

if(n==0)

break;

else

cout<<"您输入的编号已存在,请重新输入!"<<endl;

}while(1);

cout<<"请输入汽车名:";

cin>>cname;

getchar();

cout<<"请输入汽车级别:";

cin>>carclass;

getchar();

cout<<"请输入生产厂家:";

cin>>press;

getchar();

cout<<"请输入类型:";

cin>>sort;

getchar();

cout<<"请输入出厂时间:";

cin>>time;

getchar();

cout<<"请输入价格:";

cin>>price;

getchar();

p=(struct car \*)malloc(size);

p->num=num;

strcpy(p->cname,cname);

strcpy(p->carclass,carclass);

strcpy(p->press,press);

strcpy(p->sort,sort);

strcpy(p->time,time);

p->price=price;

p2=head;

ptr=p;

while((ptr->num>p2->num)&&(p2->next!=NULL)){

p1=p2;

p2=p2->next;

}

if(ptr->num<=p2->num){

if(head==p2)

head=ptr;

else{

p1->next=ptr;

p->next=p2;

}

}

else{

p2->next=ptr;

p->next=NULL;

}

return head;

}

//验证添加的汽车编号是否已存在

int is\_exist(struct car \*head,int m)

{

struct car \*p;

p=head;

while(p!=NULL){

if(p->num==m)

break;

p=p->next;

}

if(p==NULL)

return 0;

else

return 1;

}

//将新链表写入文件中

void fprint(struct car \*head)

{

FILE \*fp;

char ch='1';

struct car \*p1;

if((fp=fopen("f1.txt","w"))==NULL){

cout<<"File open error!"<<endl;

exit(0);

}

fputc(ch,fp);

for(p1=head;p1;p1=p1->next){

fprintf(fp,"%d %s %s %s %s %s %f\n",p1->num,p1->cname,p1->carclass,p1->press,p1->sort,p1->time,p1->price);

}

fclose(fp);

}

//从文件中读取汽车信息

struct car \*load()

{

FILE \*fp;

char ch;

struct car \*head,\*tail,\*p1;

head=tail=NULL;

if((fp=fopen("f1.txt","r"))==NULL){

fp=fopen("f1.txt","w");

cout<<"公司还没有任何汽车，快去购买一些吧！"<<endl;//执行一次

}

ch=fgetc(fp);

if(ch=='1'){

while(!feof(fp)){

p1=(struct car \*)malloc(sizeof(struct car));

fscanf(fp,"%d%s%s%s%s%s%f\n",&p1->num,p1->cname,p1->carclass,p1->press,p1->sort,&p1->time,&p1->price);

if(head==NULL)

head=p1;

else

tail->next=p1;

tail=p1;

}

tail->next=NULL;

fclose(fp);

return head;

}

else

return NULL;

}

//将整个链表的信息输出

void print\_car(struct car \*head)

{

struct car \*ptr;

if(head==NULL){

cout<<"没有信息!"<<endl;

return;

}

cout<<"汽车信息列表如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

for(ptr=head;ptr;ptr=ptr->next)

cout<<ptr->num<<"\t"<<ptr->cname<<"\t\t"<<ptr->carclass<<"\t"<<ptr->press<<"\t"<<ptr->sort<<"\t"<<ptr->time<<"\t"<<ptr->price<<endl;

}

//删除汽车信息

void deletecar(struct car \*head)

{

int a;

char b,ch='1';

struct car \*p1,\*p2;

FILE \*fp;

cout<<"请输入要删除的汽车编号:";

cin>>a;

p1=head;

if(p1->num==a&&p1->next==NULL){ //对于文件中只有一组数据

cout<<"是否清空文件!(y/n)"<<endl;

getchar();

cin>>b;

getchar();

switch(b){

case 'n':

break;

case 'y':

if((fp=fopen("f1.txt","w"))==NULL){

cout<<"File open error!"<<endl;

exit(0);

}

fclose(fp);

cout<<"文件已清空!"<<endl;

}

}

else{

while(p1->num!=a&&p1->next!=NULL){

p2=p1;

p1=p1->next;

}

if(p1->next==NULL){

if(p1->num==a){

p2->next=NULL;

cout<<"是否确定从文件中彻底删除该汽车?(y/n)"<<endl;

getchar();

cin>>b;

switch(b){

case 'n':

break;

case 'y':

fprint(head);

cout<<"删除成功!"<<endl;

getchar();

break;

}

}

else{

cout<<"没有找到要删除的数据!"<<endl;

getchar();

}

}

else if(p1==head){

head=p1->next;

cout<<"是否确定从文件中彻底删除该汽车?(y/n)"<<endl;

getchar();

cin>>b;

switch(b){

case 'n':

break;

case 'y':

fprint(head);

cout<<"删除成功!"<<endl;

getchar();

break;

}

}

else{

p2->next=p1->next;

cout<<"是否确定从文件中彻底删除该汽车?(y/n)"<<endl;

getchar();

cin>>b;

switch(b){

case 'n':

break;

case 'y':

fprint(head);

cout<<"删除成功!"<<endl;

getchar();

break;

}

}

}

}

//汽车查询

void Search(struct car \*head)

{

int a;

cout<<"1-按汽车编号查询 2-按汽车名查询"<<endl;

cout<<"3-按汽车类型查询 4-按汽车级别查询"<<endl;

cout<<"5-按出厂时间查询 0-退出查询"<<endl;

cout<<"请输入所选择的编号:";

cin>>a;

getchar();

switch(a){

case 0:

break;

case 1:

Num\_Search(head);

break;

case 2:

Cname\_Search(head);

break;

case 3:

Class\_Search(head);

break;

case 4:

Carclass\_Search(head);

break;

case 5:

Time\_Search(head);

break;

default:

cout<<"您的输入有误!"<<endl;

break;

}

}

//按编号查询汽车信息

void Num\_Search(struct car \*head)

{

int a;

struct car \*p;

cout<<"请选择您要查询的汽车编号:";

cin>>a;

getchar();

p=head;

while(p!=NULL){

if(p->num==a)

break;

p=p->next;

}

if(p==NULL){

cout<<"没有找到该编号的汽车!"<<endl;

}

else{

cout<<"你所查询的汽车信息如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

cout<<p->num<<"\t"<<p->cname<<"\t\t"<<p->carclass<<"\t"<<p->press<<"\t"<<p->sort<<"\t"<<p->time<<"\t"<<p->price<<endl;

}

}

//按汽车名查询汽车信息

void Cname\_Search(struct car \*head)

{

char a[50];

int flag=0;

struct car \*p;

cout<<"请选择您要查询的汽车名:";

gets(a);

p=head;

while(p!=NULL){

if(strcmp(p->cname,a)==0){

flag=1;

break;

}

p=p->next;

}

if(flag==0){

cout<<"没有找到该汽车名的汽车!"<<endl;

}

else{

cout<<"你所查询的汽车信息如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

while(p!=NULL){

if(strcmp(p->cname,a)==0){

cout<<p->num<<"\t"<<p->cname<<"\t\t"<<p->carclass<<"\t"<<p->press<<"\t"<<p->sort<<"\t"<<p->time<<"\t"<<p->price<<endl;

}

p=p->next;

}

}

}

//按汽车级别查询汽车信息

void Carclass\_Search(struct car \*head)

{

char a[50];

int flag=0;

struct car \*p;

cout<<"请选择您要查询的汽车汽车级别:";

gets(a);

p=head;

while(p!=NULL){

if(strcmp(p->carclass,a)==0){

flag=1;

break;

}

p=p->next;

}

if(flag==0){

cout<<"没有找到该汽车名的汽车!"<<endl;

}

else{

cout<<"你所查询的汽车信息如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

while(p!=NULL){

if(strcmp(p->carclass,a)==0){

cout<<p->num<<"\t"<<p->cname<<"\t\t"<<p->carclass<<"\t"<<p->press<<"\t"<<p->sort<<"\t"<<p->time<<"\t"<<p->price<<endl;

flag=1;

}

p=p->next;

}

}

}

//按汽车类型查询汽车信息

void Class\_Search(struct car \*head)

{

char a[50];

int flag=0;

struct car \*p;

cout<<"请选择您要查询的汽车类型:";

gets(a);

p=head;

while(p!=NULL){

if(strcmp(p->sort,a)==0){

flag=1;

break;

}

p=p->next;

}

if(flag==0){

cout<<"没有找到该汽车名的汽车!"<<cout;

}

else{

cout<<"你所查询的汽车信息如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

while(p!=NULL){

if(strcmp(p->sort,a)==0){

cout<<p->num<<"\t"<<p->cname<<"\t\t"<<p->carclass<<"\t"<<p->press<<"\t"<<p->sort<<"\t"<<p->time<<"\t"<<p->price<<endl;

flag=1;

}

p=p->next;

}

}

}

//按汽车出厂时间查询汽车信息

void Time\_Search(struct car \*head)

{

int flag=0;

char a[20];

struct car \*p;

cout<<"请选择您要查询的汽车出厂时间:";

cin>>a;

getchar();

p=head;

while(p!=NULL){

if(strcmp(p->time,a)==0){

flag=1;

break;

}

p=p->next;

}

if(flag==0){

cout<<"没有找到该汽车名的汽车!"<<endl;

}

else{

cout<<"你所查询的汽车信息如下"<<endl;

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

while(p!=NULL){

if(strcmp(p->time,a)==0){

cout<<p->num<<"\t"<<p->cname<<"\t\t"<<p->carclass<<"\t"<<p->press<<"\t"<<p->sort<<"\t"<<p->time<<"\t"<<p->price<<endl;

flag=1;

}

p=p->next;

}

}

}

//修改汽车信息

void Revise(struct car \*head)

{

int a,b;

char c;

struct car \*p;

cout<<"请输入要修改的汽车编号:";

cin>>a;

p=head;

while(p!=NULL){

if(p->num==a)

break;

p=p->next;

}

if(p==NULL){

cout<<"没有找到该编号的汽车!"<<endl;

getchar();

}

else{

cout<<"1-编号 2-汽车名 3-汽车级别"<<endl;

cout<<"4-生产厂家 5-类型 6-出厂时间"<<endl;

cout<<"7-价格 8-修改全部 0-放弃修改"<<endl;

cout<<"请选择你要修改的信息编号:";

cin>>b;

getchar();

switch(b){

case 1:

cout<<"请输入新编号:";

cin>>p->num;

cout<<"修改成功!"<<endl;

getchar();

break;

case 2:

cout<<"请输入新汽车名:";

gets(p->cname);

cout<<"修改成功!\n";

break;

case 3:

cout<<"请输入新汽车级别:";

gets(p->carclass);

cout<<"修改成功!"<<endl;

break;

case 4:

cout<<"请输入新生产厂家:";

gets(p->press);

cout<<"修改成功!"<<endl;

break;

case 5:

cout<<"请输入新类型:";

gets(p->sort);

cout<<"修改成功!"<<endl;

break;

case 6:

cout<<"请输入新出厂时间:";

cin>>p->time;

cout<<"修改成功!"<<endl;

getchar();

break;

case 7:

cout<<"请输入新价格:";

cin>>p->price;

cout<<"修改成功!"<<endl;

getchar();

break;

case 8:

cout<<"请输入新汽车编号:";

cin>>p->num;

cout<<"请输入新汽车名:";

cin>>p->cname;

getchar();

cout<<"请输入新汽车级别:";

cin>>p->carclass;

getchar();

cout<<"请输入新生产厂家:";

cin>>p->press;

getchar();

cout<<"请输入新类型:";

cin>>p->sort;

getchar();

cout<<"请输入新出厂时间:";

cin>>p->time;

getchar();

cout<<"请输入新价格:";

cin>>p->price;

getchar();

cout<<"修改成功!\n";

getchar();

break;

case 0:

break;

default :

cout<<"您的输入有误!"<<endl;

break;

}

cout<<"是否将修改后的信息保存到文件中?(y/n)"<<endl;

cin>>c;

getchar();

switch(c){

case 'n':

break;

case 'y':

fprint(head);

cout<<"保存成功!"<<endl;

getchar();

break;

}

}

}

//汽车排序

void Sort\_Car(struct car \*head)

{

int a;

cout<<"1.按汽车编号排序"<<endl;

cout<<"0-取消排序操作"<<endl;

cout<<"请输入您选择的编号:"<<endl;

cin>>a;

getchar();

switch(a){

case 0:

break;

case 1:

Num\_Sort(head);

break;

default:

cout<<"您的输入有误!"<<endl;

break;

}

}

//按汽车编号排序

void Num\_Sort(struct car \*head)

{

struct car \*a[1000],\*p,\*p1,\*temp;

int i,k,index,n=0;

char b;

p1=head;

for(p=head;p;p=p->next)

n++;

for(i=0;i<n;i++){

a[i]=p1;

p1=p1->next;

}

for(k=0;k<n-1;k++)

{

index=k;

for(i=k+1;i<n;i++)

{

if(a[i]->num<a[index]->num)

index=i;

}

temp=a[index];

a[index]=a[k];

a[k]=temp;

}

cout<<"排序成功!"<<endl;

cout<<"是否显示排序结果?(y/n)"<<endl;

cin>>b;

getchar();

switch(b){

case 'n':

break;

case 'y':

cout<<"编号 汽车名 汽车级别 生产厂家 类型 出厂时间 价格"<<endl;

for(i=0;i<n;i++){

cout<<a[i]->num<<"\t"<<a[i]->cname<<"\t\t"<<a[i]->carclass<<"\t"<<a[i]->press<<"\t"<<a[i]->sort<<"\t"<<a[i]->time<<"\t"<<a[i]->price<<endl;

}

break;

default:

cout<<"您的输入有误!"<<endl;

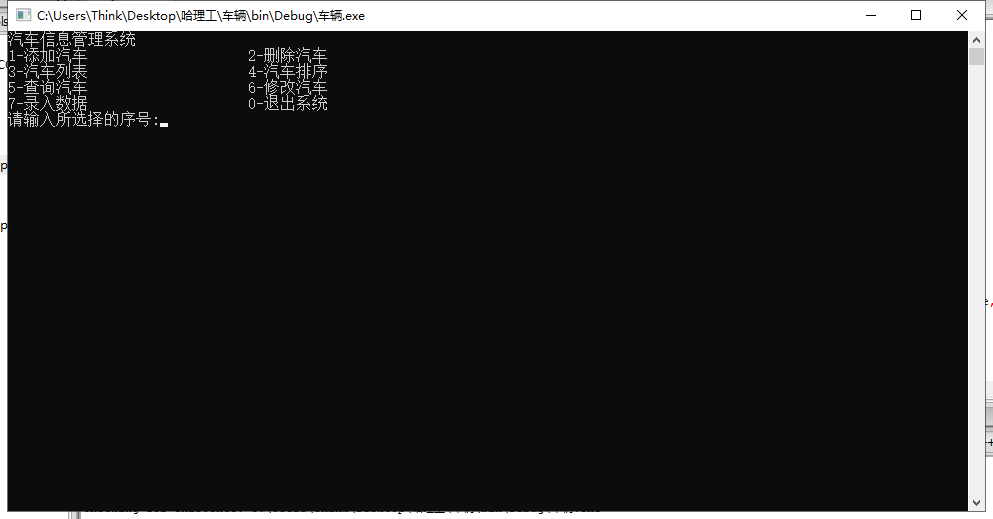
break;

}

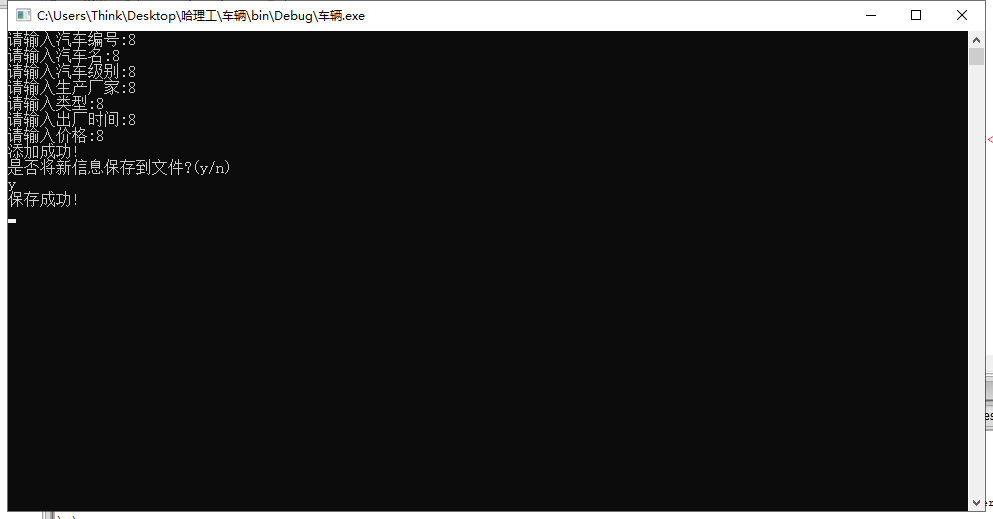
}

1. **测试结果**

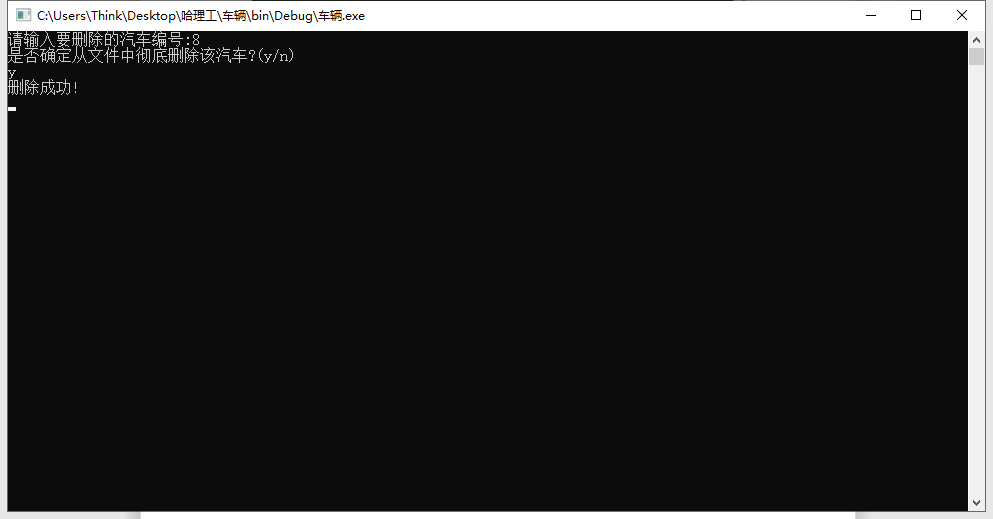
主界面



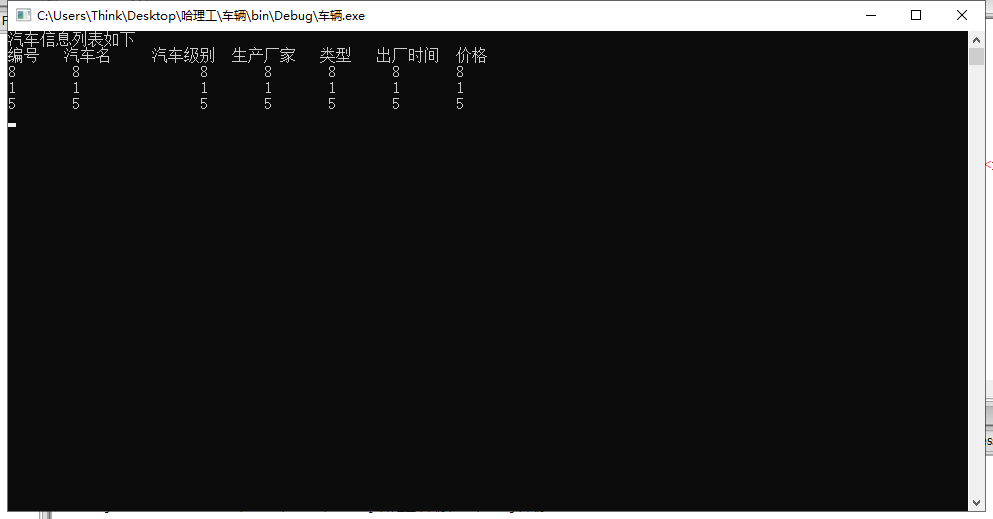
添加车辆信息



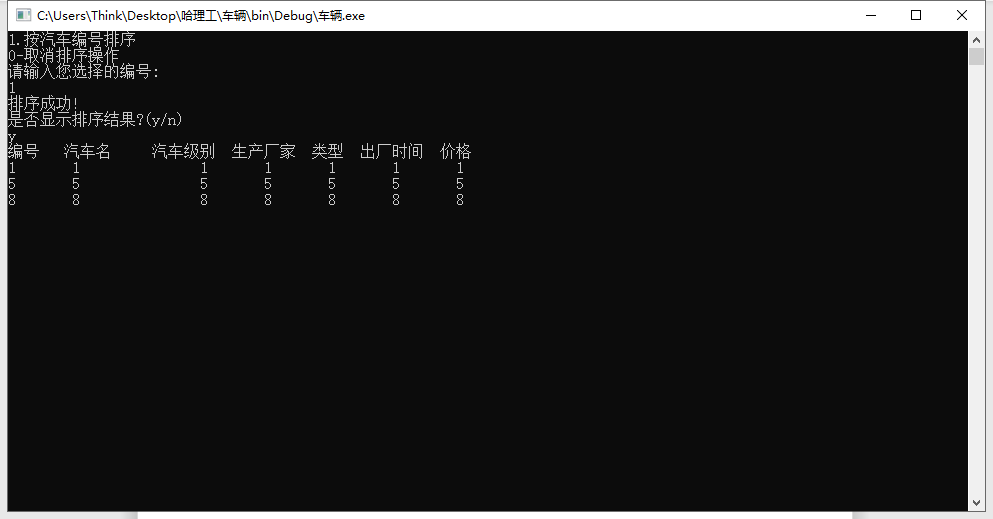
删除车辆信息



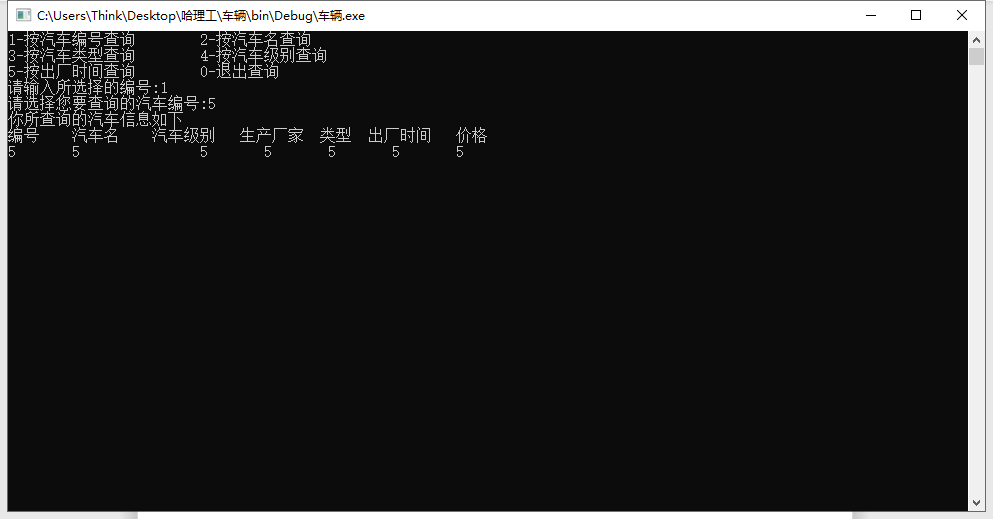
显示车辆信息



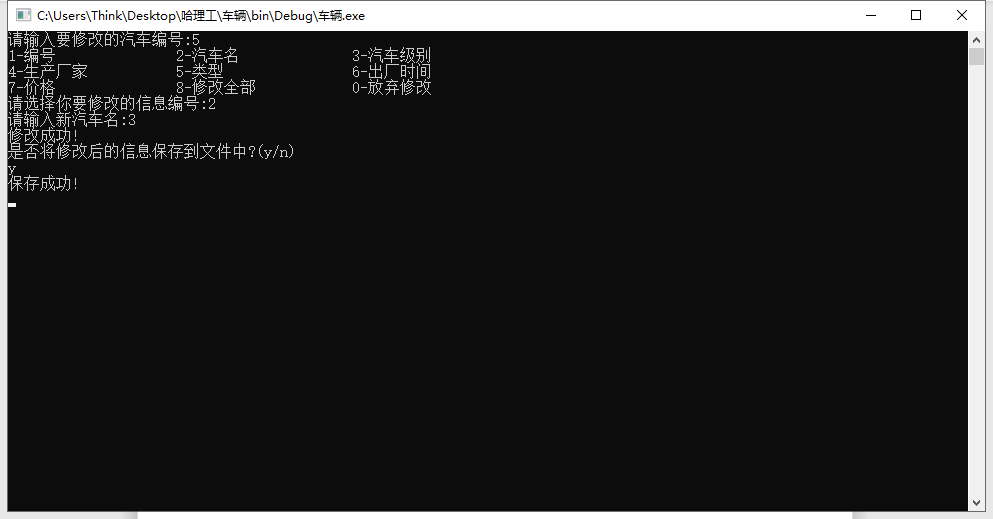
车辆按编号进行排序

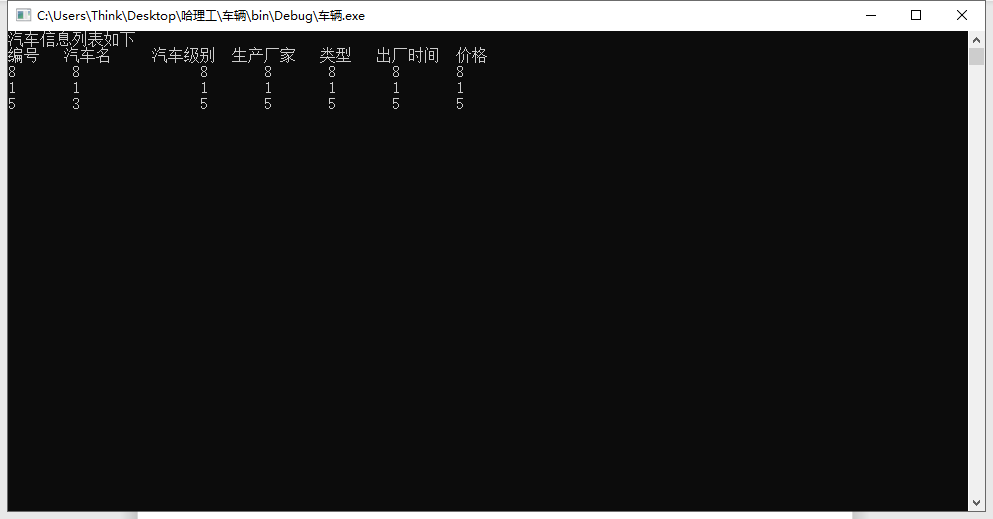


查询车辆信息



修改车辆信息





**六．总结报告**

通过写这个系统，对数据结构有了更深层次的了解与认识，使自己能够更加熟练地运用所学知识，锻炼了自己的思维能力，使自己有了一个全方位的提高。虽然现在仍然有一部分知识运用的不是很熟练，但是我相信我会在学习的道路上越走越远，加油！