

**数据结构课程设计**

**题目：** 通讯录管理系统的设计与实现

**专业班级：** 计算机科学与技术 6班

**学 号：** 20204331

**姓 名：** 王博文

**指导教师：** 隋毅

**报告日期：** 2022年4月20日

计算机科学技术学院

**2022年4月20日**

一、课程设计内容

**设计要求：**对飞机航班信息进行排序和查找。可按航班号、起点站、终点站、起飞时间及到达时间等信息进行查询。

每个航班记录包括八项：航班号、起点站、终点站、航班期、起飞时间、到达时间、机型以及票价。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **航班号** | **起点站** | **终点站** | **航班期** | **起飞时间** | **到达时间** | **机型** | **票价** |
| SC7425 | 青岛 | 海口 | 1、3、6 | 1920 | 2120 | DH4 | 1630 |
| MU5341 | 上海 | 广州 | 每日 | 1420 | 1615 | M90 | 1280 |
| … | … | … | … | … | … | … | … |

航班信息查询系统内容：

1. **航班号**
2. **起点站**
3. **终点站**
4. **起飞时间**
5. **到达时间**
6. **退出系统**

二、算法设计

**此次的实验不是很难，基本上就是对实验六进行的一个变式，唯一的难点就是在c++上对文件进行的操作，这点需要自主学习，排序方面我采用的是上次写的希尔排序，感觉上再就没什么问题了**

三、核心代码实现

#include<iostream>

#include<stdlib.h>

#include<fstream>

using namespace std;

int flag=0;

struct Node {

string flight\_no;

string origin\_station;

string destination;

string flight\_time;

int departue\_time;

int arrival\_time;

string type;

int cost;

};

void insert(Node \*head) {

cout<<"打开文件中:"<<endl;

ifstream fin;

fin.open("航班信息.txt",ios::in);

if(!fin.is\_open()) {

std::cerr<<"未打开文件"<<endl;

exit(0);

}

int i=0;

while(!fin.eof()) {

fin>>head[i].flight\_no>>head[i].origin\_station>>head[i].destination>>head[i].flight\_time>>head[i].departue\_time>>head[i].arrival\_time>>head[i].type>>head[i].cost;

i++;

flag++;

}

fin.close();

cout<<"文件内容写入成功"<<endl;

}

void search1(Node \*head) {

cout<<"请输入想要查找的航班号"<<endl;

string a;

cin>>a;

int i;

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

if(head[i].flight\_no==a)

break;

}

if(i!=flag) {

cout<<"航班号："<<head[i].flight\_no<<" "<<"起点站："<<head[i].origin\_station<<" "<<"终点站："<<head[i].destination<<" ";

cout<<"航班期："<<head[i].flight\_time<<" "<<"起飞时间："<<head[i].departue\_time<<" "<<"到达时间："<<head[i].arrival\_time<<" ";

cout<<"机型："<<head[i].type<<" "<<"票价："<<head[i].cost<<endl;

} else cout<<"查找失败"<<endl;

}

void search2(Node \*head) {

cout<<"请输入想要查找的起点站"<<endl;

string a;

cin>>a;

int i;

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

if(head[i].origin\_station==a)

break;

}

if(i!=flag) {

cout<<"航班号："<<head[i].flight\_no<<" "<<"起点站："<<head[i].origin\_station<<" "<<"终点站："<<head[i].destination<<" ";

cout<<"航班期："<<head[i].flight\_time<<" "<<"起飞时间："<<head[i].departue\_time<<" "<<"到达时间："<<head[i].arrival\_time<<" ";

cout<<"机型："<<head[i].type<<" "<<"票价："<<head[i].cost<<endl;

} else cout<<"查找失败"<<endl;

}

void search3(Node \*head) {

cout<<"请输入想要查找的终点站"<<endl;

string a;

cin>>a;

int i;

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

if(head[i].destination==a)

break;

}

if(i!=flag) {

cout<<"航班号："<<head[i].flight\_no<<" "<<"起点站："<<head[i].origin\_station<<" "<<"终点站："<<head[i].destination<<" ";

cout<<"航班期："<<head[i].flight\_time<<" "<<"起飞时间："<<head[i].departue\_time<<" "<<"到达时间："<<head[i].arrival\_time<<" ";

cout<<"机型："<<head[i].type<<" "<<"票价："<<head[i].cost<<endl;

} else cout<<"查找失败"<<endl;

}

void search4(Node \*head) {

cout<<"请输入想要查找的起飞时间"<<endl;

int a;

cin>>a;

int i;

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

if(head[i].departue\_time==a)

break;

}

if(i!=flag) {

cout<<"航班号："<<head[i].flight\_no<<" "<<"起点站："<<head[i].origin\_station<<" "<<"终点站："<<head[i].destination<<" ";

cout<<"航班期："<<head[i].flight\_time<<" "<<"起飞时间："<<head[i].departue\_time<<" "<<"到达时间："<<head[i].arrival\_time<<" ";

cout<<"机型："<<head[i].type<<" "<<"票价："<<head[i].cost<<endl;

} else cout<<"查找失败"<<endl;

}

void search5(Node \*head) {

cout<<"请输入想要查找的到达时间"<<endl;

int a;

cin>>a;

int i;

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

if(head[i].arrival\_time==a)

break;

}

if(i!=flag) {

cout<<"航班号："<<head[i].flight\_no<<" "<<"起点站："<<head[i].origin\_station<<" "<<"终点站："<<head[i].destination<<" ";

cout<<"航班期："<<head[i].flight\_time<<" "<<"起飞时间："<<head[i].departue\_time<<" "<<"到达时间："<<head[i].arrival\_time<<" ";

cout<<"机型："<<head[i].type<<" "<<"票价："<<head[i].cost<<endl;

} else cout<<"查找失败"<<endl;

}

void shellsort(Node \*head) {

int d,i,j;

Node t;

for(d=flag/2; d>=1; d=d/2) {

for(i=d; i<flag; i++) {

t=head[i];

for(j=i-d; j>=0&&t.departue\_time<head[j].departue\_time; j=j-d) {

head[j+d]=head[j];

}

head[j+d]=t;

}

}

cout<<"排序结果如下："<<endl;

for(i=1; i<flag; i++) {

cout<<"航班号："<<head[i].flight\_no<<" "<<"起点站："<<head[i].origin\_station<<" "<<"终点站："<<head[i].destination<<" ";

cout<<"航班期："<<head[i].flight\_time<<" "<<"起飞时间："<<head[i].departue\_time<<" "<<"到达时间："<<head[i].arrival\_time<<" ";

cout<<"机型："<<head[i].type<<" "<<"票价："<<head[i].cost<<endl;

}

}

int main() {

cout<<"1、对航班的起飞时间进行排序"<<endl<<"2、查询航班号"<<endl<<"3、查询起点站"<<endl;

cout<<"4、查询终点站"<<endl<<"5、查询起飞时间"<<endl<<"6、查询到达时间"<<endl<<"7、结束程序"<<endl;

Node \*head;

head=new Node[15];

insert(head);

while(1) {

int z;

cin>>z;

switch(z) {

case 1:

shellsort(head);

break;

case 2:

search1(head);

break;

case 3:

search2(head);

break;

case 4:

search3(head);

break;

case 5:

search4(head);

break;

case 6:

search5(head);

break;

case 7:

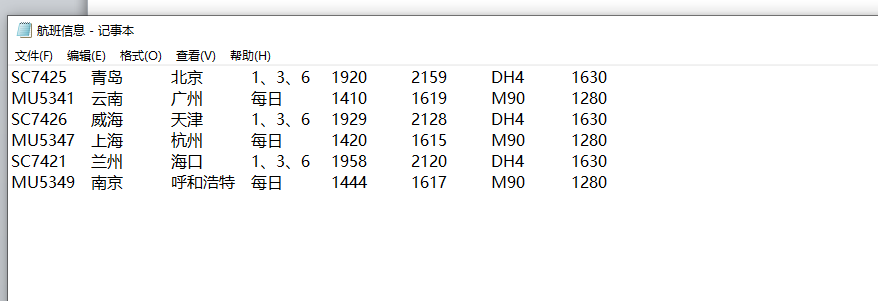
exit(0);

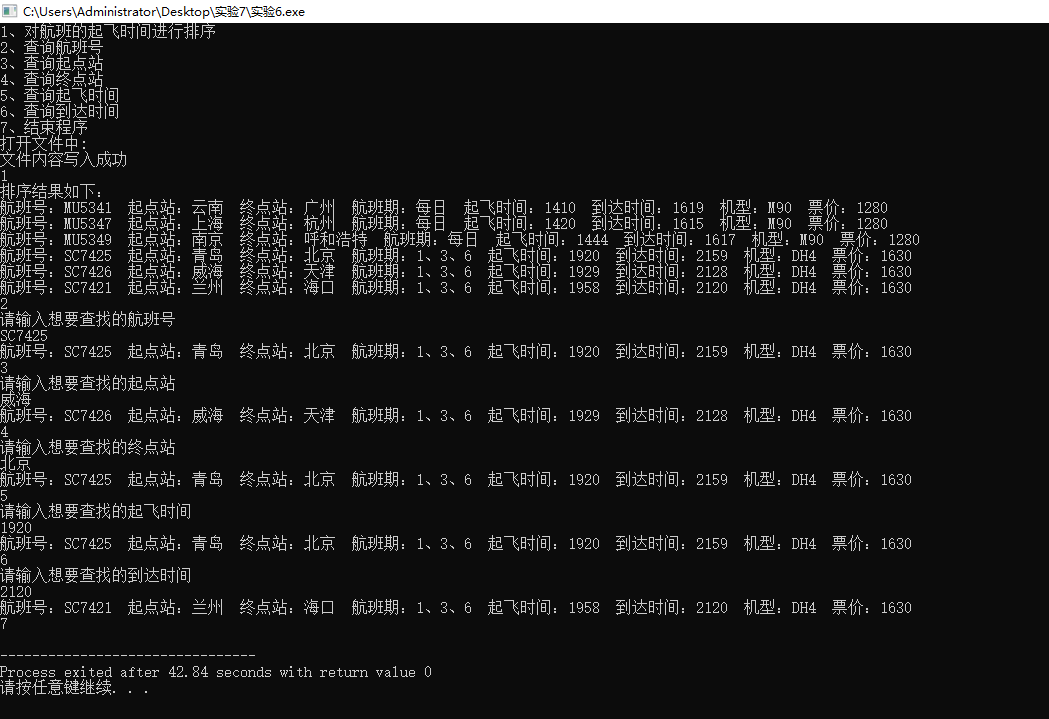
}

}

}

四、测试与分析





五、总结及体会

通过这次的实验，我从中学会了c++对文件的操作，很有收获，再就是巩固了排序的算法，以及结构体的运用。

六、参考书目和网络资源

无