#include<fstream>

#include<iostream>

#include<curses.h>

#include<stdio.h>

#include<iomanip>

#include<stdlib.h>

#include<string.h>

using namespace std;

#define datafile "stud6.txt"

#define indexfile "pri6.txt"

#define sindexfile "sec6.txt"

fstream dfile,ifile,sifile;

int i,indsize,sindsize;

char buffer[100],skey[20];

//function to open

void opener(fstream &file, char \*fn,ios\_base::openmode mode)

{

file.open(fn,mode);

if(!file)

{

cout<<"unable to open a file";

getch();

exit(1);

}

}

class student

{

char dusn[15],name[20],age[5],branch[6],sem[5];

public:

void read();

void pack();

friend int search(char \*);

void remove();

void datadisp();

void unpack();

}s;

class index

{

public:

char iusn[15],addr[5];

void initial();

void write();

}id[50],in;

class sindex

{

public:

char sname[20],susn[15];

void sinitial();

void swrite();

}sid[50],sin;

// function to copy index file to array structure

void index::initial()

{

ifile.open(indexfile,ios::in);

if(!ifile)

{

indsize=0;

return;

}

for(indsize=0;;indsize++)

{

ifile.getline(id[indsize].iusn,15,'|');

ifile.getline(id[indsize].addr,5,'\n');

if(ifile.eof())

break;

}

ifile.close();

}

//function to copy sindex file to array structure

void sindex::sinitial()

{

sifile.open(sindexfile,ios::in);

if(!sifile)

{

sindsize=0;

return;

}

for(sindsize=0;;sindsize++)

{

sifile.getline(sid[sindsize].sname,20,'|');

sifile.getline(sid[sindsize].susn,15,'\n');

if(sifile.eof())

break;

}

sifile.close();

}

// function to update the index file

void index::write()

{

opener(ifile,indexfile,ios::out);

for(i=0;i<indsize;i++)

ifile<<id[i].iusn<<"|"<<id[i].addr<<"\n";

}

//function to upadate the secondary file

void sindex::swrite()

{

opener(sifile,sindexfile,ios::out);

for(i=0;i<sindsize;i++)

sifile<<sid[i].sname<<"|"<<sid[i].susn<<"\n";

}

//function to search based on usn number

int search(char \* fusn)

{

int low=0,high=indsize-1,mid;

while(low <=high)

{

mid = (low+high)/2;

if(strcmp(fusn,id[mid].iusn)==0)

return mid;

if(strcmp(fusn,id[mid].iusn)>0)

low=mid+1;

else

high=mid-1;

}

return -1;

}

// function to read the student record

void student::read()

{

int k;

cout<<"enter the usn number="; scanf("%s",dusn);

if(search(dusn)>=0)

{

cout<<"usn is already present we can't add to index file\n";

return;

}

for(i=indsize;i>0;i--)

{

if(strcmp(dusn,id[i-1].iusn)<0)

id[i]=id[i-1];

else

break;

}

opener(dfile,datafile,ios::app);

cout<<"enter the name=";

scanf("%s",name);

cout<<"enter the age=";

scanf("%s",age);

cout<<"enter the branch="; scanf("%s",branch);

cout<<"enter the semester=";

scanf("%s",sem);

pack();

dfile.seekg(0,ios::end);

k=dfile.tellg();

dfile<<buffer<<"\n";

strcpy(id[i].iusn,dusn);

sprintf(id[i].addr,"%d",k);

indsize++;

for(i=sindsize;i>0;i--)

{

if(strcmp(name,sid[i-1].sname)<0)

sid[i]=sid[i-1];

else if((strcmp(name,sid[i-1].sname)==0) && (strcmp(dusn,sid[i-1].susn)<0))

sid[i]=sid[i-1];

else

break;

}

strcpy(sid[i].sname,name);

strcpy(sid[i].susn,dusn);

sindsize++;

}

//function to pack

void student::pack()

{

strcpy(buffer,dusn);

strcat(buffer,"|");

strcat(buffer,name);

strcat(buffer,"|");

strcat(buffer,age);

strcat(buffer,"|");

strcat(buffer,branch);

strcat(buffer,"|");

strcat(buffer,sem);

strcat(buffer,"|");

}//function to search based on usn number

//function to search based on secondary key

int sec\_search()

{

int pos,j,flag=-1;

cout<<"\nenter the name to search(sec key):";

scanf("%s",skey);

cout<<"the searched record details are :"<<endl;

cout<<setiosflags(ios::left);

cout<<"usn"<<"\t\tname"<<endl;

opener(dfile,datafile,ios::in|ios::out);

for(j=0;j<sindsize;j++)

if(strcmp(skey,sid[j].sname)==0)

{

cout<<sid[j].susn<<"\t\t"<<sid[j].sname<<endl;

flag=j;

}

return flag;

}

// function to remove the record

void student::remove()

{

char rusn[10];

int pos,spos;

cout<<"enter the usn number above listed to delete:";

cin>>rusn;

for(i=0;i<sindsize;i++)

{

if(strcmp(sid[i].susn,rusn)==0)

{

spos=i;

break;

}

}

if(strcmp(sid[spos].sname,skey)==0)

{

pos=search(rusn);

dfile.seekp(atoi(id[pos].addr),ios::beg);

dfile.put('$');

for(i=pos;i<indsize;i++)

id[i]=id[i+1];

indsize--;

for(i=spos;i<sindsize;i++)

sid[i]=sid[i+1];

sindsize--;

}

else

cout<<"usn number and name doesnot match";

}

//function to display the datafile

void student::datadisp()

{

cout<<setiosflags(ios::left);

cout<<setw(16)<<"usn"<<setw(16)<<"name"<<setw(16)<<"age"<<setw(16);

cout<<"branch"<<setw(16)<<"sem";

while(1)

{

unpack();

if(dfile.eof())

break;

}

cout<<endl<<"the index file details are "<<endl;

cout<<setw(10)<<"usn"<<setw(10)<<"address";

for(i=0;i<indsize;i++)

cout<<endl<<setw(10)<<id[i].iusn<<setw(10)<<id[i].addr;

cout<<endl<<"\n the secondary file details are " <<endl;

cout<<setw(20)<<"name"<<setw(15)<<"primary reference";

for(i=0;i<sindsize;i++)

cout<<endl<<setw(20)<<sid[i].sname<<setw(15)<<sid[i].susn;

}

//function to unpack the data file

void student::unpack()

{

dfile.getline(buffer,100,'\n');

i=0;

if(buffer[i]!='$')

while(buffer[i]!='\0')

{

if(buffer[i]=='|')

cout<<"\t\t";

else

cout<<buffer[i];

i++;

}

}//main program

int main()

{

int ch,flag;

in.initial();

sin.sinitial();

curscr;

for(;;)

{

cout<<endl<<"1-read,2-display,3-search,4-delete,5-exit\n";

cin>>ch;

switch(ch)

{

case 1: cout<<endl<<"enter student details : " <<endl;

s.read();

in.write();

sin.swrite();

break;

case 2: opener(dfile,datafile,ios::in);

cout<<"\nthe datafile,indexfile and secondary file" <<endl;

s.datadisp();

break;

case 3:cout<<"To search based on sec key ";

flag=sec\_search();

if(flag==-1)

cout<<"no data record ";

break;

case 4: flag=sec\_search();

if(flag==-1)

cout<<"no data record found";

else

{

s.remove();

in.write();

sin.swrite();

}

break;

default : exit(0);

}

dfile.close();

ifile.close();

sifile.close();

}

return 0;

}