#include<iostream>

#include<fstream>

using namespace std;

#define MAX\_NUM 20

struct employee

{

char name[MAX\_NUM];

int id;

int minuteOfIn;

int minuteOfOut;

int hourOfIn;

int hourOfOut;

int workingTime;

employee \*next;

};

char employeeTxt[MAX\_NUM];

char logTxt[MAX\_NUM];

//function to get length of a string

int strLen(char \*str,int n)

{

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

{

if(str[i]=='\0') return i;

}

return NULL;

}

//function to transform a data from string to int

int strToInt(char \*str,int n)

{

int result=0;

int length=strLen(str,n);

int count=1;

for(int i=length-1;i>=0;i--)

{

int loca=(str[i]-'0');

result+=loca\*count;

count\*=10;

}

return result;

}

//function to get item according to id

employee \*getItem(employee \*head,int id)

{

employee \*p=head;

while(p->id!=id)

{

p=p->next;

}

return p;

}

void main(int argc,char \*argv[])

{

//init an empty employee linkedList

employee \*head=new employee;

head->next=NULL;

head->id=0;

//input two data files

//cin>>employeeTxt;

//cin>>logTxt;

ifstream employeeListTxt(argv[1]);

ifstream logListTxt(argv[2]);

//store employee data

char temp[100]; //employee name temp array

char idTmp[100]; //employee id temp array stored as a string

employee \*tmp=head;

int locationOfDot=0;

while(employeeListTxt.getline(temp,100))

{

for(int i=0;i<100;i++)

{

if(temp[i]==',')

{

locationOfDot=i;

break;

}

}

employee \*employeeNode=new employee; //creat a new employee node

for(int j=0;j<locationOfDot;j++) employeeNode->name[j]=temp[j]; //store employee's name

employeeNode->name[locationOfDot]='\0';

for(int k=locationOfDot+1;k<=strLen(temp,100);k++) idTmp[k-locationOfDot-1]=temp[k];

employeeNode->id=strToInt(idTmp,100); //store employee's id

employeeNode->workingTime=0;

employeeNode->minuteOfIn=0;

employeeNode->minuteOfOut=0;

employeeNode->hourOfIn=0;

employeeNode->hourOfOut=0;

employeeNode->next=NULL; //link the new node

tmp->next=employeeNode;

tmp=tmp->next;

}

//calculate workingTime

char logRecord[100];

int hour=0;

int minute=0;

int locationOfFirstDot=0;

int locationOfSecondDot=0;

int isIn=-1;

while(logListTxt.getline(logRecord,100))

{

for(int i=0;i<100;i++)

{

if(logRecord[i]==',')

{

locationOfFirstDot=i;

break;

}

}

locationOfSecondDot=locationOfFirstDot+6;

for(int j=0;j<locationOfFirstDot;j++) idTmp[j]=logRecord[j];

idTmp[locationOfFirstDot]='\0';

int id=strToInt(idTmp,100);

tmp=getItem(head,id); //find item according to a given id and set tmp to point the item

if(logRecord[locationOfSecondDot+1]=='I') isIn=1;

else isIn=0;

int count=1;

for(int k=locationOfFirstDot+2;k>=locationOfFirstDot+1;k--)

{

hour+=(logRecord[k]-'0')\*count;

count\*=10;

}

for(int p=locationOfFirstDot+3;p<=locationOfFirstDot+4;p++)

{

minute+=(logRecord[p]-'0')\*count;

count/=10;

}

if(isIn==1) //store time info and calculate workingTime if necessary

{

tmp->minuteOfIn=minute;

tmp->hourOfIn=hour;

}else

{

tmp->minuteOfOut=minute;

tmp->hourOfOut=hour;

tmp->workingTime+=(tmp->hourOfOut-tmp->hourOfIn)\*60+(tmp->minuteOfOut-tmp->minuteOfIn);

}

hour=0;

minute=0;

}

//store information to "result.txt"

tmp=head;

ofstream out;

out.open("C:/result.txt");

while(tmp->next!=NULL)

{

tmp=tmp->next;

out<<tmp->name<<","<<tmp->id<<","<<tmp->workingTime/60<<"小?时º¡À"<<tmp->workingTime%60<<"分¤?钟¨®"<<endl;

}

out.close();

}