#include<iostream>

using namespace std;

typedef struct ElemType

{

char info[5];

int carnum;

int time1;

};

typedef struct//栈?

{

ElemType \*base;

ElemType \*top;

int stacksize;

}sqstack;

typedef struct QNode

{

ElemType data;

struct QNode \*next;

}QNode,\*QueuePtr;

typedef struct//队ó列

{

QueuePtr front;

QueuePtr rear;

}LinkQueue;

void Initstack (sqstack &s)//构1造ì一?个?空?栈?

{

s.base=(ElemType\*)malloc(sizeof(ElemType));

if(!s.base)cout<<"存?储洹?分?配?失骸?败悒?！?"<<endl;

s.top=s.base;

}

void Push(sqstack &s,ElemType e)//入?栈?

{

\*s.top++=e;

}

void Pop(sqstack &s,ElemType &e)//出?栈?

{

e=\*--s.top;

}

ElemType gettop(sqstack s)//取?栈?顶￥元a素?

{

if(s.top!=s.base)return \*(s.top-1);

}

void Initqueue(LinkQueue &q)//构1建¨一?个?空?队ó列

{

q.front=q.rear=(QueuePtr) malloc(sizeof(QNode));

if(!q.front)cout<<"存?储洹?分?配?失骸?败悒?！?"<<endl;

q.front->next=NULL;

}

void EnQueue(LinkQueue &q,ElemType e)//入?队ó

{

QueuePtr p=new QNode;

p->data=e;

p->next=NULL;

q.rear->next=p;

q.rear=p;

}

void DeQueue(LinkQueue &q,ElemType e)//出队

{

QueuePtr p=q.front->next;

e=p->data;

q.front->next=p->next;

if(q.rear==p)q.rear=q.front;

delete p;

}

ElemType gethead(LinkQueue &q)//取队头元素

{

if(q.front!=q.rear)

return q.front->next->data;

}

void input(ElemType &e)

{

cout<<"请?输?入?车μ辆?出?入?库a情é况?：阰";

cin>>e.info;

cout<<"请?输?入?车μ牌?号?：阰";

cin>>e.carnum;

cout<<"请?输?入?该?车μ到?达?时骸?间?：阰";

cin>>e.time1;

}

void show(ElemType e)

{

int time2;

cout<<"请?输?入?该?车μ离?开a的?时骸?间?：阰";

cin>>time2;

double f,k;

cout<<"请?输?入?停?车μ场?收?费?标括?准?：阰";

cin>>f;

k=f\*(time2-e.time1);

cout<<"该?车μ在ú停?车μ场?共2停?了?"<<time2-e.time1<<"分?钟ó，?需è交?费?"<<k<<"元a。￡"<<endl;

}

void main()

{

sqstack s;//停?车μ场?

Initstack(s);

cout<<"请?输?入?停?车μ场?的?容╕量?：阰";

cin>>s.stacksize;

sqstack m;//该?栈?用?于?临ⅷ?时骸?停?放?的?车μ辆?

Initstack(m);

LinkQueue q;//便?道台?

Initqueue(q);

for(int i=0;;)

{

int x;

cout<<"1.入?库a"<<endl;

cout<<"2.出?库a"<<endl;

cout<<"3.退?出?"<<endl;

cout<<"请?选?择?：阰";

cin>>x;

if(x==1)

{

if(s.top-s.base>=s.stacksize)//停?车μ场?满ú时骸?，?将?车μ停?在ú便?道台?

{

cout<<"停?车μ场?已?满ú，?请?在ú便?道台?等台?候ò！?"<<endl;

ElemType e;

input(e);

EnQueue(q,e);

cout<<"您ú已?在ú便?道台?排?队ó成é功|，?待鋣停?车μ场?有瓺空?位?您ú方?可é进?入?！?"<<endl;

}

else //停?车μ场?未′满ú，?停?入?停?车μ场?

{

ElemType e;

i++;

input(e);

Push(s,e);

cout<<"停?车μ成é功|，?您ú在ú第台?<<i<<"号?车μ位?！?"<<endl;

}

}

if(x==2)

{

cout<<"请?输?入?您ú的?车μ牌?号?：阰";

int n;

cin>>n;

ElemType e;

e=gettop(s);

if(e.carnum==n)//停?车μ场?的?最?后ó一?个?车μ即′为a所ù找ò车μ辆?

{

Pop(s,e);

show(e);

cout<<"出?库a成é功|！?"<<endl;

i=i-1;

}

else for(;e.carnum!=n&&s.base!=s.top;)//先è将?所ù找ò车μ辆?后ó的?车μ移?出?，?车μ离?开a后ó，?再ù移?入?

{

Pop(s,e);

Push(m,e);

e=gettop(s);

show(e);

cout<<"出?库a成é功|！?"<<endl;

i=i-1;

ElemType r;

while(s.top-s.base!=s.stacksize-1)

{

r=gettop(m);

Pop(m,r);//让?临ⅷ?时骸?停?放?的?车μ辆?返う?回?停?车μ场?

Push(s,r);

}

ElemType t;

t=gethead(q);

DeQueue(q,t);

Push(s,t);

cout<<"车μ牌?号?为a"<<t.carnum<<"的?车μ入?库a成é功|！?"<<endl;

break;

}

}

if(x==3)break;

}

}