**\_\_\_\_2019\_\_\_\_\_\_级\_\_\_\_\_信1901-1\_\_\_\_\_班\_\_\_\_\_2020\_\_年\_\_\_\_11\_\_\_月\_\_13\_\_\_日**

**姓名\_\_\_\_\_李朴凡\_\_\_\_\_\_\_ 学号\_\_\_\_\_\_20194311\_\_\_\_\_\_ 电话\_\_\_\_\_18333135009\_\_\_\_\_\_\_\_**

**1.实验题目**

设停车场内只有一个的停放 n 辆汽车的狭长通道，且只有一个大门可供汽车进出。汽车 在停车场内按车辆到达时间的先后顺序，依次由北向南排列（大门在最南端，最先到达的第 一辆车停放在车场的最北端），若车场内已停满 n 辆汽车，则后来的汽车只能在门外的便 道 上等候，一旦有车开走，则排在便道上的第一辆车即可开入；当停车场内某辆车要离开时， 在它之后开入的车辆必须先退出车场为它让路，待该辆车开出大门外，其 它车辆再按原次序 进入车场，每辆停放在车场的车在它离开停车场时必须按它停留的时间长短交纳费用。试为 停车场编制按上述要求进行管理的模拟程序。

**2．需求分析**

本程序由C语言编写，编写环境为VS2010

1 停车场采用栈式结构，停车场外的便道采用队列结构

2 当车辆要进入停车场时，检查停车场是否已满，未满进入栈，满了则进入便道，即队列。

3 当车辆要求出栈时，该车到栈顶的那些车辆先弹出栈，当车辆出栈完毕后，检查等候队列（便道）中是否有车，有车则从队列头取出一辆车压入栈中。

4 出栈时可以增加一个辅助栈，来保存信息

**3．概要设计**

以栈模拟停车场，以队列模拟车场外的便道，便道停车不计入停车时长。先进的车离开，后进的车停到便道上，然后再进站。

**4．详细设计**

**#include<bits/stdc++.h>**

**using namespace std;**

**int n;**

**typedef int Status;**

**typedef struct Car1**

**{int num;**

**int time1;**

**} CarNode;**

**typedef struct**

**{CarNode \*base;**

**CarNode \*top;**

**int stacksize;**

**}Park;**

**typedef struct Car2**

**{int num;**

**int time1;**

**struct Car2 \*next;**

**} \*CarPtr;**

**typedef struct**

**{CarPtr front;**

**CarPtr rear;**

**int length;**

**}Shortcut;**

**Status InitStack(Park &P)**

**{P.base = (CarNode\*)malloc(n\*sizeof(Car1));**

**if(!P.base ) exit(0);**

**P.top =P.base ;**

**P.stacksize =0;**

**return true;**

**}**

**Status Push(Park &P,CarNode e)**

**{\*P.top ++=e;**

**++P.stacksize ;**

**return true;**

**}**

**Status Pop(Park &P,CarNode &e)**

**{if(P.top ==P.base ) cout<<"停车场为空"<<endl;**

**else**

**{e=\*--P.top ;**

**--P.stacksize ;**

**}**

**return 1;**

**}**

**Status InitQuene(Shortcut &S)**

**{S.front =S.rear =(CarPtr)malloc(sizeof(Car2));**

**if(!S.front ||!S.rear ) exit(0);**

**S.front ->next=NULL;**

**S.length =0;**

**return 1;**

**}**

**Status EnQuene(Shortcut &S,int number,int time)**

**{CarPtr p;**

**p=(CarPtr)malloc(sizeof(Car2));**

**if(!p) exit(0);**

**p->num =number;**

**p->time1 =time;**

**p->next =NULL;**

**S.rear ->next=p;**

**S.rear =p;**

**++S.length ;**

**return 1;**

**}**

**Status Arrival(Park &P,Shortcut &S)**

**{int number,time;**

**cout<<"请输入车号和停车时间："<<endl;**

**cin>>number>>time;**

**if(P.stacksize<n )**

**{CarNode c;**

**c.num =number;**

**c.time1 =time;**

**Push(P,c);**

**cout<<"该车应该停在第"<<P.stacksize <<"号车道"<<endl;**

**}**

**else**

**{EnQuene(S,number,time);**

**cout<<"停车场已满，请暂时停在便道的第"<<S.length <<"个位置"<<endl;**

**}**

**return 1;**

**}**

**Status Leave(Park &P,Park &P1,Shortcut &S){//对离站车辆的处理**

**int number,le\_time,flag=1,money,ar\_time;**

**cout<<"请输入车牌号和出场时刻："<<endl;**

**cin>>number>>le\_time;**

**CarNode e,m;**

**CarPtr w;**

**while(P.stacksize)**

**{**

**Pop(P,e);**

**if(e.num==number)**

**{**

**flag=0;**

**money=(le\_time-e.time1 )\*2;**

**ar\_time=e.time1 ;**

**break;**

**}**

**Push(P1,e);**

**}**

**while(P1.stacksize)**

**{**

**Pop(P1,e);**

**Push(P,e);**

**}**

**// 车从停车场中出**

**if (flag == 0)**

**{**

**if(S.length!=0)**

**{**

**//DeQueue(S,w);**

**m.time1 =le\_time;**

**m.num=w->num;**

**Push(P,m);**

**free(w);**

**cout<<"车牌号为"<<m.num<<"的车已由便道进入停车场" <<endl;**

**}**

**cout<<"停车费为"<<money<<endl<<"现被占用车位数为"<<P.stacksize<<endl;**

**}**

**else cout<<"停车场不存在牌号为"<<number<<"的车"<<endl;**

**return 1;**

**}**

**int main(){**

**int m=1;**

**cout<<"请输入停车场容量："<<endl;**

**cin>>n;**

**getchar();**

**char flag;//选项**

**Park P,Q;**

**Shortcut S;**

**InitStack(P);**

**InitStack(Q);**

**InitQuene(S);**

**while(m)**

**{**

**cout<<"\*\*\*\*\*\*\*\*\*\*\*\*停车场管理程序 \*\*\*\*\*\*\*\*\*\*\*"<<endl;**

**cout<<"请选择操作：A.停车 D.离开 E.退出系统";**

**scanf("%c",&flag);**

**switch(flag)**

**{**

**case 'A':**

**case 'a':**

**Arrival(P,S);break; //车进入停车场**

**case 'D':**

**case 'd':**

**Leave(P,Q,S);break; //车离开停车场**

**case 'E':**

**case 'e':**

**m=0;**

**break;**

**default:**

**cout<<"请重新输入：";**

**break;**

**}**

**getchar();**

**}**

**return 0;**

**}**

**5．调试分析**

1 栈的特点是先进后出，如果先进的车要走，需要另一个栈来存后来的车辆。

2 时间计算，停留在便道上的时间应该减去

3程序还存在些许不做，代码有小部分错误，对于停车时间的问题处理不完美。

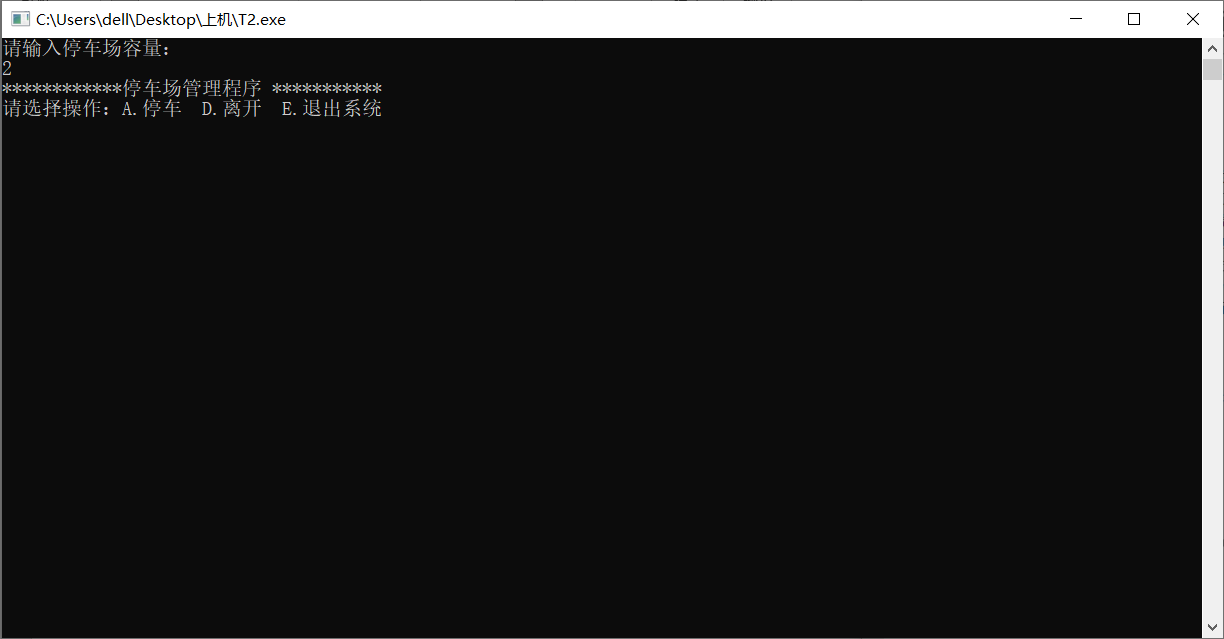
**6．使用说明**

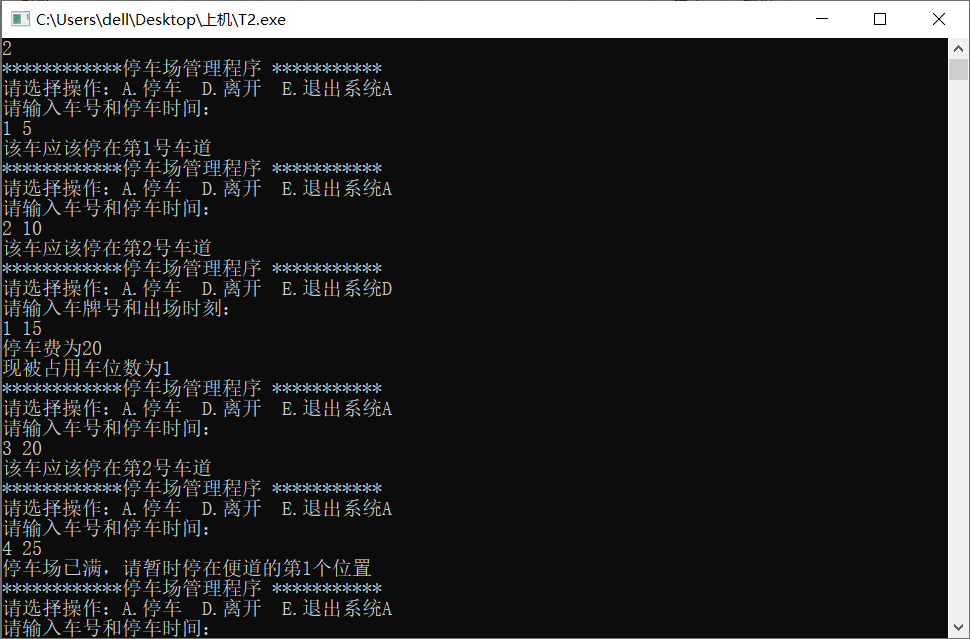
1，选择停车，并选择车号还有停车时间。

2，驶离车库。选择车号开出车库。

3，查看车库中的车辆情况。

4，退出汽车停放管理程序。



**7．测试结果**