//已知总共有45个25\*25矩阵P(k,l), k=1…9, l=0…4；

U(0)=()21\*25, U(0)1,1=1，U(0)i,j=0 if i!=0 or j!=0;

//题目总共要遍历9！种情况(主要是k的随机排列)，本程序只跑一种情况下的结果,假设Q=（1,2,3,4,5,6,7,8,9）

//U(n)是21\*25矩阵，初始情况下U(0)除左上角是1其他都是0

player\_num=0;

n=0;

//初始设置player\_num和n为0

for round in 1:9{

//第一层循环，总共9轮比赛

p\_judge=0;//每一轮比赛先把p\_judge设置为0）

while(p\_judge<=0.999)**{**

//第二层循环，根据p\_judgez值判定是否终止

U(n+1)(row 1)=U(n)(row 1)\*P(n+1,0);

U(n+1)(row 2)=U(n)(row 1)\*P(n+1,1)+ U(n)(row 2)\*P(n+1,0);

U(n+1)(row 3)= U(n)(row 1)\*P(n+1,2)+ U(n)(row 2)\*P(n+1,1) + U(n)(row 3)\*P(n+1,0);

U(n+1)(row 4)= U(n)(row 1)\*P(n+1,3)+ U(n)(row 2)\*P(n+1,2) + U(n)(row 3)\*P(n+1,1) + U(n)(row 4)\*P(n+1,0);

for x in 5:21 {

//第三层循环，U(n+1)的后续行计算

U(n+1)(row x)= U(n)(row x-4)\*P(n+1,4)+ U(n)(row x-3)\*P(n+1,3) + U(n)(row x-2)\*P(n+1,2) + U(n)(row x-1)\*P(n+1,1)+ U(n)(row x)\*P(n+1,0);

}

//第三层循环结束

p\_judge=(1,…,1)\*U(n+1)(column 25);

player\_num=n+1;

n=player\_num mod 9;

//同余

**}**

//p\_judge<=0.999时第二层循环结束，记录此时U(n+1）的最后列

U(0)(col 1)=U(n+1)(col 25);

U(0)(col i)=(0…0)T, if i!=1;

//更新U（0）,除第一列其他都为0

}

//第一层循环结束alpha=（0,1,2,…,20）\*U(0)(col1);//对应Q情况计算出的alpha值