**1045: 黑白棋**

时间限制: 1 Sec  内存限制: 128 MB  
提交: 282  解决: 111

**题目描述**

定义一种新的黑白棋：

1. 棋盘大小为5\*5的格子；

2. 有些格子不能放棋子；

3. 同一个格子最多放一个棋子；

4. 先手执白棋，后手执黑棋；

5. 先手第一次可以把棋放在任意可以放的位置上；

6. 接下来两人轮流放棋子，这个棋子必须与上一个人放的棋子相邻

请问：两人都是最优策略，是先手赢，还是先手输？

**输入**

有多组输入数据，第一行为一个数字T，代表有T组输入数据 (0<T≤10)。

接下来为T组数据。

每组数据分5行、每行5个数字构成，每个数字为0或1。0表示这个位置可以放棋子，1表示这个位置不能放棋子。

**输出**

对于每组数据，在一行上输出“win”或“lose”，表示先手赢或输。

**样例输入**

2

11111

11111

11111

11111

00000

11111

11111

11111

11111

10000

**样例输出**

win

lose

**提示**

**来源**

[2013西电ACM校赛现场赛](http://acm.xidian.edu.cn/problemset.php?search=2013%E8%A5%BF%E7%94%B5ACM%E6%A0%A1%E8%B5%9B%E7%8E%B0%E5%9C%BA%E8%B5%9B)

//敌对搜索

#include<iostream>

#include<stdio.h>

#include<algorithm>

#include<cstring>

using namespace std;

const int dx[4]={1,-1,0,0};

const int dy[4]={0,0,1,-1};

int chess[10][10];

int dfs(int x,int y) //上一步走的是x,y //dfs=0表明是必胜态

{

chess[x][y]=1;//我已经走了xy这一步了，然后再向四周看，如果周围有一个必胜态，那我就挂了

//我在这一步下了，那么周围有四个位置对手可以下，看看

for(int i=0;i<4;i++)

{

int xx=x+dx[i];

int yy=y+dy[i];

if(!chess[xx][yy] && dfs(xx,yy)==0)

{//如果他的周围有一个必胜态，那么如果我在这里下了，

//对手就会在周围的必胜态下，那么我就输了

//所以周围不能有必胜态

chess[x][y]=0;

return 1;

}

}

chess[x][y]=0;//还原现场

return 0;//没有找到必胜态，说明是必败态，我们要找的就是必败态

}

int main()

{

// freopen("input.txt","r",stdin);

int T;

cin>>T;

for(int i=0;i<=6;i++)

chess[i][0]=chess[i][6]=chess[0][i]=chess[6][i]=1;

while(T--)

{

for(int i=1;i<=5;i++)

for(int j=1;j<=5;j++)

scanf("%1d",&chess[i][j]);

int flag(0);

for(int i=1;i<=5;i++)

for(int j=1;j<5;j++)

if(!chess[i][j] && !dfs(i,j))

{

flag=1;

break;

}

//找每个能下的棋子，看是先手胜还是后手胜

//只要有一个位置是先手胜，那么先手就行

if(flag)puts("win");//flag=1是先手必胜态

else puts("lose");

}

//即我要找到必败态，从而让对方失败

return 0;

}