**ai第一次实验 (dfs+减枝)**

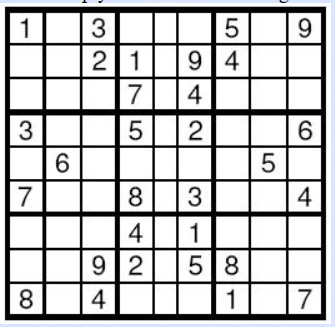
这是poj上的一道直接dfs+剪枝就能够完成的题目。(poj 2676)

**题目:**

Sudoku is a very simple task. A square table with 9 rows and 9 columns is divided to 9 smaller squares 3x3 as shown on the Figure. In some of the cells are written decimal digits from 1 to 9. The other cells are empty. The goal is to fill the empty cells with decimal digits from 1 to 9, one digit per cell, in such way that in each row, in each column and in each marked 3x3 subsquare, all the digits from 1 to 9 to appear. Write a program to solve a given Sudoku-task.

题目大意：

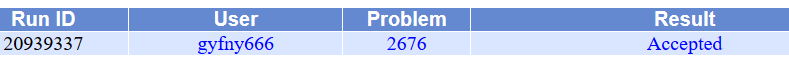
其实就是一道9\*9数独问题，在下图这种方格中，有的地方的数字已经给了，然后填入1—9，使得每行每列，每一个小方格没有重复的数字。



**解题思路：**

这个就是直接一个简单的暴力dfs，每一行每一列地去试一试，然后不满足不重复的条件就减去枝。

**题目提交通过截图：**



**ac code:**

#include<iostream>

#include<algorithm>

#include<cstdio>

#include<cstring>

#include<set>

using namespace std;

const int MAXN=20;

int a[MAXN][MAXN];

set<int> line[MAXN];//行

set<int> row[MAXN];//列

set<int> square[5][5];

int times[5][5];

bool dfs(int x,int y)

{

if(x==0)

return 1;

if(a[x][y]!=0){

if(y==1){

if(dfs(x-1,9))

return 1;

}

else{

if(dfs(x,y-1))

return 1;

}

}

for(int i=1;i<=9;i++){

if(line[x].count(i)==0&&row[y].count(i)==0&&square[(x+2)/3][(y+2)/3].count(i)==0){

line[x].insert(i);

row[y].insert(i);

square[(x+2)/3][(y+2)/3].insert(i);

if(y==1){

if(dfs(x-1,9)){

a[x][y]=i;

return 1;

}

}

else{

if(dfs(x,y-1)){

a[x][y]=i;

return 1;

}

}

line[x].erase(i);

row[y].erase(i);

square[(x+2)/3][(y+2)/3].erase(i);

}

}

return 0;

}

int main()

{

char ch;

int q;

scanf("%d",&q);

getchar();

while(q--)

{

for(int i=1;i<=9;i++){

line[i].clear();

row[i].clear();

}

for(int i=1;i<=3;i++)

for(int j=1;j<=3;j++){

square[i][j].clear();

// times[i]=0;

}

for(int i=1;i<=9;i++){

for(int j=1;j<=9;j++){

scanf("%c",&ch);

a[i][j]=int(ch-'0');

square[(i+2)/3][(j+2)/3].insert(a[i][j]);

line[i].insert(a[i][j]);

row[j].insert(a[i][j]);

if(a[i][j]!=0)

times[(i+2)/3][(j+2)/3]++;//用time记录每一个格子已经被填的个数

}

getchar();

}

dfs(9,9);

for(int i=1;i<=9;i++){

for(int j=1;j<=9;j++){

printf("%d",a[i][j]);

}

printf("\n");

}

}

return 0;

}