4皇后问题的解

**状态空间**：

将每个可行解表示为一个4维向量<x1,x2,x3,x4>,其中xi表示第i行放置皇后的位置（列号），所有可行解为4维向量构成的集合{<x1,x2,x3,x4> | 1<=xi<=4,1<=i<=4},将它排列为完全四叉树。

**操作**：

如果X[i] = X[k]，则第i行和第k行皇后在同一列上

如果第i行的皇后在第j列，第k行的皇后在第t列，即X[i] = j 和 X[k] = t，则只要i - j = k- t 或者 i + j = k+ t，说明两个皇后在对角线上，对两个等式进行变换后，得到结论，只要|i- k| = |j - t|（即|i- k| = |x[i] - x[k]|），则皇后在同一对角线上.

从根节点出发，一直往下，看看这个方向行不行，不行就返回，剪枝，行就继续深度搜索直到可以无冲突的摆放四个皇后。

#include<stdio.h>

#include<math.h>

int x[9]={0};

bool PLACE(int k)//检测第k个皇后能否放进棋盘

{

int i=1;

while(i<k)

{

if(x[i]==x[k]||fabs(x[i]-x[k])==fabs(i-k))

return false;

i++;

}

return true;

}

void NQUEENS(int n)

{

int i,k=1; //k为当前行号

x[1]=0;//x[k]为第k行皇后所放的列号

while(k>0)

{

x[k]++;

while(x[k]<=n&&!PLACE(k))//该行不符合，则放入下一行

x[k]++;

if(x[k]<=n)

{

if(k==n)//输出x[]

{

for(i=1;i<=n;i++)

printf("x[%d]:%d ",i,x[i]);

printf("\n");

}

else//判断下一行

{

k++; x[k]=0;

}

}

else k--;//没找到，则回溯

}

return ;

}

int main()

{

NQUEENS(4);

return 0;

}