**八皇后算法**

问题描述：在一个8\*8的棋盘上放置八个皇后 要求每一个皇后都不能吃掉对方（不能在同一行，同一列，同一斜线上。）

采用回溯法（按照选优条件进行搜索，以达到目标。当探索到下一步时，发现原先选择并不最优或达不到目标，就退回一步重新选择）

实验代码如下：

#include <iostream>

using namespace std;

int c[20],n=8,number=0;

void print()

{

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

{for(int j=0;j<n;j++)

{

if(j==c[i]) cout<<"1 ";

else cout<<"0 ";

}

cout<<endl;}

cout<<endl;

}

void search(int r){

if(r==n){

++number;

cout<<number<<endl;

print();

return ;

}

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

{

c[r]=i;

int ok=1;

for (int j=0;j<r;j++)

if(c[r]==c[j]||r-j==c[r]-c[j]||r-j==c[j]-c[r])

{

ok=0;

break;

}

if(ok) search(r+1);

}

}

int main()

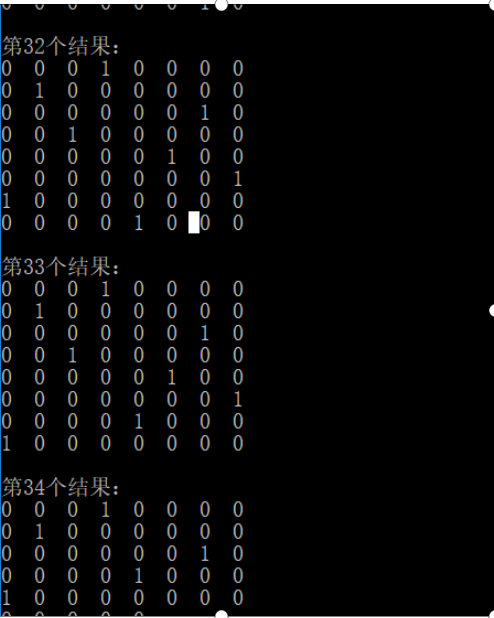
{

search (0);

cout<<number<<endl;

return 0;

}



代码分析：运用一位数组存储，c[0]=1表示第0行的皇后在第一个位子（拥有第0行或第0列的说法）