分析：用分治法，加判断。

n=m时，分成4部分

n>m时，分成上下两部分

n<m时，分成左右两部分

代码：

#include <iostream>

#include <cstdio>

using namespace std;

int \*\*shu;

void tian3(int s\_row, int s\_col, int n,int m)

{

int i,j;

int sr,sc,cha=2\*4\*(1<<(n+m-2));

sr=s\_row;

sc=s\_col+(1<<m);

int k=sc;

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

{

for(j=1,sc=k;j<(1<<(m+1));j+=2,sc++)

{

shu[i][sc] = shu[i][sc-j]+cha;

}

}

}

void tian2(int s\_row, int s\_col, int n,int m)

{

int i,j;

int sr,sc,cha=4\*(1<<(n+m-2));

sr=s\_row;

sc=s\_col+(1<<m);

int k=sc;

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

{

for(j=1,sc=k;j<(1<<(m+1));j+=2,sc++)

{

shu[i][sc] = shu[i][sc-j]+cha;

}

}

}

void tian(int s\_row, int s\_col, int n,int m, int fx)

{

int i,j;

int sr,sc,cha=fx\*4\*(1<<(n+m-2));

if(fx==1)

{

sr=s\_row+(1<<n);

sc=s\_col;

for(i=1,sr;i<(1<<(n+1));i+=2,sr++)

{

for(j=sc;j<(1<<m);++j)

{

shu[sr][j] = shu[sr-i][j]+cha;

}

}

}

else

{

sr=s\_row;

sc=s\_col+(1<<m);

int k=sc;

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

{

for(j=1,sc=k;j<(1<<(m+1));j+=2,sc++)

{

shu[i][sc] = shu[i][sc-j]+cha;

}

}

}

}

void qiu(int n,int m,int s\_row, int s\_col)

{

if(n==m)

{

if(n>1)

{

qiu(n-1,m-1,s\_row,s\_col);

}

tian(s\_row,s\_col,n,m,1);

tian(s\_row,s\_col,n,m,2);

tian3(s\_row+(1<<n),s\_col,n,m);

}

else if(n>m)

{

qiu(n-1,m,s\_row,s\_col);

tian(s\_row,s\_col,n,m,1);

}

else

{

qiu(n-1,m,s\_row,s\_col);

tian2(s\_row,s\_col,n,m);

}

}

int main()

{

int m,n,i,j;

while(~scanf("%d %d",&n,&m))

{

shu = new int \*[1<<(n+1)];

for(i=0;i<(1<<(n+1));++i)

{

shu[i] = new int[1<<(m+1)];

}

shu[0][0]=0;

shu[1][0]=1;

shu[0][1]=2;

shu[1][1]=3;

if(n==m)

{

if(n>1)

qiu(n-1,m-1,0,0);

}

else if(n>m)

{

qiu(n-1,m,0,0);

tian(0,0,n-1,m,1);

}

else

{

qiu(n,m-1,0,0);

tian2(0,0,n,m-1);

}

for(i=0;i<(1<<n);++i)

{

for(j=0;j<(1<<m)-1;++j)

{

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

}

printf("%d\n",shu[i][j]);

}

for(i=0;i<(1<<(n+1));++i)

delete[] shu[i];

delete[] shu;

}

return 0;

}