**[压缩状态DP](http://www.cppblog.com/qywyh/archive/2006/10/18/13823.html)**

压缩状态DP, 一般用于n\*m矩阵 n << m 或 m << n的情况.  
用每一行或每一列表示状态(关键在于进制表示和状态转移)  
  
pku3020代码链接:<http://www.cppblog.com/qywyh/articles/13822.html>  
  
看了ghost\_wei大牛的代码, 滚动数组, 位运算, 出神入化啊, 而且敲题时间为10分钟(我用了一个下午-\_-) , 无限仰慕啊!:)

 myCode:

http://www.cppblog.com/Images/OutliningIndicators/None.gif#include  < iostream >   
http://www.cppblog.com/Images/OutliningIndicators/None.gifusing   namespace  std;  
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifconst   int  INF  =   1   <<   28 ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  n, r, c;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifint  e[ 11 ]  =   http://www.cppblog.com/Images/dot.gif{ 1 ,  2 ,  4 ,  8 ,  16 ,  32 ,  64 ,  128 ,  256 ,  512 ,  1024 } ;  
http://www.cppblog.com/Images/OutliningIndicators/None.gifchar  m[ 50 ][ 20 ];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  d[ 50 ][ 1024 ];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  b[ 20 ];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  cc[ 20 ];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  ss;  
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  Try( int  x,  int  s)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     if  (x  >=  c)  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         int  k  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif         for  ( int  i = 0 ; i < c; i ++ )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            k  +=  b[i]  \*  e[i];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         if  (d[ 0 ][k]  ==   - 1   ||  d[ 0 ][k]  >  s)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            d[ 0 ][k]  =  s;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         return  ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     if  (m[ 0 ][x]  ==   ' o ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Try(x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    }   else   if  (m[ 0 ][x]  ==   ' \* ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         int  t1  =  b[x], t2  =  b[x + 1 ], t3  =  s;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        b[x]  =   1 ; b[x + 1 ]  =   1 ; s  +=   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Try(x + 2 , s);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        b[x]  =  t1; b[x + 1 ]  =  t2; s  =  t3;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         if  (r  !=   1   ||  m[ 0 ][x]  ==   ' o ' ) Try(x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }     
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  DFS( int  i,  int  x,  int  s)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     if  (x  >=  c)  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         int  k  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif         for  ( int  j = 0 ; j < c; j ++ )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            k  +=  cc[j]  \*  e[j];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         if  (d[i][k]  ==   - 1   ||  d[i][k]  >  ss  +  s)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            d[i][k]  =  ss  +  s;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         return  ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     if  (b[x]  ==   0 )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif         if  (m[i - 1 ][x]  ==   ' \* ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            cc[x]  =   1 ; s  +=   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            DFS(i, x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif        }   else   http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif             if  (m[i][x]  ==   ' \* ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 int  t1  =  cc[x], t2  =  cc[x + 1 ], t3  =  s;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                cc[x]  =   1 ;  cc[x + 1 ]  =   1 ; s  +=   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 if  (b[x + 1 ]  ==   0   &&  m[i - 1 ][x + 1 ]  ==   ' \* ' )  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    s  +=   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                DFS(i, x + 2 , s);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                cc[x]  =  t1; cc[x + 1 ]  =  t2; s  =  t3;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 if  (i  !=  r - 1   ||  m[i][x]  ==   ' o ' ) DFS(i, x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif            }   else   if  (m[i][x]  ==   ' o ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                DFS(i, x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    }   else   http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif             if  (m[i][x]  ==   ' \* ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 int  t1  =  cc[x], t2  =  cc[x + 1 ], t3  =  s;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                cc[x]  =   1 ;  cc[x + 1 ]  =   1 ; s  +=   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 if  (b[x + 1 ]  ==   0   &&  m[i - 1 ][x + 1 ]  ==   ' \* ' )  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    s  +=   1 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                DFS(i, x + 2 , s);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                cc[x]  =  t1; cc[x + 1 ]  =  t2; s  =  t3;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 if  (i  !=  r - 1   ||  m[i][x]  ==   ' o ' ) DFS(i, x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif            }   else   if  (m[i][x]  ==   ' o ' )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                DFS(i, x + 1 , s);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }         
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  init()  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    memset(d[ 0 ],  - 1 ,  sizeof (d[ 0 ]));  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    memset(b,  0 ,  sizeof (b));  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    Try( 0 ,  0 );  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid  Solve()   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  i, j, k;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    init();  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     for  (i = 0 ; i < r - 1 ; i ++ )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        memset(d[i + 1 ],  - 1 ,  sizeof (d[i + 1 ]));  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif         for  (k = 0 ; k < e[c]; k ++ )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif             if  (d[i][k]  !=   - 1 )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                 int  t  =  k, j  =   0 , kk  =   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                memset(b,  0 ,  sizeof (b));  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                memset(cc,  0 ,  sizeof (cc));  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif                 while  (t  !=   0 )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    b[j ++ ]  =  t  %   2 ;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    t  /=   2 ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif                }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                ss  =  d[i][k];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                DFS(i + 1 ,  0 ,  0 );  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif            }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     int  ans  =  INF;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     for  (k = 0 ; k < e[c]; k ++ )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif         if  (d[r - 1 ][k]  !=   - 1   &&  d[r - 1 ][k]  <  ans)  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            ans  =  d[r - 1 ][k];  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }   
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    cout  <<  ans  <<  endl;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint  main()  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    cin  >>  n;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif     while  (n --   !=   0 )  http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        cin  >>  r  >>  c;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif         for  ( int  i = 0 ; i < r; i ++ ) cin  >>  m[i];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        Solve();  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    system( " pause " );  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif     return   0 ;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}   
http://www.cppblog.com/Images/OutliningIndicators/None.gif

ghost\_wei大牛的code,  放出来供大家学习,  用了滚动数组优化, 而且位运算用得出神入化:)

http://www.cppblog.com/Images/OutliningIndicators/None.gif#include<iostream.h>  
http://www.cppblog.com/Images/OutliningIndicators/None.gif#include <fstream.h>  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifconst int k2[11]=http://www.cppblog.com/Images/dot.gif{1,2,4,8,16,32,64,128,256,512,1024};  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint n,m,c[2][1024];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifchar d[40][10];  
http://www.cppblog.com/Images/OutliningIndicators/None.gifinline void min(int &i,int j)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    if (i>j) i=j;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
http://www.cppblog.com/Images/OutliningIndicators/None.gifvoid work()  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int i,j,km,k,e7,e8,l,t,ans;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    km=k2[m];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    for (i=0;i<km;i++) c[0][i]=100000;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    c[0][0]=0;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    e7=0; e8=1;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    for (i=0;i<n;i++)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        for (j=0;j<km;j++) c[e8][j]=100000;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        for (j=1;j<m;j++)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            if (d[i][j]=='\*')  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                for (k=0;k<km;k++)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    min(c[e7][k|k2[j]|k2[j-1]],c[e7][k]+1);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        for (k=0;k<km;k++)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif        http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            l=0; t=0;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            for (j=0;j<m;j++)   
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                if (!(k&k2[j])&&d[i][j]=='\*')  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif                http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    l+=k2[j];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                    t++;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif                }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            min(c[e8][l],c[e7][k]+t);  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif        }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        e7=e7^1; e8=e8^1;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    ans=100000;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    for (k=0;k<km;k++)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        min(ans,c[e7][k]);  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    cout<<ans<<endl;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
http://www.cppblog.com/Images/OutliningIndicators/None.gifint main()  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedBlock.gifhttp://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    int tc,cas,i,j;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    cin>>tc;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    for (cas=1;cas<=tc;cas++)  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockStart.gifhttp://www.cppblog.com/Images/OutliningIndicators/ContractedSubBlock.gif    http://www.cppblog.com/Images/dot.gif{  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        cin>>n>>m;  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        for (i=0;i<n;i++)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif            for(j=0;j<m;j++)  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif                cin>>d[i][j];  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif        work();  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedSubBlockEnd.gif    }  
http://www.cppblog.com/Images/OutliningIndicators/InBlock.gif    return 0;  
http://www.cppblog.com/Images/OutliningIndicators/ExpandedBlockEnd.gif}  
http://www.cppblog.com/Images/OutliningIndicators/None.gif