

## POJ 2411

在  $h \times w$  的矩阵中铺  $1 \times 2$  或  $2 \times 1$  的方块，要铺满，问方案数。

题解：看到状压tag了。我的状压是用1表示这一块是  $2 \times 1$  的上面那块，需要向下延伸，0表示其他。但是这个样子其实是有一点容易TLE<sup>1)</sup>做了一下读入 `if(!f[i-1][k])continue` 和 `if1)`

1)

`h*w)%2)puts("0")` 的优化卡过去了。看到很多人用记忆化搜索感觉那个样子可能会好一些

```
#include<iostream>
#include<cstdio>
#include<cstring>
#include<cmath>
#include<cstdlib>
using namespace std;
int h,w;
long long f[12][(1<<12)];
int read()
{
    int x=0,f=1;char c=getchar();
    while(c<'0' || c>'9'){if(c=='-')f=-1;c=getchar();}
    while(c>='0' && c<='9'){x=x*10+c-'0';c=getchar();}
    return x*f;
}
int main()
{
    h=read(),w=read();
    while(h&&w)
    {
        if((h*w)%2){puts("0"),h=read(),w=read();continue;}
        memset(f,0,sizeof(f));
        if(!h&&!w)break;
        for(int i=1;i<=h;i++)
        {
            for(int j=0;j<(1<<w);j++)
            {
                if(i==1)
                {
                    f[i][j]=1;
                    for(int l=1;l<(1<<w);l<=1)
                    {
                        if((j&l)==0)
                        {
                            if((l<<1)>=(1<<w)){f[i][j]=0;break;}
                            else if(j&(l<<1)){f[i][j]=0;break;}
                            else l<=1;
                        }
                    }
                }
            }
        }
        else
        {
```

