

上海理工校赛题解报告

小结

数学题，不会

B题 BHEITH I NGRAL E

当初挣扎了一下，然后不会求给定 i 行 j 列的格子有多少单调曲线，就放弃了。

分析

- 题目的核心是求给定 i 行 j 列的格子能够构造多少单调曲线。可以用dp或者组合求解，不喜欢数学的我毅然投奔了dp。
- 不难发现，当给定了宽为 i ，高为 j 的格子矩阵的时候，我们分析右上角那块，如果这块不画，那么所有情况就会变成 $dp[i][j-1]$ ，如果这块画上，那么情况就会是 $dp[i-1][j]$ 。则得到状态转移方程 $dp[i][j]=dp[i][j-1]+dp[i-1][j]$ 。当然，边界条件还是值得考虑一下的。
- 算出dp数组后，性高彩烈的用 n^3 去套老鹅。其实我们确定了山顶左边位置 l 之后不用去确定右边的 r ， $dp[n-l][h]$ 就是右边包含山顶在内的全部可能。所以 $ans=ans+dp[l-1][h-1]*dp[n-l][h]$ 即可。

代码

```
#include<iostream>
using namespace std;

long long dp[2003][2003];
long long mod=1e9+7;
long long ans;
int main(){
    int n,m;
    cin >>n>>m;
    // 初始化高度为1时
    dp[1][1]=2;
    for(int i = 2;i<=n;++i){
        dp[i][1]=dp[i-1][1]+1;
    }
    // 初始化宽度为1的时候
    for(int j = 2;j<=m;++j){
        dp[1][j]=dp[1][j-1]+1;
    }
    // 求dp, i为宽，j为高
```

```

for(int i = 2;i<=n;++i){
    for(int j = 2;j<=m;++j){
        dp[i][j]=(dp[i][j-1]+dp[i-1][j])%mod;
    }
}

// 宽度为0时答案应该为1，因为下面用的乘法
for(int j=0;j<=m;++j){
    dp[0][j]=1;
}

// 高度为0时，也应该是1，即取0
for(int i = 0;i<=n;++i){
    dp[i][0]=1;
}

// cout<<dp[1][1]<<endl<<dp[2][1]<<endl<<dp[1][2]<<endl<<dp[2][2]<<endl<<dp[1][3]
// <<endl<<dp[2][3]<<endl;

// 枚举山顶的情况,l是左边，r是右边，h是山高
for(int l = 1;l<=n;++l){
    for(int h = 1;h<=m;++h){
        ans = (ans+(dp[l-1][h-1]*dp[n-l][h])%mod)%mod; // 是左边的情况乘以右边
的情况
    }
}

// 加上山顶全为0的情况，只有一种
ans=(ans+1)%mod;

cout<<ans<<endl;

}

```