

# Single Problem Set (1)

---

## 1.方格取数

- **Date:** 2020/5/12
- **Tag:** DP
- **Source:** <https://www.luogu.com.cn/problem/P1004>
- **Problem:** 设有  $N \times N$  的方格图 ( $N \leq 9$ ) 我们将其中的某些方格中填入正整数，而其他的方格中则放入数字 0。某人从图的左上角的 A 点出发，可以向下行走，也可以向右走，直到到达右下角的 B 点。在走过的路上，他可以取走方格中的数（取走后的方格中将变为数字 0）。此人从 A 点到 B 点共走两次，试找出 2 条这样的路径，使得取得的数之和为最大。
- **Solution:** 不妨两次同时走，则两次总步数是相同的，只需要用  $ans[step][x1][x2]$  存储“步数为  $step$  第一、二次行动的横坐标分别为  $x1$   $x2$ ”时的最优解，递归即可。
- <hidden Code>

```
#include<bits/stdc++.h>
```

```
int main(){  
}
```

</hidden>

---


## 2.传纸条

- **Date:** 2020/5/14
- **Tag:** DP
- **Source:** <https://www.luogu.com.cn/problem/P1006>
- **Problem:** 设有  $M \times N$  的方格图 ( $M, N \leq 50$ ) 所有方格填入非负整数；其中，保证左上右下为 0。从左上到右下选择两条路径，每条必须满足要么向右要么向下，且两条不能有交叉。求路径数字之和最大值。
- **Solution:** 解类似 1，不过在递归时，取消了  $x1-1 == x2-1$  一项。另外，需要注意的是，多维数组不要忘了哪个分量为行或者列。

Last  
update:  
2020/05/17 20:18 2020-2021:teams:bigbros:jerydeak:sps1 <https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:bigbros:jerydeak:sps1&rev=1589717883>  
20:18

---

From:  
<https://wiki.cvbbacm.com/> - **CVBB ACM Team**

Permanent link:  
<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:bigbros:jerydeak:sps1&rev=1589717883> 

Last update: **2020/05/17 20:18**