

# P3397 | 二维差分

Wednesday, March 26, 2025 7:29 PM

## 题目描述

[复制 Markdown](#) [展开](#) [进入 IDE 模式](#)

在  $n \times n$  的格子地上有  $m$  个地毯。

给出这些地毯的信息，问每个点被多少个地毯覆盖。

## 输入格式

第一行，两个正整数  $n, m$ 。意义如题所述。

接下来  $m$  行，每行两个坐标  $(x_1, y_1)$  和  $(x_2, y_2)$ ，代表一块地毯，左上角是  $(x_1, y_1)$ ，右下角是  $(x_2, y_2)$ 。

## 输出格式

输出  $n$  行，每行  $n$  个正整数。

第  $i$  行第  $j$  列的正整数表示  $(i, j)$  这个格子被多少个地毯覆盖。

## 输入输出样例

输入 #1

[复制](#)

输出 #1

[复制](#)

```
5 3
2 2 3 3
3 3 5 5
1 2 1 4
```

```
0 1 1 1 0
0 1 1 0 0
0 1 2 1 1
0 0 1 1 1
0 0 1 1 1
```

Ac code



P3397\_CPP

```
1 // Two-dimensional difference array
2 #include<bits/stdc++.h>
3
4 using namespace std;
5
6 int n, m;
7 int diff[1005][1005];
8 int sum[1005][1005];
9
10 int main()
11 {
12     scanf("%d%d", &n, &m);
13     while (m--)
14     {
15         int x1, y1, x2, y2;
16         scanf("%d%d%d%d", &x1, &y1, &x2, &y2);
17         diff[x1][y1]++;
18         diff[x1][y2 + 1]--;
19         diff[x2 + 1][y1]--;
20         diff[x2 + 1][y2 + 1]++;
21     }
22     for (int i = 1; i <= n; i++)
23     {
24         for (int j = 1; j <= n; j++)
25         {
26             sum[i][j] = sum[i - 1][j] + sum[i][j - 1] - sum[i - 1][j - 1] + diff[i][j];
27             printf("%d ", sum[i][j]);
28         }
29         printf("\n");
30     }
31     return 0;
```

] 区间 (矩形) 修改差分!

] 差分 前缀和的方式!

```

28     }
29     printf("\n");
30 }
31 return 0;
32 }

```

附：二维前缀和

```

5  int m, n;
6  int grid[1005][1005];
7  int sum[1005][1005];
8
9  int query(int x1, int y1, int x2, int y2)
10 {
11     return sum[x2][y2] - sum[x1 - 1][y2] - sum[x2][y1 - 1] + sum[x1 - 1][y1 - 1];
12 }
13
14 int main()
15 {
16     cin >> n >> m;
17     for (int i = 1; i <= n; i++)
18     {
19         for (int j = 1; j <= m; j++)
20         {
21             cin >> grid[i][j];
22         }
23     }
24
25     for (int i = 1; i <= n; i++)
26     {
27         for (int j = 1; j <= m; j++)
28         {
29             sum[i][j] = grid[i][j] + sum[i - 1][j] + sum[i][j - 1] - sum[i - 1][j - 1];
30         }
31     }
32 }

```