

# AcWing 796. 子矩阵的和 【c++详细题解】

原题链接 (<https://www.acwing.com/problem/content/description/798/>)

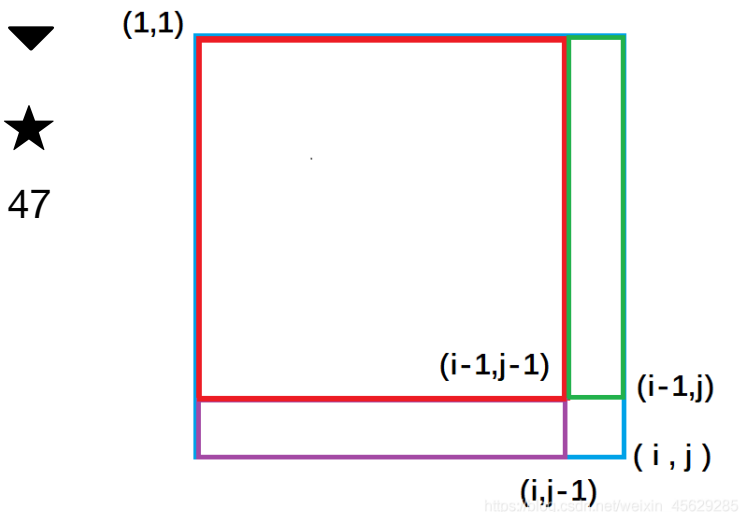
简单

作者: 林小鹿 (/user/myspace/index/29688/), 2020-12-19 20:58:37, 所有人可见, 阅读 4955

## 二维前缀和推导

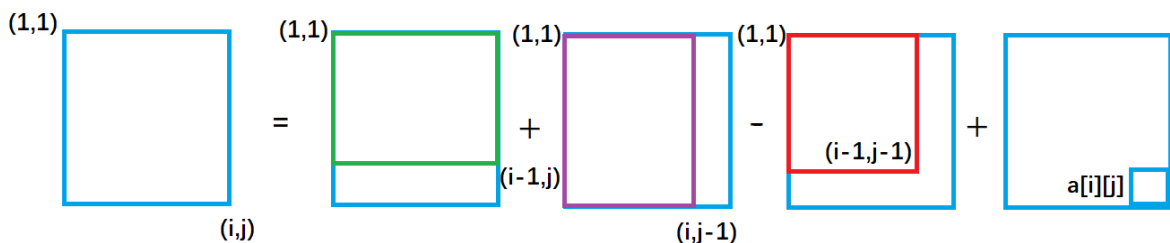
76

如图:



★

47

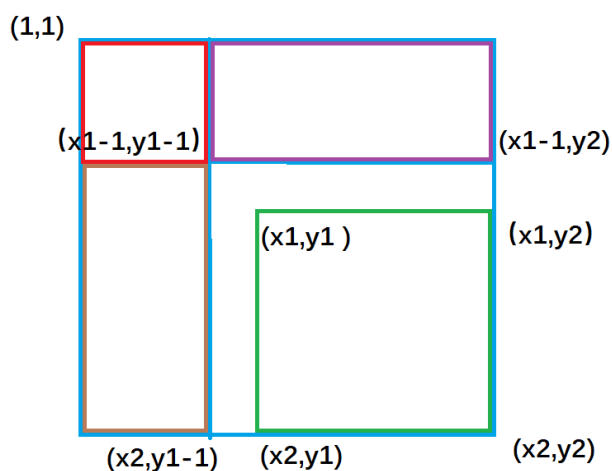


green ↑ purple ↑ small box ↑ red ↑

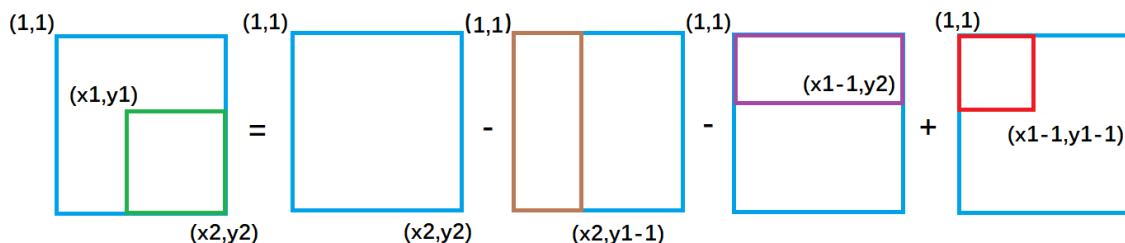
$$s[i][j] = s[i-1][j] + s[i][j-1] + a[i][j] - s[i-1][j-1]$$



11:00  
2022-11-



[https://blog.csdn.net/weixin\\_45629285](https://blog.csdn.net/weixin_45629285)



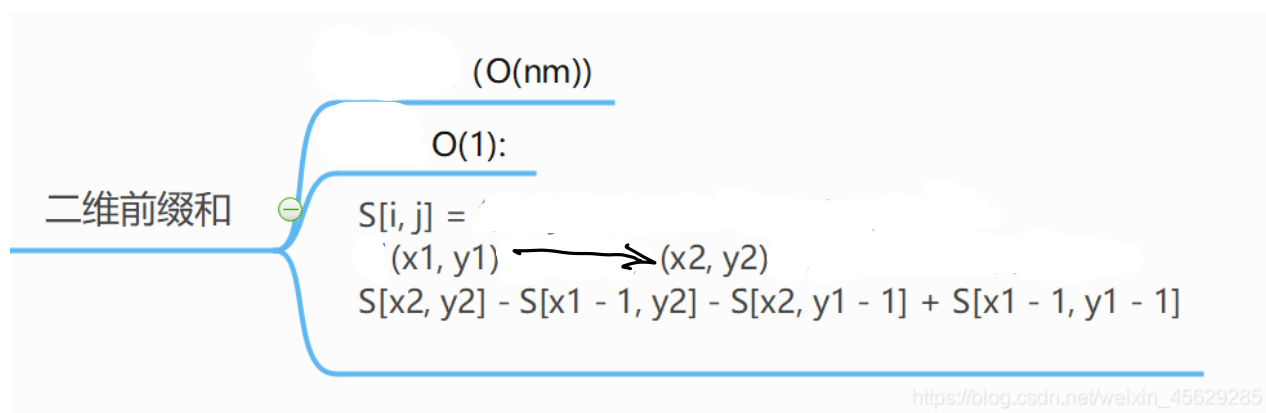
[https://blog.csdn.net/weixin\\_45629285](https://blog.csdn.net/weixin_45629285)

$$\text{green} = \text{Blue} - \text{Brown} - \text{Purple} + \text{red}$$

$$x_1, y_1 \rightarrow x_2, y_2 = S(x_2, y_2) - S(x_2, y_1 - 1) - S(x_1 - 1, y_2)$$

总结:

$$+ S(x_1 - 1, y_1 - 1)$$



## 前缀和与差分的个人心得总结

([https://blog.csdn.net/weixin\\_45629285/article/details/111146240](https://blog.csdn.net/weixin_45629285/article/details/111146240))

代码:

```
#include <iostream>

using namespace std;

const int N = 1010;

int n, m, q;
int s[N][N];

int main()
{
    scanf("%d%d%d", &n, &m, &q);

    for (int i = 1; i <= n; i ++ )
        for (int j = 1; j <= m; j ++ )
            scanf("%d", &s[i][j]);

    for (int i = 1; i <= n; i ++ )
        for (int j = 1; j <= m; j ++ )
            s[i][j] += s[i - 1][j] + s[i][j - 1] - s[i - 1][j - 1];

    while (q -- )
    {
        int x1, y1, x2, y2;
        scanf("%d%d%d%d", &x1, &y1, &x2, &y2);
        printf("%d\n", s[x2][y2] - s[x1 - 1][y2] - s[x2][y1 - 1] + s[x1 - 1][y1 - 1]);
    }

    return 0;
}
```