

前綴和 (0base)

一维:

```
std::vector a(n, 0), s(n + 1, 0LL);
for (int i = 0; i < n; i++) {
    s[i + 1] = s[i] + a[i];
}
```

二维:

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

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

    std::vector a(n, std::vector<int>(m));

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            scanf("%d", &a[i][j]);
        }
    }

    std::vector s(n + 1, std::vector<int>(m + 1));
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            s[i + 1][j + 1] = s[i][j + 1] + s[i + 1][j] - s[i][j] + a[i][j];
        }
    }

    while (q--) {
        int l1, r1, l2, r2;
        scanf("%d%d%d%d", &l1, &r1, &l2, &r2);
        l1--, l2--, r1--, r2--;
        printf("%d\n", s[l2 + 1][r2 + 1] - s[l1][r2 + 1] - s[l2 + 1][r1] + s[l1][r1]);
    }

    return 0;
}
```

差分(0base)

一维

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

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

    std::vector<int> a(n);

    for (int i = 0; i < n; i++) {
        scanf("%d", &a[i]);
    }

    std::vector<int> diff(n + 1);
    for (int i = 0; i < n; i++) {
        diff[i] = a[i] - (i - 1 >= 0 ? a[i - 1] : 0);
    }

    while (m--) {
        int l, r, v;
        scanf("%d%d%d", &l, &r, &v);
        l--;
        diff[l] += v;
        diff[r] -= v;
    }

    auto ans = diff;
    for (int i = 0; i < n; i++) {
        ans[i + 1] += ans[i];
    }

    for (int i = 0; i < n; i++) {
        printf("%d%c", ans[i], " \n"[i == n - 1]);
    }

    return 0;
}
```

二维

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

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

    std::vector a(n, std::vector<int>(m));
    for (int i = 0; i < n; i++) {
```

```

        for (int j = 0; j < m; j++) {
            scanf("%d", &a[i][j]);
        }
    }

    std::vector diff(n + 1, std::vector<int>(m + 1));
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            diff[i][j] = a[i][j];
            if (i > 0) {
                diff[i][j] -= a[i - 1][j];
            }
            if (j > 0) {
                diff[i][j] -= a[i][j - 1];
            }
            if (i > 0 && j > 0) {
                diff[i][j] += a[i - 1][j - 1];
            }
        }
    }

    while (q--) {
        int l1, r1, l2, r2, v;
        scanf("%d%d%d%d", &l1, &r1, &l2, &r2, &v);
        l1--, r1--, l2--, r2--;
        diff[l1][r1] += v;
        diff[l1][r2 + 1] -= v;
        diff[l2 + 1][r1] -= v;
        diff[l2 + 1][r2 + 1] += v;
    }

    std::vector ans(n + 1, std::vector<int>(m + 1));
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            ans[i + 1][j + 1] = ans[i + 1][j] + ans[i][j + 1] - ans[i][j] + diff[i][j];
        }
    }

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= m; j++) {
            printf("%d%c", ans[i][j], " \n"[j == m]);
        }
    }

    return 0;
}

```

