

树状数组

单点修改，区间查询

```
#include <bits/stdc++.h>
using namespace std;

typedef long long ll;
const int N = 2e5 + 5;
ll tr[N];
int n, q;
ll lowbit(ll x){return x&(-x);}
void add(ll x, ll k){
    for(int i = x; i <= n; i += lowbit(i)) tr[i] += k;
}
ll sum(ll x){
    ll res = 0;
    for(int i = x; i; i -= lowbit(i)) res += tr[i];
    return res;
}

int main(){
    cin >> n >> q;
    string s;
    for(int i = 0; i < q; i++){
        ll a, b;
        cin >> s >> a >> b;
        if(s[0] == 'x'){
            add(a, b);
        }
        else{
            cout << sum(b) - sum(a - 1) << '\n';
        }
    }

    return 0;
}
```

区间修改，单点查询

```
#include <bits/stdc++.h>
using namespace std;

typedef long long ll;
const int N = 5e5 + 5;
ll tr[N], a[N];
int n, q;
ll lowbit(ll x){return x&(-x);}
void add(ll x, ll k){
    for(int i = x; i <= n; i += lowbit(i)) tr[i] += k;
}
```

```

ll sum(ll x){
    ll res = 0;
    for(int i = x; i; i -= lowbit(i)) res += tr[i];
    return res;
}

int main(){
    cin >> n >> q;
    for(int i = 1; i <= n; i++) cin >> a[i];
    for(int i = 0; i < q; i++){
        int op, l, r;
        ll k;
        cin >> op;
        if(op == 1){
            cin >> l >> r >> k;
            add(l, k);
            add(r + 1, -k);
        }
        else{
            cin >> l;
            cout << a[l] + sum(l) << '\n';
        }
    }

    return 0;
}

```