

48 FenwickTree (BIT) sum query on 1D-Array (range update-query):

```

49
50 //answer range update and range query
51 struct FenwickTreeSum{
52     int n;
53     vector<ll> M , A;
54
55     FenwickTreeSum(int n){
56         this->n = n;
57         M.assign(n , 0);
58         A.assign(n , 0);
59     }
60
61     FenwickTreeSum(vector<ll> &a) : FenwickTreeSum(a.size()) {
62         for (int i = 0; i < a.size(); i++){
63             update(i , i , a[i]);
64         }
65     }
66
67     void upd(int i, ll mul, ll add){
68         while (i < n){
69             M[i] += mul;
70             A[i] += add;
71             i |= (i + 1);
72         }
73     }
74
75     void update(int l, int r, ll x){
76         upd(l, x, -x * (l - 1));
77         upd(r, -x, x * r);
78     }
79
80     ll query(int i){
81         ll mul = 0, add = 0;
82         ll st = i;
83         while (i ≥ 0) {
84             mul += M[i];
85             add += A[i];
86             i = (i & (i + 1)) - 1;
87         }
88         return (mul * st + add);
89     }
90
91     ll query(int l, int r){
92         return query(r) - query(l - 1);
93     }
94 };

```