

树状数组：单点修改 + 区间查询

线段树一般只能单点修改、区间查询！

单点修改了之后，如何回馈给上层？

```
struct BIT
{
    long long n;
    std::vector<long long> a;

    BIT(long long len) : n(len), a(len + 1) {}
    BIT(){}

    long long lowbit(long long x)
        { return x & -x; }

    // 单点增量 (核心)
    void add(long long id, long long x)
    {
        while (id <= n)
        {
            a[id] += x;
            id += lowbit(id);
        }
    }

    // 前缀和 (核心)
    long long sum(long long id)
    {
        long long ans = 0;
        while (id > 0)
        {
            ans += a[id];
            id -= lowbit(id);
        }
        return ans;
    }

    // 单点修改
    void modify(long long id, long long val)
    {
        long long x = val - (sum(id) - sum(id - 1));
        add(id, x);
    }

    // 区间查询
    long long query(long long left, long long right)
    {
        return sum(right) - sum(left - 1);
    }
}
```

```
};  
}
```