

校门外的树

某校大门外长度为 L 的马路上有一排树，每两棵相邻的树之间的间隔都是1米。我们可以把马路看成一个数轴，马路的一端在数轴0的位置，另一端在 L 的位置；数轴上的每个整数点，即0, 1, 2, ..., L ，都种有一棵树。

由于马路上有一些区域要用来建地铁。这些区域用它们在数轴上的起始点和终止点表示。已知任一区域的起始点和终止点的坐标都是整数，区域之间可能有重合的部分。现在要把这些区域中的树（包括区域端点处的两棵树）移走。你的任务是计算将这些树都移走后，马路上还有多少棵树。

第一行有两个整数： L ($1 \leq L \leq 10000$) 和 M ($1 \leq M \leq 100000$)， L 代表马路的长度， M 代表区域的数目， L 和 M 之间用一个空格隔开。接下来的 M 行每行包含两个不同的整数，用一个空格隔开，表示一个区域的起始点和终止点的坐标。

链接: <https://ac.nowcoder.com/acm/contest/20960/1017>

校门外的树(改版)

某校大门外长度为L的马路上有一排树，每两棵相邻的树之间的间隔都是1米。我们可以把马路看成一个数轴，马路的一端在数轴0的位置，另一端在L的位置；数轴上的每个整数点，即0, 1, 2, …, L, 都种有一棵树。

由于马路上有一些区域要用来建地铁。这些区域用它们在数轴上的起始点和终止点表示。已知任一区域的起始点和终止点的坐标都是整数，区域之间可能有重合的部分。现在有三种操作：

- . 1 x y 将x点到y点范围内的树全部移除
- . 2 x y 将x点到y点范围内的树全部种上
- . 3 查询马路上还有多少棵树.

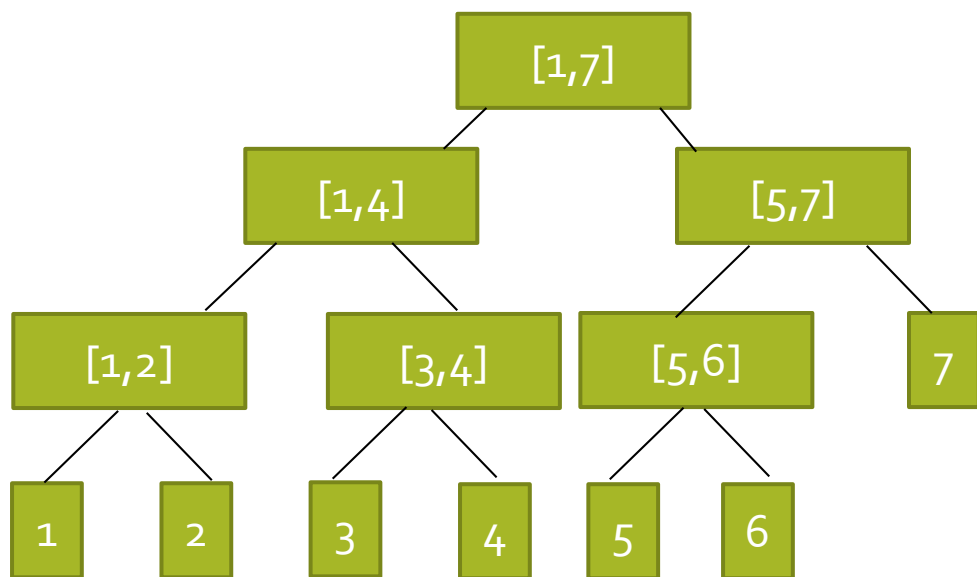
第一行有两个整数：L ($1 \leq L \leq 10000$) 和 M ($1 \leq M \leq 100000$)，L代表马路的长度，M代表区域的数目，L和M之间用一个空格隔开。接下来的M行每行包含两个不同的整数，用一个空格隔开，表示一个区域的起始点和终止点的坐标。

动态查询

线段树

李卓锦

线段树的形式:



[5,5]
[3,4]
[1,5]
[1,7]

在进行区间操作和统计时把区间等价转换成若干个子区间($\log(n)$ 个)的相同操作

任意的线段 $[a,b]$ 在线段树的查询或查找过程中把这个线段最多分成 $\log(b-a)$ 份 (每一层最多两个区间)

线段树除建树操作外都是 $\log(n)$ 级别的复杂度.

G1:

给定两个数字,n,m

第2行给定n个数字

第3行到第3+m-1行给定两个数字,a,b,询问a到b段的数字之和.

怎么用线段树来求这个问题呢?每一段区间该存储什么信息?怎么存储?

1. 建树

```
void build(int pos, int l, int r) {  
    if (l == r) {  
        tree[pos] = a[l];  
        return;  
    }  
    int mid = (l + r) >> 1;  
    build(2 * pos, l, mid);  
    build(2 * pos + 1, mid + 1, r);  
    tree[pos] = tree[2 * pos] + tree[2 * pos + 1];  
}
```

2. 查询

```
void query(int pos, int l, int r, int x, int y) {  
    if (x <= l && y >= r) {  
        return tree[pos];  
    }  
    int mid = (l + r) >> 1;  
    if (y <= mid) return query(2 * pos, l, mid, x, y);  
    else if (x >= mid + 1) return query(2 * pos + 1, mid + 1, r, x, y);  
    else {  
        return query(2 * pos, l, mid, x, y) + query(2 * pos + 1, mid + 1, r, x, y);  
    }  
}
```

```
def buildtree(self, k: int, l: int, r: int) -> None:
```

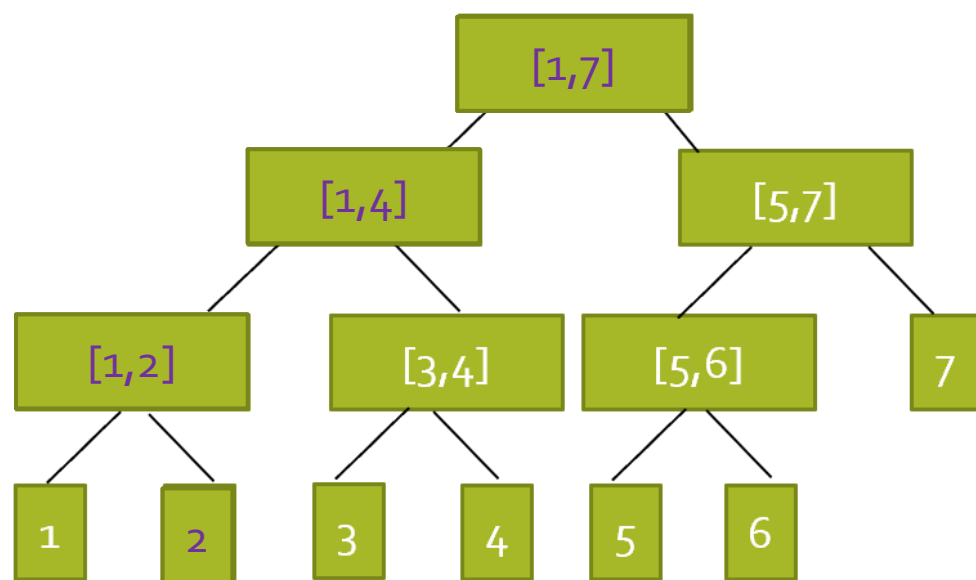
```
    if l == r:  
        self.f[k] = self.arr[l]  
        return  
    m = (l + r) >> 1  
    self.buildtree(k + k, l, m)  
    self.buildtree(k + k + 1, m + 1, r)  
    self.f[k] = self.f[k + k] + self.f[k + k + 1]
```

```
    p += self.v[k]  
    if l == s and r == t:  
        return p * (r - l + 1) + self.f[k]  
    m = (l + r) >> 1  
    if t <= m:  
        return self.calc(k + k, l, m, s, t, p)  
    elif s > m:  
        return self.calc(k + k + 1, m + 1, r, s, t, p)  
    else:  
        return self.calc(k + k, l, m, s, m, p) + self.calc(  
            k + k + 1, m + 1, r, m + 1, t, p)
```

1.点修改—区间查询

2.区间修改—区间查询

以[2,2]为例



```
void change(int pos, int l, int r, int x, int val) {
    if (l==r) {
        tree[pos]=val;
        return;
    }
    int mid=(l+r)>>1;
    if (x<=mid) change(2*pos, l, mid, x, val);
    else if (x>=mid+1) change(2*pos+1, mid+1, r, x, val);
    pushup();
}
```

```
void pushup(int pos, int l, int r){
    tree[pos]=tree[2*pos]?tree[2*pos+1];
}
```


G2:

Can you answer these queries III

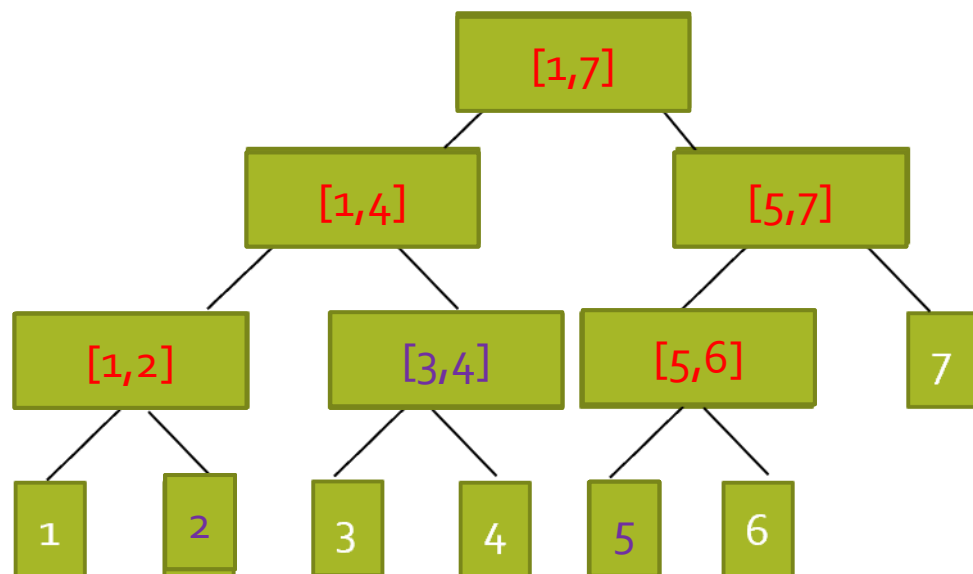
n 个数, q 次操作

操作 $o\ x\ y$ 把 A_x 修改为 y

操作 $1\ l\ r$ 询问区间 $[l,r]$ 的最大子段和

[GSS3 - Can you answer these queries III - 洛谷 | 计算机科学教育新生态 \(luogu.com.cn\)](https://www.luogu.com.cn/problem/GSS3)

以[2,5]为例



第一行包含两个整数 N, M , 分别表示数列中实数的个数和操作的个数

第二行包含 N 个实数, 其中第 i 个实数表示数列的第 i 项

接下来 M 行, 每行为一条操作, 格式为以下三种之一:

操作1: $1\ x\ y\ k$ 表示将第 x 到第 y 项每项加上 k , k 为一实数

操作2: $2\ x\ y$ 表示求出第 x 到第 y 项这一子数列的平均数

操作3: $3\ x\ y$ 表示求出第 x 到第 y 项这一子数列的方差

$N \leq 1e5, M \leq 1e5$

[P1471 方差 - 洛谷 | 计算机科学教育新生态 \(luogu.com.cn\)](https://www.luogu.com.cn/problem/P1471)

难点在于方差怎么求

$$\frac{1}{n} \sum_{i=1}^n (A_i - A)^2 = \frac{1}{n} \sum_{i=1}^n (A_i^2 - 2 * A * A_i + A^2)$$

维护两个线段树,区间和以及区间平方和

在遇到特殊的线段树题时,例如
求区间 $\sin()$ 和,区间 $\cos()$ 和等等.
通过泰勒公式展开,展开成多项式. 展开几层视精度而定.

给你 n 个数,然后 q 次查询, $t=0$ 的时候把区间都开根取整, $t=1$ 的时候就输出区间的和

https://atcoder.jp/contests/abc265/tasks/abc265_g

有几个古希腊书籍中包含了对传说中的亚特兰蒂斯岛的描述。其中一些甚至包括岛屿部分地图。但不幸的是，这些地图描述了亚特兰蒂斯的不同区域。您的朋友 **Bill** 必须知道地图的总面积。你自告奋勇写了一个计算这个总面积的程序

输入包含多组测试用例。对于每组测试用例，第一行包含整数 n ，表示总的地图数量。接下来 n 行，描绘了每张地图，每行包输入包含多组测试用例。

对于每组测试用例，第一行包含整数 n ，表示总的地图数量。

接下来 n 行，描绘了每张地图，每行包含四个数字 $x1,y1,x2,y2$ （不一定是整数）， $(x1,y1)$ 和 $(x2,y2)$ 分别是地图的左上角位置和右下角位置。

注意，坐标轴 x 轴从上向下延伸， y 轴从左向右延伸。

当输入用例 $n=0$ 时，表示输入终止，该用例无需处理。含四个数字 $x1,y1,x2,y2$ （不一定是整数）， $(x1,y1)$ 和 $(x2,y2)$ 分别是地图的左上角位置和右下角位置。

注意，坐标轴 x 轴从上向下延伸， y 轴从左向右延伸。
当输入用例 $n=0$ 时，表示输入终止，该用例无需处理。

[247. 亚特兰蒂斯 - AcWing题库](#)

