

编译时不打开任何优化开关, 栈大小与空间限制相等.

1 假人

arachne.cpp/in/out

时间限制: 0.777s

空间限制: 512MB

1.1 题目描述

悠久的 Reflesia 大陆抵抗地灾军团入侵的战争进入了僵持的阶段, 世代生活在 Reflesia 这片大陆的精灵们开始寻找远古时代诸神遗留的神器, 试图借助神器的神秘力量帮助她们战胜 BE-MANI 军团。

在付出了惨痛的代价后, 精灵们从步步凶险的远古战场取回了一件保存尚完好的神杖。但在经历过那场所有史书都视为禁忌的“REBELLIO”后, 神杖上镶嵌的奥术宝石已经残缺, 神力也几乎消耗殆尽。精灵高层在至高会议中决定以举国之力收集残存至今的奥术宝石, 并重金悬赏天下能工巧匠修复这件神杖。

你作为神术一脉第五百七十三位传人, 接受了这个艰巨而神圣的使命。神杖上从左到右镶嵌了 n 颗奥术宝石, 奥术宝石一共有 $10^9 + 1$ 种, 用数字 $0 - 10^9$ 表示, 宝石的种类编号就是其能量值。所有位置的宝石已经残缺, 你需要用完好的奥术宝石填补每一处残缺的部分。其中第 i 个位置只能放置能量为 $[l_i, r_i]$ 之间的宝石。

神杖的初始神力值为 $Magic = 0$ 。拼装好的神杖会把上面宝石的能量按顺序 $1 - n$ 视为一个序列, 之后它会一个个遍历这个序列的所有 2^n 个子序列, 将子序列中任意相邻两颗宝石的能量差的绝对值的和求和, 之后若这个值比当前神力值大, 它就会把神力值更新为这个值。具体地说, 令一个序列 A 的 $AR = \sum_{i=1}^{|A|-1} |A_i - A_{i+1}|$, 而最终的神力值就是神杖所有子序列中 AR 值的最大值。

Pastel 希望你能够计算出神杖的神力值最大可能是多少, 以此排兵布阵。

1.2 输入格式

第一行一个正整数 n 。

接下来 n 行每行两个非负整数 l_i, r_i 。

1.3 输出格式

输出一个整数表示最大神力值。

1.4 样例数据

1.4.1 样例 1 输入

```
4
0 2
1 5
6 7
3 3
```

1.4.2 样例 1 输出

11

1.4.3 样例 1 解释

令序列为 $[0, 3, 7, 3]$ ，取子序列 $[0, 7, 3]$ 。

1.4.4 样例 2

见下发文件 arachne0.in/out。

1.5 数据范围

对于所有数据, $n \leq 10^5, l_i \leq r_i \leq 10^9$ 。

测试点编号	n	特殊性质
1, 2, 3	$= 5$	$r_i \leq 2$
4, 5	$= 10^5$	$l_i = r_i$
6, 7	$= 10^5$	$l_i = 0$
8, 9	$= 10^5$	$r_i - l_i \leq 10$
10	$= 10^5$	

2 愚者

stulti.cpp/in/out

时间限制: 1.888s

空间限制: 512MB

2.1 题目描述

BEMANI 的军师 STULTI 从潜伏在精灵高层的密探手中得知了神杖的情报，他对奥术宝石中蕴含的远古神秘力量十分感兴趣。他设计夺取了数块奥术宝石，并带领手下的研究人员全力破解。经过了一个月的艰苦尝试，终于破译了其内部能量结构。

奥数宝石可以抽象成一个 n 个点与 m 条边的无向图，点的编号为 $1 - n$ 。其中每个点有一个基础能量 a_i ，每个点的基础能量互不相同。整个宝石有一个附加能量 b 。宝石的所有点必须能互相到达，才能释放出魔力。宝石的魔力值被定义为所有点的基础能量的最小值加上宝石的整体能量，再乘以宝石中的点数。具体地说，一个 n 个点的连通图的魔力值为 $(\forall i, \min\{a_i\} + b) * n$ 。

同时，宝石的结构是可以修改的。你可以删除宝石中任意个点，得到新的宝石，并可能改变其魔力值。注意宝石需要是一个连通图才有魔力值。现在 STULTI 手中有一块没有重边的宝石，他会询问 q 次，在整个宝石的附加能量 b 为他询问的值时，经过修改后宝石的魔力值最大可能是多少。

一个没有点的图的魔力值为 -10^{10} 。

2.2 输入格式

第一行三个正整数 n, m, q 。

接下来一行 n 个整数，第 i 个表示 a_i 。

接下来 m 行每行两个数 u_i, v_i 表示一条连接 u_i, v_i 的边。

接下来 q 行每行一个整数 b_i 。

2.3 输出格式

输出 q 行，每行一个整数表示当前最大魔力值。

2.4 样例数据

2.4.1 样例 1 输入

```
4 2 3
1 2 3 4
1 2
3 4
1000
-1000
0
```

2.4.2 样例 1 输出

2006

-996

6

2.4.3 样例 1 解释

分别保留点 $\{3, 4\}, \{4\}, \{3, 4\}$ 。

2.4.4 样例 2

见下发文件 stulti0.in/out。

这个样例有 0.01% 的可能性是错误的。

2.5 数据范围

对于所有数据, $n, q \leq 2 * 10^5, \frac{n}{2} \leq m \leq 3 * 10^5, m \leq 2 * n, -10^6 \leq a_i, b \leq 10^6$ 。

测试点编号	n	q	特殊性质
1, 2, 3	$= 5$	$= 1$	
4, 5	$= 100$	$= 100$	图是一棵树
6, 7	$= 100$	$= 100$	
8, 9, 10	$= 1000$	$= 3000$	图是一棵树
11, 12, 13, 14	$= 2 * 10^5$	$= 1$	
15, 16, 17	$= 1000$	$= 3 * 10^5$	图是一条 $1 - n$ 顺次连接的链
18, 19, 20	$= 2 * 10^5$	$= 3 * 10^5$	

3 拙者

sessha.cpp/in/out

时间限制: 2.573s

空间限制: 512MB

3.1 题目描述

So happy.

So super happy.

So happy I could fly like a butterfly. (Squeals from delight)

Super yeah.

Super happy association.

Something like the feeling of eating a bunch of shrimp.

So, if you are eating some shrimp, please play and be happy happy happy, ha ha ha!

--VERY CLASSICAL DANCE WITH VERY EUPHORIA FOR VERRRRRY JUVENILE

我们定义一个括号序列是高兴的, 当且仅当它右括号的个数不超过左括号的个数。

现在你要维护一个长度为 n 的括号序列, 编号为 $1 - n$, 支持以下三种操作:

1 p 表示将第 p 个括号反向, 既 (变为),) 变为 (。

2 $l\ r$ 表示询问: 如果要使第 l 个括号到第 r 个括号组成的序列的每个前缀都是高兴的, 至少需要删去多少个位置上的括号。

3 $l\ r$ 表示询问: 如果要使第 l 个括号到第 r 个括号组成的序列的每个前缀和每个后缀都是高兴的, 至少需要删去多少个位置上的括号。

3.2 输入格式

第一行一个正整数 n 表示序列长度, 接下来一行 n 个字符描述原始序列。

第三行一个正整数 q 表示操作数, 接下来 q 行每行一组操作, 格式同题目描述。

3.3 输出格式

输出等同于操作 2, 3 次数的行数, 每行一个整数表示查询的答案。

3.4 样例数据

3.4.1 样例 1 输入

```
7
(( )) ( ) (
4
2 2 6
3 2 6
1 3
3 2 6
```

3.4.2 样例 1 输出

1
2
1

3.4.3 样例 1 解释

一种可行的方案为三次询问分别删除位置在原序列中 $\{4\}$, $\{4, 6\}$, $\{6\}$ 的括号。

3.4.4 样例 2

见下发文件中的 sessha0.in/out。这个样例中只包含操作 2,3。

这个样例有 0.01% 的可能性是错误的。

3.5 数据范围

对于所有数据, $n \leq 2 * 10^5$, $q \leq 3 * 10^5$, $1 \leq x \leq n$, $1 \leq l \leq r \leq n$ 。

测试点编号	n	q	可能出现的操作
1, 2	$= 5$	$= 1$	2,3
3, 4, 5	$= 100$	$= 200$	1,2,3
6, 7, 8	$= 1000$	$= 2000$	1,2,3
9, 10, 11, 12	$= 2 * 10^5$	$= 3 * 10^5$	1,2
13, 14, 15, 16	$= 2 * 10^5$	$= 3 * 10^5$	2,3
17, 18, 19, 20	$= 2 * 10^5$	$= 3 * 10^5$	1,2,3