

牛客 CSP-S 提高组赛前集训营 2

比赛地址: <https://ac.nowcoder.com/acm/contest/1101>

题目名称	服务器需求	沙漠点列	维护序列
题目类型	传统型	传统型	传统型
每个测试点时 限	C/C++ 3 秒, 其他语言 6 秒	C/C++ 2 秒, 其他语言 4 秒	C/C++ 5 秒, 其他语言 10 秒
内存限制	C/C++ 256MB, 其他语言 512MB	C/C++ 512MB, 其他语言 1024MB	C/C++ 512MB, 其他语言 1024MB
子任务数目	20	10	10
测试点是否等 分	是	是	是

注意事项

- 所有参与牛客 OI 赛前集训营的选手必须遵守约定的纪律:
 - 比赛账号不能外传。
 - 比赛中不能抄袭代码。
 - 比赛中不能恶意卡评测。
 - 报名支付账号即为比赛账号。
 - 一旦报名牛客 OI 赛前集训营活动, 不支持退费, 请考虑清楚后报名。
 - 本活动解释权归牛客网所有, 活动介绍未尽事宜以牛客网官方解释为准。
- 欢迎关注“比赛自动姬”公众号, 关注更多比赛资讯~



服务器需求

【题目描述】

小多计划在接下来的 n 天里租用一些服务器，所有的服务器都是相同的。接下来 n 天中，第 i 天需要 a_i 台服务器工作，每台服务器只能在这 n 天中工作 m 天，这 m 天可以不连续。

但是计划不是一成不变的，接下来有 q 次修改计划（修改是永久的），每次修改某一天 k 的需求量 a_k 。

小多希望知道每次修改之后，最少需要多少台服务器。

【输入格式】

第一行三个正整数 n, m, q ，分别表示计划的天数，每台服务器能工作的天数和修改次数。

随后一行 n 个非负整数，第 i 个数字 a_i 表示原计划第 i 天需要多少台服务器工作。

随后 q 行，每行两个正整数 p_i, c_i ，表示把第 p_i 天需要的服务器数目改成 c_i 。

【输出格式】

第一行输出原计划需要的最少服务器数量。

随后 q 行，每行输出对应的修改之后，需要的最少的服务器的数量。

【样例 1 输入】

5 3 2

1 1 1 1 1

1 2

2 3

【样例 1 输出】

2

2

3

【样例 1 说明】

未修改时，可以租用 2 台服务器，分别安排给{1,4,5}和{2,3}这些天。

当第一次修改时，第一天需要两台服务器，需求变为了 2 1 1 1 1，故可以安排成 {1,2,3}和{1,4,5}，满足所有的需求。

第二次修改时，第二天需要三台服务器，需求变为了 2 3 1 1 1。可以安排三台服务器，每台服务器安排的日子分别为{1,2,3},{1,2,4}和{2,5}，这样可以满足所有天的需求。

【数据范围】

对于 100% 的数据 有 $a_i \leq 1000000000, 1 \leq p_i \leq n, c_i \leq 1000000000$ 。

数据点	n	m	q
1	100	≤ 100	100
2			
3	1000	1	1000
4			
5			
6		≤ 1000	200000
7			
8			
9	100000	1	100000
10			
11		≤ 100000	200000
12			
13	200000	1	100000
14		≤ 100	
15	400000	1	200000
16			
17			
18		≤ 400000	
19			
20			

沙漠点列

【题目描述】

我们称一张无向图是仙人掌，当且仅当这张无向图连通且每条边最多属于一个简单环。我们称一张无向图是沙漠，当且仅当这张无向图中所有连通子图都是仙人掌。

给出一个 n 个点， m 条边的沙漠，你可以删去其中的 k 条边。求能分成的连通块数量最大值。

【输入格式】

第一行输入三个自然数 n, m, k ($3 \leq n \leq 10^6; 0 \leq k \leq m \leq 2 \times 10^6$)。

接下来 m 行，每行输入两个正整数 u, v ($1 \leq u, v \leq n$)。保证无重边、无自环。。

【输出格式】

输出一行一个正整数，表示答案。

【样例 1 输入】

5 5 3

1 2

2 3

2 4

2 5

4 5

【样例 1 输出】

3

【样例 1 说明】

一种最优的方案是：删去 2-4，2-5，4-5 这三条边，剩下三个连通块： $\{1,2,3\}, \{4\}, \{5\}$ 。

【数据范围】

输入数据较大，建议使用读入优化。

测试点编号	n	m	特殊性质
1	$\leq 1\,000$	$\leq 2\,000$	$k = 0$
2	$\leq 10^6$	$\leq 2 \times 10^6$	
3	$\leq 1\,000$	$\leq 2\,000$	原图是森林
4	$\leq 10^6$	$\leq 2 \times 10^6$	
5	$\leq 1\,000$	$\leq 2\,000$	原图是基环树森林
6	$\leq 10^6$	$\leq 2 \times 10^6$	
7	$\leq 1\,000$	$\leq 2\,000$	原图是仙人球沙漠
8	$\leq 10^6$	$\leq 2 \times 10^6$	
9	$\leq 1\,000$	$\leq 2\,000$	无特殊性质
10	$\leq 10^6$	$\leq 2 \times 10^6$	

森林：原图中每个连通块都是树，即 x 个点 $x - 1$ 条边的无向连通图。

基环树森林：原图中每个连通块都是树或基环树，即 x 个点 x 条边的无向连通图。

仙人球沙漠：原图中每个连通块都是仙人球，即每个点最多属于一个简单环的无向连通图。

维护序列

【题目描述】

给出一个长度为 n 的序列 a_1, a_2, \dots, a_n 。你需要实现一个数据结构，支持以下操作：

- $1\ l\ r\ x$ 将 $a_l, a_{l+1}, a_{l+2}, \dots, a_r$ 修改为 x 。
- $2\ x\ y$ 查询序列中所有满足 $a_i = x, a_j = y$ 的点 (i, j) 里 $|i - j|$ 的最小值 (i 和 j 可以相同)，如找不到满足条件的点对，输出 -1 。

【输入格式】

第一行输入两个正整数 n, m, f ($1 \leq n, m \leq 10^5, f \in \{0, 1\}$)，表示序列长度、操作数，是否强制在线。 $f = 0$ 表示没有强制在线， $f = 1$ 表示强制在线。

接下来输入一行 n 个正整数 a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$)，描述初始序列。

接下来 m 行，每行先输入一个数 opt ($opt \in \{1, 2\}$)，表示操作类型。

- 若 $opt = 1$ ，则再输入三个正整数 l, r, x 保证解密之后满足 ($1 \leq l \leq r \leq n, 1 \leq x \leq n$)，意义如题目所述；
- 若 $opt = 2$ ，则再输入两个正整数 x, y 保证解密之后满足 ($1 \leq x, y \leq n$)，意义如题目所述。

若 $f = 1$ ，你需要对输入 l, r, x, y 都异或上次询问的答案 $lastans$ 进行解密，初始时 $lastans = 0$ 。若 $lastans = -1$ ，则异或 0 。

【输出格式】

对于每个 $\text{opt} = 2$ 的操作，输出所求的最小值。

【样例 1 输入】

```
4 5 0
1 2 3 2
2 1 3
1 1 2 1
2 1 3
1 4 4 1
2 2 3
```

【样例 1 输出】

```
2
1
-1
```

【样例 1 说明】

初始序列为 $[1, 2, 3, 2]$ 。

对于第一次询问，满足条件的点对只有 $(1, 3)$ ，输出 2。

第一次修改后序列为 $[1, 1, 3, 2]$ 。

对于第二次询问，满足条件的点对有 $(1, 3)$ 和 $(2, 3)$ ， $|1 - 3| = 2 > |2 - 3| = 1$ ，
因此输出 1。

第二次修改后序列为 $[1, 1, 3, 1]$ 。

对于第三次询问，无法找到 2，因此输出 -1。

【样例 2 输入】

4 5 1
1 2 3 2
2 1 3
1 3 0 3
2 3 1
1 5 5 0
2 3 2

【样例 2 输出】

2
1
-1

【样例 2 说明】

这是样例 1 的强制在线版本。

【数据范围】

输入数据较大，建议使用读入优化。

测试点编号	n, m	f	特殊性质
1	≤ 100	0	数据随机生成
2		1	
3	$\leq 1\,000$	0	
4		1	
5		0	
6		1	
7	$\leq 10^5$	0	无修改操作
8		1	
9		0	
10		1	
11		0	对于每个修改操作， $l_i = r_i$
12		1	
13		0	
14		1	
15		0	无
16		1	
17		0	
18		1	
19		0	
20		1	