

NOI 模拟赛

题目名称	简单函数	消除方块	摩天大楼
题目类型	传统型	传统型	传统型
可执行文件名	function	clear	skyscraper
输入文件名	function.in	clear.in	skyscraper.in
输出文件名	function.out	clear.out	skyscraper.out
时间限制	5.0 秒	0.5 秒	2.0 秒
内存限制	1024 MiB	512 MiB	512 MiB
测试点数目	20	25	25
测试点等分	是	是	是

提交源程序文件名

对于 C++ 语言	function.cpp	clear.cpp	skyscraper.cpp
-----------	--------------	-----------	----------------

编译选项

对于 C++ 语言	-lm -O2 -std=c++14
-----------	--------------------

1. C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 0。
2. 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格进行分隔。
3. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 在终端下可使用命令 `ulimit -s unlimited` 将栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
6. 对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。
7. 考试过程中若对题目有疑问，请联系出题人。
8. 题目很简单，AK 了请勿大声喧哗。

简单函数 (function)

【题目描述】

定义 $f(x) = (x \oplus a) - b$, 其中 a, b 为给定的参数, \oplus 符号表示按位异或。

给定 n 个变量 $x_1 \sim x_n$, 需要支持 q 次以下操作:

1 k v, 表示将 x_k 修改为 v 。

2 a b, 询问满足 $i \in [1, n-1]$, $f(x_i) \times f(x_{i+1}) \leq 0$ 的位置 i 。如果不存在输出 -1 , 如果有多个合法 i 输出任意一个即可。

【输入格式】

第一行两个数 n, q , 表示序列长度和操作次数。

第二行 n 个数 $x_1 \sim x_n$, 含义见题目描述。

接下来 q 行每行三个数, 表示每次操作。

【输出格式】

对于每个 2 操作, 输出一行一个数, 表示答案。

【样例 1 输入】

```
1 5 6
2 3 5 1 2 4
3 2 0 2
4 2 1 1
5 2 2 3
6 2 3 2
7 2 4 2
8 2 5 8
```

【样例 1 输出】

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

【样例 2】

见选手目录下 *function2.in* 与 *function2.ans*, 该样例满足测试点 1 ~ 3 的性质。

【样例 3】

见选手目录下 *function3.in* 与 *function3.ans*, 该样例满足测试点 4 ~ 6 的性质。

【子任务】

对于全部的数据, $1 \leq n, q \leq 10^6, 0 \leq x_i, a, b, v < 2^{30}, 1 \leq k \leq n$ 。

测试点编号	$n, q \leq$	特殊性质
1 ~ 3	2000	无
4 ~ 6	10^5	有
7 ~ 10	10^6	有
11 ~ 14	10^5	无
15 ~ 20	10^6	无

特殊性质: 保证不存在操作 1。

消除方块 (clear)

【题目描述】

给定一个长度为 n 的序列，每个位置有一个方块，方块有 R 或 B 的一种颜色。

给定 r, b ，如果一段长度为 $r + b$ 的区间满足有 r 个颜色为 R 的方块， b 个颜色为 B 的方块，那么可以消除这个区间。

有一些方块的颜色仍未确定，用 W 表示。求这些方块的颜色任意分配下，可能达到的最大消除次数为多少。

【输入格式】

第一行三个数 n, r, b 含义见题目描述。

接下来一个长度为 n 的字符串，表示颜色序列。

【输出格式】

一行一个数，表示答案。

【样例 1 输入】

```
1 4 1 1
2 BBWR
```

【样例 1 输出】

```
1 2
```

【样例 2】

见选手目录下 *clear2.in* 与 *clear2.ans*，该样例满足测试点 4 ~ 7 的性质。

【样例 3】

见选手目录下 *clear3.in* 与 *clear3.ans*，该样例满足测试点 17 ~ 21 的性质。

【子任务】

对于全部的数据，满足 $1 \leq n \leq 3 \times 10^5, 1 \leq r, b \leq n - 1, 2 \leq r + b \leq n$ 。

测试点编号	$n \leq$	特殊性质
1 ~ 3	20	无
4 ~ 7	500	无
8 ~ 9	5000	$r + b \geq 100$
10 ~ 13	5000	无
14 ~ 16	3×10^5	$r + b \geq 500$
17 ~ 21	10^5	无
22 ~ 25	3×10^5	无

摩天大楼 (skyscraper)

【题目描述】

有 n 栋摩天大楼排成一排，第 i 座的高度为 h_i 。每栋摩天大楼都有一个电梯，第 i 栋摩天大楼的电梯上下一层需要 w_i 的时间。

每栋摩天大楼的每层都有恰好一个人，你想修建若干天桥让任意两人都可以通过电梯和天桥见面。但由于材料有限，你只能修建 $n - 1$ 个天桥。修建天桥需要满足以下条件：天桥必须是水平的，也就是说，你只能在第 i 栋楼的第 x 层和第 j 栋楼的第 x 层间修建天桥。并且天桥不能穿过其他大楼，也就是对于任意 $k \in [i + 1, j - 1]$ ，需要满足 $h_k < x$ 。当然也需要满足 $h_i, h_j \geq x$ 。经过一个天桥的时间是 V 。

你想要让大家想见面时能尽快见到对方。具体来说，设一共有 $R = \sum_{i=1}^n h_i$ 个人， $d(x, y)$ 表示第 x 个人去见到第 y 个人需要的时间。你需要求出 $\sum_{i=1}^R \sum_{j=i+1}^R d(i, j)$ 的最小值。

【输入格式】

第一行两个数 n, V ，表示摩天大楼的数量和经过一个天桥的时间。

接下来 n 行每行两个数 h_i, w_i ，分别表示摩天大楼的高度和电梯运行一层时间。

【输出格式】

一行一个数表示答案。

【样例 1 输入】

```
1 2 1
2 3 3
3 3 2
```

【样例 1 输出】

```
1 59
```

【样例 2】

见选手目录下 *skyscraper2.in* 与 *skyscraper2.ans*，该样例满足测试点 6 ~ 8 的性质。

【样例 3】

见选手目录下 *skyscraper3.in* 与 *skyscraper3.ans*, 该样例满足测试点 13 ~ 15 的性质。

【样例 4】

见选手目录下 *skyscraper4.in* 与 *skyscraper4.ans*, 该样例满足测试点 16 ~ 18 的性质。

【样例 5】

见选手目录下 *skyscraper5.in* 与 *skyscraper5.ans*, 该样例满足测试点 19 ~ 20 的性质。

【子任务】

对于全部的数据, $1 \leq n \leq 60, 1 \leq R \leq 3000, 1 \leq w_i, V \leq 10^6$ 。

测试点编号	$n \leq$	$R \leq$	特殊性质
1 ~ 3	8	50	无
4 ~ 5	13	100	无
6 ~ 8	18	300	无
9 ~ 10	30	1000	无
11 ~ 12	40	1500	h_i 全部相同
13 ~ 15	40	1500	$w_i = 1, V = 10^6$
16 ~ 18	40	1500	$w_i = 10^6, V = 1$
19 ~ 20	40	1500	无
21 ~ 22	50	2000	无
23 ~ 25	60	3000	无