Water Problem Choose Talk

April 1, 2021

CF679E

E. Bear and Bad Powers of 42

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Limak, a bear, isn't good at handling queries. So, he asks you to do it.

We say that powers of 42 (numbers 1, 42, 1764, ...) are *bad*. Other numbers are *good*.

You are given a sequence of n good integers $t_1, t_2, ..., t_n$. Your task is to handle q queries of three types:

1. `1 i` — print t_i in a separate line.
2. `2 a b x` — for $i \in [a, b]$ set t_i to x. It's guaranteed that x is a good number.
3. `3 a b x` — for $i \in [a, b]$ increase t_i by x. After this repeat the process while at least one t_i is bad.

You can note that after each query all t_i are good.

Input

The first line of the input contains two integers n and q ($1 \le n, q \le 100\,000$) — the size of Limak's sequence and the number of queries, respectively.

The second line of the input contains n integers $t_1, t_2, ..., t_n$ ($2 \le t_i \le 10^9$) — initial elements of Limak's sequence. All t_i are good.

Then, q lines follow. The i-th of them describes the i-th query. The first number in the line is an integer $type_i$ ($1 \le type_i \le 3$) — the type of the query. There is at least one query of the first type, so the output won't be empty.

In queries of the second and the third type there is $1 \le a \le b \le n$.

In queries of the second type an integer x ($2 \le x \le 10^9$) is guaranteed to be good.

In queries of the third type an integer x ($1 \le x \le 10^9$) may be bad.

Output

For each query of the first type, print the answer in a separate line.

Solution

维护线段树上每个节点离 42 的倍数最近的那个，有跨过 42 的倍数的数就向下递归。

Luogu P4198 楼房重建

题目描述

展开

小 A 的楼房外有一大片施工工地，工地上有 N 栋待建的楼房。每天，这片工地上的房子拆了又建、建了又拆。他经常无聊地看着窗外发呆，数自己能够看到多少栋房子。

为了简化问题，我们考虑这些事件发生在一个二维平面上。小 A 在平面上 $(0,0)$ 点的位置，第 i 栋楼房可以用一条连接 $(i,0)$ 和 (i,H_i) 的线段表示，其中 H_i 为第 i 栋楼房的高度。如果这栋楼房上任何一个高度大于 0 的点与 $(0,0)$ 的连线没有与之前的线段相交，那么这栋楼房就被认为是可见的。

施工队的建造总共进行了 M 天。初始时，所有楼房都还没有开始建造，它们的高度均为 0。在第 i 天，建筑队将会将横坐标为 X_i 的房屋的高度变为 Y_i（高度可以比原来大—修建，也可以比原来小—拆除，甚至可以保持不变—建筑队这天什么事也没做）。请你帮小 A 数数每天在建筑队完工之后，他能看到多少栋楼房?

输入格式

第一行两个正整数 N,M。

接下来 M 行，每行两个正整数 X_i,Y_i。

输出格式

M 行，第 i 行一个整数表示第 i 天过后小 A 能看到的楼房有多少栋。

说明/提示

对于 100% 的数据，$1\le X_i\le N$，$1\le Y_i\le 10^9$，$1\le N,M\le 10^5$。

solution

简化一下：给定序列 a，求有多少数大于前面所有数，支持修改，每次改一个数。

考虑维护一个支持以下操作的线段树：

对于线段树上任一颗子树（对应区间 $l \sim r$），给定 x，求出 $x\ a_l\ a_{l+1}\ \dots\ a_r$ 这个序列在原问题下的答案；支持单点修改。

令 m 为这颗子树的分界点，令 p 为 $l\ m$ 中最大的数的位置，如果 $x > a_p$，那么相当于 $x\ a_{m+1}\ a_{m+2}\ \dots\ a_r$ 的答案，否则就是 $x\ a_l\ a_{l+1}\ \dots\ a_m$ 的答案加上 $a_p\ a_{m+1}\ a_{m+2}\ \dots\ a_r$ 的答案减一，发现后者的答案于 x 无关，递归下去可以做到 $O(\log n)$。

修改时，没有被修改的字数不变，修改过的子树只有 $O(\log n)$ 个，每次重新计算这 $O(\log n)$ 个子树即可，单次修改询问复杂度 $O(\log^2 n)$。

火星在住

给定一棵树，带点权，对于每个 k，求出大小为 k 的最大权独立集，$n \leq 10^5$。

solution

树刨，分治，闵可夫斯基和。

CF414E

E. Forensic Examination

time limit per test: 6 seconds
memory limit per test: 768 megabytes
input: standard input
output: standard output

The country of Reberland is the archenemy of Berland. Recently the authorities of Berland arrested a Reberlandian spy who tried to bring the leaflets intended for agitational propaganda to Berland illegally . The most leaflets contain substrings of the Absolutely Inadmissible Swearword and maybe even the whole word.

Berland legal system uses the difficult algorithm in order to determine the guilt of the spy. The main part of this algorithm is the following procedure.

All the m leaflets that are brought by the spy are numbered from 1 to m. After that it's needed to get the answer to q queries of the following kind: "*In which leaflet in the segment of numbers $[l, r]$ the substring of the Absolutely Inadmissible Swearword $[p_l, p_r]$ occurs more often?*".

The expert wants you to automate that procedure because this time texts of leaflets are too long. Help him!

Input

The first line contains the string s ($1 \le |s| \le 5 \cdot 10^5$) — the Absolutely Inadmissible Swearword. The string s consists of only lowercase English letters.

The second line contains the only integer m ($1 \le m \le 5 \cdot 10^4$) — the number of texts of leaflets for expertise.

Each of the next m lines contains the only string t_i — the text of the i-th leaflet. The sum of lengths of all leaflet texts doesn't exceed $5 \cdot 10^4$. The text of the leaflets consists of only lowercase English letters.

The next line contains integer q ($1 \le q \le 5 \cdot 10^5$) — the number of queries for expertise.

Finally, each of the last q lines contains four integers l, r, p_l, p_r ($1 \le l \le r \le m$, $1 \le p_l \le p_r \le |s|$), where $|s|$ is the length of the Absolutely Inadmissible Swearword.

Output

Print q lines. The i-th of them should contain two integers — the number of the text with the most occurences and the number of occurences of the substring $[p_l, p_r]$ of the string s. If there are several text numbers print the smallest one.

solution

变为括号序列。

Interactive Sort

Problem I. Interactive Sort

Time limit: `10 seconds`

Ivan wants to play a game with you. He took all integers from 1 to n inclusive, shuffled them and then put all even numbers into array e and all odd numbers into array o.

Your task is to find arrays e and o.

You can ask Ivan questions of certain kind. Each question consists of two integers i and j. For each question Ivan says whether $e[i] < o[j]$ or not.

You can ask at most 300 000 questions.

Interaction Protocol

First, the testing system writes the integer n ($1 \le n \le 10\,000$) — the number of integers Ivan used.

Your solution shall print requests of two types:

- "? i j". $1 \le i \le \lfloor \frac{n}{2} \rfloor, 1 \le j \le \lceil \frac{n}{2} \rceil$. The testing system responds with the symbol "<" if $e[i] < o[j]$ or with the symbol ">" otherwise.
- "! e_1 e_2 ... $e_{\lfloor \frac{n}{2} \rfloor}$ o_1 o_2 ... $o_{\lceil \frac{n}{2} \rceil}$" tells the values of e and o that your program has determined.

Don't forget to flush the output after each request!

Your solution must issue exactly one request of the second type, which must be the last request, and the solution must terminate gracefully after issuing it.

Your solution is allowed to issue at most 300 000 requests of the first type.

For each test case the number n is fixed and the numbers are shuffled using Java built-in shuffle function with fixed seed.

Mole Tunnels

Problem M. Mole Tunnels

Input file: `mole.in`
Output file: `mole.out`

Moles create tunnels for traveling between their holes. In this problem we investigate one tunnel system that was built by moles. It consists of n holes and $n-1$ tunnels connecting them. Let us number all holes from 1 to n. Then for all $i > 1$, a hole number i is connected by a tunnel to the hole number $\lfloor \frac{i}{2} \rfloor$. Tunnels are bidirectional. For each hole i we know the *amount of food* c_i in that hole. It means that there is enough food for exactly c_i moles in that hole.

There are m moles living in the tunnel system. For each mole i you are given an integer p_i — the hole number where the mole i is currently sleeping. In the morning, the first k moles wake up and want to eat, while $m-k$ others are sleeping. Each of k woken up moles chooses some hole and crawls to it. They are quite smart, so they want to minimize the total distance traveled. The distance traveled by one mole is the total number of tunnels which it uses to get from one hole to another. The first k moles who woke up want to move in such a way, so that there is enough food for them at the holes they choose to crawl to. It means that in the hole i there are no more than c_i woken up moles after all their movements are done.

You must find the minimum total distance for all k from 1 to m. It is guaranteed that there always exists a way for all moles to eat.

Input

The first line contains two integers n and m ($1 \le n, m \le 10^5$) — the number of holes and moles. The second line contains n integers c_i ($0 \le c_i \le m$) — the amount of food in the hole i. The third line contains m integers p_i ($1 \le p_i \le n$) — the starting positions of the moles.

Output

You must print m numbers. The k-th number is the minimum total distance the first k moles need to travel if they woke up first.

Indiana Jones and the Uniform Cave

Indiana Jones has stuck in the Uniform Cave. There are many round chambers in the cave, and all of them are indistinguishable from each other. Each chamber has the same number of one-way passages evenly distributed along the chamber's wall. Passages are indistinguishable from each other, too. The Cave is magical. All passages lead to other chambers or to the same one. However, the last passage, after all passages are visited, leads to the treasure. Even the exact number of chambers is a mystery. It is known that each chamber is reachable from each other chamber using the passages.

Dr. Jones noticed that each chamber has a stone in the center. He decided to use these stones to mark chambers and passages. A stone can be placed to the left or to the right of one of the passages. When Indiana Jones enters the chamber all that he can observe is the location of the stone in the chamber. He can move the stone to the desired location and take any passage leading out of the chamber.

Your task is to help Indiana Jones to visit every passage in the Uniform Cave and find the treasure.

Interaction Protocol

First, the testing system writes the integer m — the number of passages in each chamber ($2 \le m \le 20$).

Dr. Jones enters the chamber and sees, in the next line, where the stone is placed: either in the "`center`" of the chamber or to the "`left`", or to the "`right`" of some passage. On the first visit to the chamber, the stone is in the center.

Your solution shall output his actions: the number and the side of the passage to place the stone to, and the number of the passage to take. Both numbers are relative to the passage marked by the stone, counting clockwise from 0 to $m - 1$. If the stone is in the center of the chamber, the origin is random.

For example, "`3 left 1`" tells that Dr. Jones moves the stone three passages clockwise and places it to the left of the passage, then he takes the passage to the right of the initial stone position.

After each move testing system tells either the location of the stone in the next chamber or "`treasure`", if Indiana Jones had found it. The testing system writes "`treasure`" when all the passages are visited.

If Dr. Jones does not find the treasure room after $[20\,000]_r^1$ passages are taken, he starves to death, and your solution receives the "Wrong Answer" outcome. You also receive this outcome if your solution terminates before all passages are taken. Err(1)

The total number of chambers in the cave is unknown, but you may assume that it does not exceed 20, and that each chamber is reachable from every other chamber.

History Course

Problem G: History course

You are to give a series of lectures on important historical events, one event per lecture, in some order. Each event lasted for some time interval $[a_i, b_i]$. We say that two events are *related* if their intervals have a common point. It would be convenient to schedule lectures on related events close to each other. Moreover, lectures on unrelated events should be given in the order in which the events have taken place (if an event A preceded an unrelated event B, then the lecture on A should precede the lecture on B).

Find the minimum integer $k \geqslant 0$ and an order of the lectures such that any two related events are scheduled at most k lectures apart from each other (lectures number i and j are considered to be $|i - j|$ lectures apart).

Input

The first line of input contains the number of test cases T. The descriptions of the test cases follow:

The first line of each test case contains the number n, $1 \leqslant n \leqslant 50\,000$. Each of the next n lines contains two integers a_i and b_i, $-10^9 \leqslant a_i \leqslant b_i \leqslant 10^9$ – the ends of the i-th interval. The intervals are pairwise different.

Output

Print the answers to the test cases in the order in which they appear in the input. The first line of the answer to each test case should contain the minimum value of k. The next n lines should list the intervals (in the same format as in the input) in an order such that any two related events are scheduled at most k lectures apart. Remember to put any unrelated intervals in the proper order!