

Задача А. Военный парад

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В честь Дня Независимости Байтландии правительство страны решило организовать военный парад. В воинскую часть почетного караула Байтландии пришёл приказ подготовить торжественную шеренгу солдат. Руководство части, в которой уже много лет исправно служит капитан Килобайтин, доверило это ответственное поручение именно ему.

Капитану известно, что всего в части служат N солдат, рост каждого i -го ($1 \leq i \leq N$) солдата равен H_i нанометров. Шеренгой будем называть любую последовательность целых чисел A_i , таких, что $1 \leq A_i \leq N$ и $A_i \neq A_j$, если $i \neq j$. Длина шеренги – это длина соответствующей последовательности. Шеренга называется торжественной, если разница в росте любых двух стоящих рядом солдат отличается не более чем на K нанометров. То есть если для последовательности A_i длиной M выполняется правило, что для любого $1 \leq i \leq M - 1$ верно $|H_{A_i} - H_{A_{i+1}}| \leq K$.

Капитан полагает, что получение им нового воинского звания напрямую зависит от длины подготовленной им торжественной шеренги. Ваша задача – помочь капитану Килобайтину выполнить приказ и подготовить торжественную шеренгу максимально возможной длины.

Формат входных данных

Первая строка входных данных содержит два натуральных числа, разделенные одиночным пробелом N ($2 \leq N \leq 10^5$) и K ($0 \leq K \leq 10^9$) соответственно.

Вторая строка входного файла содержит ровно N целых чисел H_i ($1 \leq H_i \leq 10^9$) – рост i -го солдата. Числа разделены одиночными пробелами. Солдаты нумеруются последовательно в порядке их ввода начиная с единицы.

Формат выходных данных

Первая строка выходного файла должна содержать одно число M – максимальную длину торжественной шеренги.

Вторая строка выходного файла должна описывать торжественную шеренгу и содержать M целых чисел A_i , числа должны быть разделены одиночными пробелами. Солдаты нумеруются последовательно в порядке их ввода. Если решений несколько, то выведете любое из них.

Примеры

стандартный ввод	стандартный вывод
3 20 1830 1800 1700	1 2
5 7 7 20 16 29 15	3 3 5 2
8 140 170 601 300 200 500 10 170 999	4 3 1 4 7

Замечание

В задаче кроме тестов из условия есть 20 тестов, каждый оценивается отдельно в 5 баллов.

Задача В. Контейнеры: перезагрузка

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В компании грузоперевозок Нурлаш и КО inc. для перевозок грузов используют контейнеры разных размеров. Когда нужда в них отсутствует, некоторые из них вкладывают в другие для экономии места. Когда контейнер становится нужным, специальный робот достает его и вынимает содержимое.

Вы главный разработчик в компании и конечно знаете что разборочный робот работает следующим образом : сначала он вытаскивает все содержимое контейнера который нужно разобрать и выписывает его номер. После он запускает процедуры разбора для контейнеров которые вынули, в том порядке в котором они находились внутри него.

В конце процедуры разбора у вас появляется список номеров и вам очень не нравится когда в нем встречается плохая пара чисел. Плохой парой последовательности $A[]$ будем называть такую пару i и j что $A[i] > A[j]$ причем $i < j$. Вы решили переписать робота упаковщика, но для начала вы хотите узнать, на самом ли деле все так плохо? Для этого вы решили подсчитать, для каждого контейнера, сколько плохих пар получится в выписанной последовательности.

Формат входных данных

Первая строка входных данных содержит единственное число — $N(1 \leq N \leq 2 * 10^5)$, количество контейнеров. Все контейнеры пронумерованы различными целыми числами от 1 до N .

Следующие N строк содержат информацию о контейнерах. В $i + 1$ строке содержится описание контейнера под номером i в виде $K_i(1 \leq K_i < N)$ и K_i чисел $V_i(1 < V_i \leq N, V_i \neq i)$. Где K_i это количество контейнеров находящихся внутри i -го, V_i это их номера, в том порядке в котором они лежат внутри.

Гарантируется что каждый контейнер на прямую лежит ровно в одном другом, кроме контейнера под номером 1. Также все контейнеры на прямую или через другие контейнеры лежат внутри контейнера под номером 1.

Формат выходных данных

В единственную строку входного файла выведите N чисел. Где i -ое число — количество плохих пар которые получатся если разборочный робот начнет разбирать контейнер под номером i .

Система оценки

Данная задача содержит пять подзадач:

- $1 \leq N \leq 1000$, $K_1 = N - 1$. Подзадача оценивается в 10 баллов.
- $1 \leq N \leq 2 * 10^5$, $K_1 = N - 1$. Подзадача оценивается в 20 баллов.
- $1 \leq N \leq 500$. Подзадача оценивается в 10 баллов.
- $1 \leq N \leq 5000$. Подзадача оценивается в 25 баллов.
- $1 \leq N \leq 2 * 10^5$. Подзадача оценивается в 35 баллов.

Пример

стандартный ввод	стандартный вывод
3	1 0 0
2 3 2	
0	
0	

Задача С. Дробь

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Эта задача немного необычна — в ней вам предстоит реализовать интерактивное взаимодействие с тестирующей системой. Это означает, что вы можете делать запросы и получать ответы в online-режиме. Обратите внимание, что ввод/вывод в этой задаче — стандартный (то есть с экрана на экран). После вывода очередного запроса обязательно используйте функции очистки потока, чтобы часть вашего вывода не осталась в каком-нибудь буфере. Например, на C++ надо использовать функцию `fflush(stdout)`, на Java вызов `System.out.flush()` и `stdout.flush()` для языка Python.

В школах Бейтландии собираются ввести новую методику преподавания математики, которая предполагает, что все уроки будут вестись в интерактивно-игровой форме. Одной из самых всеми любимых игр является игра «Дробь», применяемая для изучения рациональных дробей.

Игра предназначена для двух игроков. Вначале игры игрокам предоставляется некоторая несократимая рациональная дробь $\frac{A}{B}$. Далее игроки ходят по очереди начиная с первого игрока. За один ход игрок обязан вычесть единицу либо из числителя дроби, если числитель больше единицы, либо из знаменателя, если знаменатель больше единицы. Если после этого дробь можно сократить, то выполняется сокращение дроби до несократимой.

Далее ход переходит к другому игроку. Игра продолжается до тех пор, пока игрок может сделать ход, то есть выигрывает тот игрок, после хода которого получается дробь $\frac{1}{1} = 1$.

Для примера выше если в начале игры дана дробь $\frac{2}{5}$, то после хода первого игрока может быть получена дробь $\frac{1}{5}$ (вычитанием единицы из числителя) либо дробь $\frac{1}{2}$ (вычитанием единицы из знаменателя и последующего сокращения).

Игра стала настолько популярна, что руководство школы решило в конце учебного года провести чемпионат школы по данной игре. Отличник Петя, учащийся пятого математического класса, решил выиграть этот чемпионат. Петя заметил, что из всех записавшихся на участие в чемпионате он является самым младшим, а также, что все его потенциальные соперники достаточно вежливые люди, которые предоставят Пете право сделать ход первым. Пете сказали, что он сможет победить независимо от ходов второго игрока, если пойдёт первым и будет оптимально ходить. Осознавая свое преимущество, Петя уверен, что можно разработать программу, которая позволит ему для любой заданной дроби одержать победу. Ваша задача — помочь юному дарованию стать чемпионом.

Формат входных данных

В единственной строке входных данных будут даны два целых положительных числа A и B ($1 \leq A, B \leq 2 \cdot 10^5$), задающие начальную дробь. Гарантируется, что всегда существует стратегия, позволяющая Пете выиграть.

Протокол взаимодействия

В каждый ход Пети выведите «А» (без кавычек), если Петя должен уменьшить числитель дроби и «В» (без кавычек), если Петя должен уменьшить знаменатель дроби. **Не забывайте делать операцию «flush»!**

В ответ на ваш запрос программа жюри выведет «А» , если второй игрок уменьшил числитель и «В», если второй игрок уменьшил знаменатель.

Как только дробь становится равной 1, ваше решение будет признано корректным если вы сделали последний ход.

В конце игры программа жюри выведет «OK» если вы выиграли и сделали последний ход и «WA» если вы проиграли и последний ход был сделан программой жюри. После этого ваша программа должна завершить работу.

Обратите внимание на оценку задачи.

Примеры

стандартный ввод	стандартный вывод
3 2 OK	A
5 2 A A OK	B A A

Замечание

В этой задаче есть 30 тестов, каждый оценивается отдельно. В каждом тесте подбираются случайные A и B , такие что они взаимно просты и для них существует выигрышная стратегия. При каждом новом запуске числа A и B новые. У каждого теста есть параметры P_1, P_2, P_3, P_4, P_5 . Программа жюри, которая играет против вас, при возможности уменьшит и числитель и знаменатель с вероятностью P_1 процентов пойдёт случайно, с вероятностью P_2 процентов уменьшит числитель, с вероятностью P_3 процентов уменьшит знаменатель, с вероятностью P_4 процентов пойдёт оптимально и с вероятностью P_5 процентов пойдёт не оптимально.

Обратите внимание, что вашей программе не передаётся номер теста.

№	Ограничения на A	Ограничения на B	P_1	P_2	P_3	P_4	P_5	Балл	Примечания
1	$A = 3$	$B = 2$	0	0	0	100	0	0	тест из условия
2	$A = 5$	$B = 2$	0	0	0	0	100	0	тест из условия
3	$A \leq 20$	$B \leq 20$	1	22	13	64	0	5	—
4	$A \leq 20$	$B \leq 20$	0	0	0	100	0	5	—
5	$A \leq 50$	$B \leq 50$	0	0	0	100	0	5	—
6	$A \leq 1000$	$B \leq 1000$	0	0	0	100	0	5	—
7	$A \leq 1000$	$B \leq 1000$	0	0	0	100	0	5	—
8	$A \leq 1000$	$B \leq 1000$	0	0	0	100	0	5	—
9	$A \leq 1000$	$B \leq 1000$	0	0	0	100	0	5	—
10	$A \leq 1000$	$B \leq 1000$	0	0	0	100	0	5	—
11	$A \leq 10000$	$B \leq 100$	0	0	0	100	0	5	—
12	$A \leq 100$	$B \leq 10000$	0	0	0	100	0	5	—
13	$A \leq 100000$	$B \leq 100000$	11	61	6	18	4	2	—
14	$A \leq 100000$	$B \leq 100000$	100	0	0	0	0	2	—
15	$A \leq 100000$	$B \leq 100000$	0	100	0	0	0	3	—
16	$A \leq 100000$	$B \leq 100000$	0	0	100	0	0	2	—
17	$A \leq 100000$	$B \leq 100000$	0	0	0	100	0	5	—
18	$A \leq 100000$	$B \leq 100000$	0	0	0	0	100	1	—
19	$A \leq 100000$	$B \leq 100000$	26	8	17	12	37	1	—
20	$A \leq 100000$	$B \leq 100000$	100	0	0	0	0	3	—
21	$A \leq 100000$	$B \leq 100000$	0	100	0	0	0	2	—
22	$A \leq 100000$	$B \leq 100000$	0	0	100	0	0	1	—
23	$A \leq 100000$	$B \leq 100000$	0	0	0	100	0	5	—
24	$A \leq 100000$	$B \leq 100000$	0	0	0	0	100	1	—
25	$A \leq 100000$	$B \leq 100000$	3	1	15	37	44	4	—
26	$A \leq 100000$	$B \leq 100000$	0	0	0	100	0	5	—
27	$A \leq 100000$	$B \leq 100000$	43	20	28	9	0	2	—
28	$A \leq 100000$	$B \leq 100000$	0	0	0	100	0	5	—
29	$A \leq 100000$	$B \leq 100000$	0	12	23	10	55	1	—
30	$A \leq 100000$	$B \leq 100000$	0	0	0	100	0	5	—

Задача D. Ghiță, Lică Sămădăul и Buză Spartă

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

«Ghiță» очень любит программирование. Его любимые занятия — играть с перестановками и проводить время со своей женой Зинаидой. На свою 10-летнюю годовщину свадьбы Зинаида подарила ему очень красивую перестановку, ведь она знала, что это лучший подарок, который может получить «Ghiță». Пусть P_j — это j -элемент перестановки для каждого $1 \leq j \leq N$.

«Ghiță» был так рад такому подарку, что начал вычислять значение Q_i для каждого i что $1 \leq i \leq N$. Q_i — это число возрастающих подпоследовательностей на префиксе длины i в перестановке.

Более формально для каждого $1 \leq i \leq N$, Q_i равняется числу последовательностей j_1, j_2, \dots, j_k , что $1 \leq j_1 < j_2 < \dots < j_{k-1} < j_k \leq i$ и $P_{j_1} < P_{j_2} < \dots < P_{j_k}$.

Q конечно не перестановка, но тоже крутая штука, поэтому «Ghiță» положил её рядом с P .

Все было нормально, пока не появился «Lică Sămădăul». Он хотел использовать систему эпиднадзора «Ghiță» для аморальных целей, а «Ghiță», будучи честным человеком, не помог ему. Разгневанный ответом «Ghiță», «Lică Sămădăul» нанял «Buză Spartă», чтобы помочь ему украсть самые ценные активы «Ghiță»: его перестановку и его жену. И так он и сделал.

На следующий день «Ghiță» выяснил, что P отсутствует, и теперь единственная возможность для «Ghiță» восстановить перестановку — это использовать массив Q , который у него есть. Как вы уже догадались, ваша задача — помочь «Ghiță» восстановить перестановку P , используя массив Q .

Формат входных данных

В первой строке ввода записано одно число N ($1 \leq N \leq 70\,000$).

Во второй строке через пробел записаны N целых чисел Q_1, Q_2, \dots, Q_N .

Размер входного файла не превышает **115 МБ**.

Мы советуем вам самостоятельно проверять время работы и использование памяти читающей части вашей программы, чтобы убедиться, что возможная неэффективность вашей программы не связана с этой частью.

Формат выходных данных

В первой и единственной строке выведите P — украденную перестановку.

Гарантируется, что существует ровно один возможный ответ (только один P имеет заданный Q).

Система оценки

№	Баллы	Ограничения	T = длине Q_i	Размер входных данных
0	0	Тесты из условия	—	—
1	10	$N \leq 9$	—	—
2	15	$N \leq 400$	$T \leq 18$	—
3	18	$N \leq 700$	—	—
4	17	$N \leq 40\,000$	$T \leq 171$	4.5 МБ
5	11	$N \leq 70\,000$	$T \leq 258$	10 МБ
6	7	$N \leq 70\,000$	$T \leq 314$	16 МБ
7	16	$N \leq 70\,000$	—	85 МБ
8	6	$N \leq 70\,000$	—	115 МБ

Примеры

стандартный ввод	стандартный вывод
4 1 2 5 6	3 2 4 1
6 1 3 5 9 11 21	1 6 3 4 2 5

Замечание

В первом примере $N = 4$ и $P = \{3, 2, 4, 1\}$

$Q_1 = 1$, так как $\{3\}$ — единственная возрастающая подпоследовательность $\{3\}$

$Q_2 = 2$, потому что $\{3\}$ и $\{2\}$ — единственные возрастающие подпоследовательности $\{3, 2\}$

$Q_3 = 5$, потому что $\{3\}, \{3, 4\}, \{2\}, \{2, 4\}, \{4\}$ — единственные возрастающие подпоследовательности $\{3, 2, 4\}$

$Q_4 = 6$, потому что $\{3\}, \{3, 4\}, \{2\}, \{2, 4\}, \{4\}, \{1\}$ — единственные возрастающие подпоследовательности $\{3, 2, 4, 1\}$.