

## Задача А. Смех, да и только!

Имя входного файла: `laugh.in`  
Имя выходного файла: `laugh.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Колобок любит много смеяться. Чтобы подготовиться к встрече с потенциальным противником, Лиса решает изучить его смех.

Лиса считает, что смех — это последовательность чередующихся букв «a» и «h». Так например, «ahahaha», «hah» и «a» являются смехом, а «abacaba» и «hh» — нет.

Колобок разговаривает очень быстро, поэтому все его слова сливаются в одно большое. Для исследования Лиса хочет понять, как долго он может смеяться. У неё есть строка — запись разговора Колобка. Лиса хочет узнать наибольшую длину смеха в этом разговоре.

Лиса просит вас помочь ей с этой задачей.

### Формат входного файла

В первой строке входного файла находится одно натуральное число  $n$  ( $1 \leq n \leq 10^5$ ) — длина строки с разговором колобка. Во второй строке находится строка из строчных латинских букв длины  $n$  — запись разговора колобка.

### Формат выходного файла

В выходной файл выведите одно число — наибольшую длину смеха в разговоре Колобка.

### Пример

<code>laugh.in</code>	<code>laugh.out</code>
5 ahaha	5
24 ahahrundayahahsofasthah	4
10 ahahaahaha	5

### Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение  $n \leq 5000$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 47 баллов.

Вторая группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 53 балла.

## Задача В. Путь в никуда

Имя входного файла: `path.in`  
Имя выходного файла: `path.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

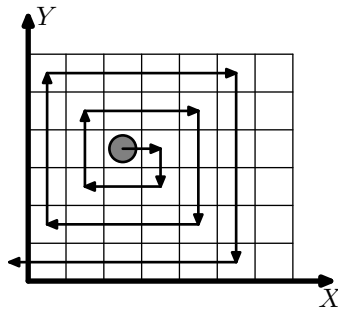
Колобку снится странный сон. В нём Колобок находится на клетчатом поле размера  $n \times m$  в клетке с координатами  $(x, y)$ .

Изначально Колобок смотрит вдоль положительного направления оси  $X$ . Затем он начинает идти по полю со следующей закономерностью:

- Пройти на одну клетку вперед. Повернуть на  $90^\circ$  вправо.
- Пройти на одну клетку вперед. Повернуть на  $90^\circ$  вправо.
- Пройти на две клетки вперед. Повернуть на  $90^\circ$  вправо.
- Пройти на две клетки вперед. Повернуть на  $90^\circ$  вправо.
- Пройти на три клетки вперед. Повернуть на  $90^\circ$  вправо.
- Пройти на три клетки вперед. Повернуть на  $90^\circ$  вправо.
- Пройти на четыре клетки вперед. Повернуть на  $90^\circ$  вправо.
- И так далее...

Движение продолжается до тех пор, пока Колобок не выйдет за границы поля. После этого Колобок просыпается.

Утром Колобок решил проанализировать свой сон. Он догадался, что в каждой клетке он был максимум один раз, но никак не может вспомнить, сколько клеток он посетил. Колобок просит вас написать программу, которая посчитает количество посещённых им клеток.



### Формат входного файла

В первой строке входного файла находятся два натуральных числа  $n, m$  ( $1 \leq n, m \leq 10^9$ ) — размеры доски вдоль оси  $X$  и оси  $Y$  соответственно. Во второй строке находятся два натуральных числа  $x, y$  ( $1 \leq x \leq n; 1 \leq y \leq m$ ) — координаты стартовой позиции колобка.

### Формат выходного файла

В выходной файл выведите одно число — количество клеток, посещенных Колобком во сне.

### Пример

<code>path.in</code>	<code>path.out</code>
7 6 3 4	36
2 2 1 1	2
2 2 1 2	4

## Комментарий

На рисунке наглядно показан первый пример.

## Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение  $n, m \leq 1000$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 25 баллов.

Вторая группа состоит из тестов, для которых выполняется ограничение  $n, m \leq 100\,000$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 27 баллов.

Третья группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 48 баллов.

## Задача С. Постройка забора

Имя входного файла: `polygon.in`  
Имя выходного файла: `polygon.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

После побега Колобка Дед и Баба решили построить забор вокруг своего дома, чтобы не допустить повторения истории.

Забор представляет собой многоугольник ненулевой площади, сторонами которого являются доски. Пилить или ломать доски нельзя. Например, из трех досок с длинами 10, 11 и 12 можно построить забор, а из четырех досок с длинами 100, 1, 2 и 3 — нельзя.

У Деда нашлось целых  $n$  досок, поэтому они с Бабой задались вопросом: а сколько различных способов выбрать несколько досок из имеющихся, чтобы из них затем можно было построить забор? Способы считаются различными, если существует доска, которая используется в одном из них, но не используется в другом.

Пожилым людям надо помогать, так что вам не составит труда решить для них эту задачу! Количество способов может быть довольно большим, поэтому выведите остаток от деления этого количества на число  $10^9 + 7$ .

### Формат входного файла

В первой строке входного файла находится одно натуральное число  $n$  ( $1 \leq n \leq 4000$ ) — количество досок. Во второй строке дано  $n$  чисел  $l_i$  ( $1 \leq l_i \leq 4000$ ) — длины досок.

### Формат выходного файла

В выходной файл выведите одно число — количество способов выбрать доски для постройки забора, взятое по модулю  $10^9 + 7$ .

### Пример

<code>polygon.in</code>	<code>polygon.out</code>
3 10 11 12	1
4 100 1 2 3	0
4 5 5 5 5	5

### Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение  $n \leq 20$  и  $l_i \leq 100$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 33 балла.

Вторая группа состоит из тестов, для которых выполняется ограничение  $n \leq 100$  и  $l_i \leq 100$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 40 баллов.

Третья группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 27 баллов.

## Задача D. Карандаши

Имя входного файла: `pencils.in`  
Имя выходного файла: `pencils.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Новое увлечение Колобка — рисование. Он решил купить  $k$  наборов карандашей. Каждый набор состоит из одного или нескольких карандашей. Каждый карандаш имеет положительную длину, которая выражается целым числом миллиметров.

В магазине продаются  $n$  наборов карандашей. После того, как Колобок купит ровно  $k$  наборов, он придёт домой и сложит все карандаши в одну коробку. Колобок очень обрадуется, если разница в длине между наибольшим и наименьшим карандашами в этой коробке будет минимальна.

Поэтому он просит вас помочь ему: выберите из  $n$  наборов карандашей ровно  $k$  так, чтобы разница между максимальным и минимальным среди всех купленных карандашей была как можно меньше.

### Формат входного файла

В первой строке находятся два натуральных числа  $n, k$  ( $1 \leq n \leq 10^5, 1 \leq k \leq n$ ) — количество наборов карандашей, имеющихся в магазине, и количество наборов, необходимое Колобку.

В каждой из следующих  $n$  строк находится  $c_i$  ( $1 \leq c_i \leq 2 \cdot 10^5$ ) — количество карандашей в наборе. Далее, в этой же строке, следуют  $c_i$  натуральных чисел  $a_{ij}$  ( $1 \leq a_{ij} \leq 10^9$ ) — длины карандашей в  $i$ -м наборе.

Гарантируется, что сумма всех  $c_i$  не превосходит  $2 \cdot 10^5$ .

### Формат выходного файла

В единственной строке выведите наименьшую разницу между максимальным и минимальным купленными карандашами, которую можно достичь.

### Пример

<code>pencils.in</code>	<code>pencils.out</code>
3 2 3 1 3 4 3 5 1 2 1 4	3
5 3 3 2 1 3 2 4 1 3 4 2 4 4 3 2 3 3 2 5 6	3

### Система оценивания

Первая группа состоит из тестов, для которых выполняется ограничение  $n \leq 20$  и  $k > 1$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 31 балл.

Вторая группа состоит из тестов, для которых выполняются ограничения  $n \leq 2000$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 31 балл.

Третья группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 38 баллов.

## Задача Е. Рассадка зверей

Имя входного файла: `seating.in`  
Имя выходного файла: `seating.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Сегодня Колобок созвал всех волков и лис к себе в гости на чаепитие. Чаепитие пройдет за круглым столом, за которым всего  $n$  мест. Колобок хочет рассадить зверей по-особенному — так, чтобы волки не сидели только с волками, а лисы только с лисами. Поэтому для каждого места он записал одно целое число — сколько лис должно сидеть на расстоянии не более  $d$  от этого места, включая это место.

Два места находятся на расстоянии не более  $d$ , если между ними встречаются не более  $d-1$  места при движении по или против часовой стрелки от одного к другому. Таким образом, для заданного места всего существует  $2d+1$  место, находящееся на расстоянии не более  $d$  от него.

Теперь он хочет придумать какую-нибудь рассадку зверей, удовлетворяющую этим ограничениям.

### Формат входного файла

В первой строке находятся два натуральных числа  $n, d$  ( $3 \leq n \leq 10^5, 3 \leq 2d+1 \leq n$ ) — количество мест за круглым столом и расстояние  $d$ .

В следующей строке находятся  $n$  неотрицательных целых чисел  $a_i$  ( $0 \leq a_i \leq 2d+1$ ) — количество лис на расстоянии не более  $d$  от этого места, включая это место. Информация о местах перечислена в порядке их следования по кругу.

### Формат выходного файла

Если решения не существует, выведите «NO», иначе в первой строке выведите «YES», а в следующей  $n$  чисел: 1 в том случае, если на этом месте сидит лиса, и 0, если на этом месте сидит волк. Если ответов несколько, разрешается вывести любой.

### Пример

<code>seating.in</code>	<code>seating.out</code>
5 1 2 2 1 2 2	YES 1 0 1 0 1
9 2 3 4 4 3 3 2 2 2 2	YES 1 0 1 1 1 0 0 0 1
6 1 3 3 3 3 3 1	NO

### Система оценивания

Первая группа состоит из тестов, для которых выполняются ограничения  $n \leq 18$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 20 баллов.

Вторая группа состоит из тестов, для которых выполняются ограничения  $n \leq 1000, d \leq 5$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 23 балла.

Третья группа состоит из тестов, для которых выполняются ограничения  $n \leq 2000$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 25 баллов.

Четвертая группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 32 балла.

## Задача F. Воздушные потоки

Имя входного файла: `minsumdiff.in`  
Имя выходного файла: `minsumdiff.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Колобок ушёл от бабушки и поехал путешествовать. Неожиданно для себя он забрёл в страну Ивэнлэнд. Первые трудности встали на его пути: Колобка и вход в страну отделял огромный ров с водой, которая, как известно, не очень хорошо влияет на нашего героя. К счастью, повсюду расположены воздушные потоки, которые могли поднимать того, кто на них встает, на определённую высоту. Страна не просто так названа Ивэнлэнд, поэтому все высоты, на которые могут поднять героя воздушные потоки — это чётные числа.

Представим воздушные потоки как массив  $h[1..n]$  из  $n$  натуральных чисел — высот потоков. Для каждого  $1 \leq i \leq n$  посчитаем  $G[i]$  — индекс ближайшего элемента слева, строго большего  $h[i]$ . Более формально,  $g[i] = \max\{j \mid j < i \text{ и } h[j] > h[i]\}$ . Если  $i = 1$  или до  $h[i]$  нет ни одного элемента больше него, то  $G[i]$  считается равным 0.

Колобок считает, что оптимальность расположения воздушных потоков определяется суммой

$$S(h) = \sum_{i=1}^n (i - G[i]).$$

Чем меньше сумма, тем расположение оптимальнее. Всё, что может сейчас сделать Колобок — это увеличить высоту одного из воздушных потоков не более чем на  $m$ . После этого действия высота потока должна остаться целым числом, но может, если необходимо, стать и нечётной.

Помогите Колобку сделать оптимальное изменение, которое позволит добиться, чтобы сумма  $S(h)$ , описанная выше, после проделанного действия была минимальна.

### Формат входного файла

В первой строке входного файла даны числа  $n, m$  ( $1 \leq n \leq 10^5, 1 \leq m \leq 10^9$ ) — количество воздушных потоков и максимальное значение, на которое можно увеличить высоту одного из них.

Во второй строке даны высоты воздушных потоков  $h[i]$  ( $1 \leq h[i] \leq 10^9$ ). Гарантируется, что все высоты — чётные числа.

### Формат выходного файла

В единственной строке выходного файла выведите одно целое число — минимальную искомую сумму.

### Примеры

<code>minsumdiff.in</code>	<code>minsumdiff.out</code>
3 100 4 2 6	4
3 2 4 2 6	5
3 10 2 2 2	4

### Пояснение к примерам

В первом примере Колобку выгодно, например, увеличить первый элемент массива на 3, сделав его равным 7.

Тогда  $h = [7, 2, 6]$ ,  $G[1] = 0$ ,  $G[2] = 1$  и  $G[3] = 1$ , сумма  $S(h) = (1-0) + (2-1) + (3-1) = 1+1+2 = 4$ . Добиться значения  $S(h) < 4$  невозможно.

Во втором примере  $m = 2$  и увеличить первый элемент так сильно не получится. После любого действия Колобка  $S(h)$  будет равно 5 или 6, пример действия, которое приводит к значению 5 — увеличить первый элемент на 1, получив массив  $[5, 2, 6]$ .

### Система оценивания

Первая группа состоит из тестов, для которых выполняются ограничения  $1 \leq n \leq 200$ ,  $m = 10^9$  и  $a_i \leq 10$  для всех  $i$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 19 баллов.

Вторая группа состоит из тестов, для которых выполняется ограничение  $1 \leq n \leq 50$  и  $m = 10^9$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 18 баллов.

Третья группа состоит из тестов, для которых выполняется ограничение  $1 \leq n \leq 2000$  и  $m = 10^9$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 23 балла.

Четвертая группа состоит из тестов, для которых выполняется ограничение  $1 \leq n \leq 10^5$  и  $m = 10^9$ . Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 21 балл.

Пятая группа состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 19 баллов.