



ЗАДАЧИ ОТОСЛАТЬ СТАТУС ПОЛОЖЕНИЕ ЗАПУСК

D. Построение массива

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Вам задана массив a длины n , состоящий из нулей. Вы выполняете n действий с этим массивом: в течение i -го действия происходит следующая последовательность операций:

- 1. Выбирается максимальный по длине подмассив (последовательный подотрезок), состоящий **только** из нулей, среди всех таких отрезков выбирается **самый левый**;
- 2. Пусть этот отрезок равен $[l; r]$. Если $r - l + 1$ нечетно (не делится на 2), то присваивается $a[\frac{l+r}{2}] := i$ (где i — номер текущего действия), иначе (если $r - l + 1$ четно) присваивается $a[\frac{l+r-1}{2}] := i$.

Рассмотрим массив a длины 5 (изначально $a = [0, 0, 0, 0, 0]$). Тогда он меняется следующим образом:

- 1. Сначала мы выбираем отрезок $[1; 5]$ и присваиваем $a[3] := 1$, таким образом a становится равен $[0, 0, 1, 0, 0]$;
- 2. затем мы выбираем отрезок $[1; 2]$ и присваиваем $a[1] := 2$, таким образом a становится равен $[2, 0, 1, 0, 0]$;
- 3. затем мы выбираем отрезок $[4; 5]$ и присваиваем $a[4] := 3$, таким образом a становится равен $[2, 0, 1, 3, 0]$;
- 4. затем мы выбираем отрезок $[2; 2]$ и присваиваем $a[2] := 4$, таким образом a становится равен $[2, 4, 1, 3, 0]$;
- 5. и наконец мы выбираем отрезок $[5; 5]$ и присваиваем $a[5] := 5$, таким образом a становится равен $[2, 4, 1, 3, 5]$.

Ваша задача — найти массив a длины n после выполнения всех n действий. **Заметьте, что ответ существует и единственен.**

Вам необходимо ответить на t независимых наборов тестовых данных.

Входные данные

Первая строка входных данных содержит одно целое число t ($1 \leq t \leq 10^4$) — количество наборов тестовых данных. Затем следуют t наборов тестовых данных.

Единственная строка набора тестовых данных содержит одно целое число n ($1 \leq n \leq 2 \cdot 10^5$) — длину a .

Гарантируется, что сумма n по всем наборам тестовых данных не превосходит $2 \cdot 10^5$ ($\sum n \leq 2 \cdot 10^5$).

Выходные данные

Для каждого набора тестовых данных выведите ответ — массив a длины n после выполнения n действий, описанных в условии задачи. **Заметьте, что ответ существует и единственен.**

Пример

входные данные	Скопировать
6 1 2 3 4 5 6	
выходные данные	Скопировать
1 1 2 2 1 3 3 1 2 4 2 4 1 3 5 3 4 1 5 2 6	

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

А. Отель Гильберта

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 256 мегабайт

Отель Гильберта это очень необычный отель, потому что количество комнат в нем бесконечно! Для каждого целого числа существует ровно одна комната с таким номером, **включая ноль и отрицательные числа**. Не менее странно то, что сейчас отель полностью заполнен, что означает, что в каждой комнате находится ровно один гость. Менеджер отеля, сам Давид Гильберт, решил переместить гостей, потому что у него есть предположение, что за счет этого образуются свободные места.

Для любого целого числа k и положительного целого числа n обозначим за $k \bmod n$ остаток при делении числа k на число n . Более формально, $r = k \bmod n$ это наименьшее неотрицательное целое число такое, что $k - r$ делится на n . Всегда выполнено, что $0 \leq k \bmod n \leq n - 1$. Например, $100 \bmod 12 = 4$ и $(-1337) \bmod 3 = 1$.

Процесс перемещения гостей выглядит следующим образом: есть массив, состоящий из n целых чисел a_0, a_1, \dots, a_{n-1} . Тогда для всех целых чисел k гость из комнаты с номером k перемещается в комнату с номером $k + a_{k \bmod n}$.

Определите, верно ли, что после этого процесса перемещения в каждой комнате по-прежнему находится ровно один гость. Это означает, что нет пустых комнат и комнат, в которых больше одного гостя.

Входные данные

Каждый тест состоит из нескольких тестовых случаев. Первая строка содержит единственное целое число t ($1 \leq t \leq 10^4$) — количество тестовых случаев. Следующие $2t$ строк содержат описания тестовых случаев.

В первой строке описания каждого тестового случая находится единственное целое число n ($1 \leq n \leq 2 \cdot 10^5$) — длина массива.

Во второй строке описания каждого тестового случая находятся n целых чисел a_0, a_1, \dots, a_{n-1} ($-10^9 \leq a_i \leq 10^9$).

Гарантируется, что сумма n по всем тестовым случаям не превосходит $2 \cdot 10^5$.

Выходные данные

Для каждого тестового случая выведите единственную строку, содержащую «YES», если в каждой комнате после перемещения находится ровно один гость, и «NO» иначе. Вы можете выводить каждый символ в любом регистре.

Пример

входные данные	Скопировать
<pre>6 1 14 2 1 -1 4 5 5 5 1 3 3 2 1 2 0 1 5 -239 -2 -100 -3 -11</pre>	
выходные данные	Скопировать
<pre>YES YES YES NO NO YES</pre>	

Примечание

В первом тестовом случае номер комнаты каждого гостя увеличился на 14, поэтому по-прежнему в каждой комнате находится ровно один гость.

Во втором тестовом случае гости в комнатах с четными номерами перемещаются в комнату с номером, на 1 большим исходного; гости в комнатах с нечетными номерами перемещаются в комнату с номером, на 1 меньшим исходного. Можно показать, что по-прежнему в каждой комнате находится ровно один гость.

В третьем тестовом случае каждый четвертый гость перемещается в комнату с номером, на 1 большим исходного, а остальные гости перемещаются в комнату с номером, на 5 большим. Можно показать, что по-прежнему в каждой комнате находится ровно один гость.

В четвертом тестовом случае гости, исходно находящиеся в комнатах 0 и 1, перемещаются в комнату с номером 3.

В пятом тестовом случае гости, находящиеся в комнатах 1 и 2, перемещаются в комнату с номером 2.

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов

Соревнования по программированию 2.0

Время на сервере: 12.02.2025 16:51:38^{UTC+5} (13).

Мобильная версия, переключиться на [десктопную](#).

[Privacy Policy](#)

При поддержке



ИТМО

[ЗАДАЧИ](#)
[ОТΟΣЛАТЬ](#)
[СТАТУС](#)
[ПОЛОЖЕНИЕ](#)
[ЗАПУСК](#)

С. Феникс и распределение

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

У Феникса есть строка s , состоящая из строчных букв латинского алфавита. Он хочет распределить все буквы своей строки по k **непустым** строкам a_1, a_2, \dots, a_k так, что каждая буква из s попадет ровно в одну из строк a_i . Строки a_i **не обязаны** быть подстроками s . Феникс может распределить буквы s и переупорядочить их внутри каждой строки a_i так как захочет.

Например, если $s = \text{baba}$ и $k = 2$, Феникс может распределить буквы своей строки множеством способов, в том числе:

- ba и ba
- a и abb
- ab и ab
- bb и aa

Однако получить такие варианты он не может:

- baa и ba
- b и ba
- baba и пустая строка (a_i должны быть непустыми)

Феникс хочет разделить свою строку s на k строк a_1, a_2, \dots, a_k так, чтобы **минимизировать** лексикографически максимальную строку среди них, т. е. минимизировать $\max(a_1, a_2, \dots, a_k)$. Помогите ему найти оптимальное распределение и выведите минимально возможное значение $\max(a_1, a_2, \dots, a_k)$.

Строка x лексикографически меньше, чем строка y , если либо x является префиксом y (и $x \neq y$), либо существует такой индекс i ($1 \leq i \leq \min(|x|, |y|)$), что $x_i < y_i$ и для всех j ($1 \leq j < i$) $x_j = y_j$. Здесь $|x|$ обозначает длину строки x .

Входные данные

Входные данные состоят из нескольких наборов. В первой строке задано целое число t ($1 \leq t \leq 1000$) — количество наборов входных данных. Каждый набор состоит из двух строк.

В первой строке каждого набора задано два целых числа n и k ($1 \leq k \leq n \leq 10^5$) — длина строки s и количество не пустых строк, в которые Феникс хочет распределить буквы s , соответственно.

Во второй строке каждого набора задана строка s длины n , состоящая из строчных латинских букв.

Гарантируется, что сумма n по всем наборам входных данных $\leq 10^5$.

Выходные данные

Выведите t ответов — по одному на набор входных данных; i -й ответ — минимально возможный $\max(a_1, a_2, \dots, a_k)$ в i -м наборе.

Пример

входные данные	Скопировать
<pre>6 4 2 baba 5 2 baacb 5 3 baacb 5 3 aaaaa 6 4 aaxzz 7 1 phoenix</pre>	
выходные данные	Скопировать
<pre>ab abb b aa x ehinorx</pre>	

Примечание

В первом наборе входных данных, одно из оптимальных решений — разбить `baba` на `ab` и `ab`.

Во втором наборе входных данных, одно из оптимальных решений — разбить `baacb` на `abbc` и `a`.

В третьем наборе, одно из оптимальных решений — разбить `baacb` на `ac`, `ab` и `b`.

В четвертом наборе, одно из оптимальных решений — разбить `aaaaa` на `aa`, `aa` и `a`.

В пятом наборе, одно из оптимальных решений — разбить `aaxxz` на `az`, `az`, `x` и `x`.

В шестом наборе, одно из оптимальных решений — разбить `phoenix` на `ehinorx`.

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 12.02.2025 16:51:46^{UTC+5} (13).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке

**ИТМО**

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

Е. Рекламное агентство

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

Маша работает в рекламном агентстве. В целях продвижения нового бренда она хочет заключить договор с некоторыми блогерами. Всего у Маши есть контакты n разных блогеров. Блогер с номером i имеет a_i подписчиков.

Так как бюджет у Маши ограничен, она может заключить договор только с k разными блогерами. Конечно же, Маша хочет, чтобы ее рекламу увидело как можно больше людей. Поэтому она должна нанять блогеров с максимальным суммарным количеством подписчиков.

Помогите ей, найдите количество способов выбрать k блогеров так, чтобы суммарное количество их подписчиков было максимально. Два способа считаются разными, если в первом способе есть хотя бы один блогер, которого нет во втором способе. Маша считает, что у всех блогеров разная аудитория (то есть не существует подписчика, который был бы подписан на двух разных блогеров).

Например, если $n = 4$, $k = 3$, $a = [1, 3, 1, 2]$, то у Маши есть два способа выбрать 3-х блогеров с максимальным суммарным количеством подписчиков:

- заключить договор с блогерами с номерами 1, 2 и 4. В таком случае количество подписчиков будет равно $a_1 + a_2 + a_4 = 6$.
- заключить договор с блогерами с номерами 2, 3 и 4. В таком случае количество подписчиков будет равно $a_2 + a_3 + a_4 = 6$.

Так как ответ может быть достаточно большим, **выведите его по модулю $10^9 + 7$** .

Входные данные

В первой строке записано одно целое число t ($1 \leq t \leq 1000$) — количество наборов входных данных. Далее следуют t наборов входных данных.

В первой строке каждого набора входных данных содержатся два целых числа n и k ($1 \leq k \leq n \leq 1000$) — количество блогеров и со сколькими из них можно подписать договор.

Во второй строке каждого набора входных данных содержится n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) — количества подписчиков у блогеров.

Гарантируется, что сумма n по всем наборам входных данных не превосходит 1000.

Выходные данные

Для каждого набора входных данных в отдельной строке выведите одно целое число — количество способов выбрать k блогеров так, чтобы суммарное количество их подписчиков было максимально.

Пример

входные данные	Скопировать
3 4 3 1 3 1 2 4 2 1 1 1 1 2 1 1 2	
выходные данные	Скопировать
2 6 1	

Примечание

Первый набор входных данных разобран в условии.

Во втором наборе входных данных следующие варианты являются подходящими:

- заключить договор с блогерами с номерами 1 и 2. В таком случае количество подписчиков будет равно $a_1 + a_2 = 2$;
- заключить договор с блогерами с номерами 1 и 3. В таком случае количество подписчиков будет равно $a_1 + a_3 = 2$;
- заключить договор с блогерами с номерами 1 и 4. В таком случае количество подписчиков будет равно $a_1 + a_4 = 2$;

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

E1. Чтение книг (простая версия)

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

Простая и сложная версии на самом деле являются разными задачами, поэтому прочитайте условия обеих задач полностью и внимательно.

Летние каникулы начались, поэтому Алиса и Боб хотят играть и веселиться, но... Их мама не согласна с этим. Она говорит, что они должны прочитать какое-то количество книг перед всеми развлечениями. Алиса и Боб прочитают каждую книгу **вместе**, чтобы быстрее закончить это задание.

В семейной библиотеке есть n книг. i -я книга характеризуется тремя целыми числами: t_i — количество времени, которое Алиса и Боб должны потратить, чтобы прочитать ее, a_i (равное 1, если Алисе нравится i -я книга, и 0, если не нравится), и b_i (равное 1, если Бобу нравится i -я книга, и 0, если не нравится).

Поэтому им нужно выбрать какие-то книги из имеющихся n книг таким образом, что:

- Алисе нравятся **не менее** k книг из выбранного множества и Бобу нравятся **не менее** k книг из выбранного множества;
- общее время, затраченное на прочтение этих книг **минимизировано** (ведь они дети и хотят начать играть и веселиться как можно скорее).

Множество, которое они выбирают, **одинаковое** и для Алисы и для Боба (они читают одни и те же книги), и они читают все книги **вместе**, таким образом, суммарное время чтения равно сумме t_i по всем книгам, которые находятся в выбранном множестве.

Ваша задача — помочь им и найти любое подходящее множество книг или определить, что такое множество найти невозможно.

Входные данные

Первая строка теста содержит два целых числа n и k ($1 \leq k \leq n \leq 2 \cdot 10^5$).

Следующие n строк содержат описания книг, по одному описанию в строке: i -я строка содержит три целых числа t_i , a_i и b_i ($1 \leq t_i \leq 10^4$, $0 \leq a_i, b_i \leq 1$), где:

- t_i — количество времени, необходимое для прочтения i -й книги;
- a_i , равное 1, если Алисе нравится i -я книга, и 0 в обратном случае;
- b_i , равное 1, если Бобу нравится i -я книга, и 0 в обратном случае.

Выходные данные

Если подходящего решения не существует, выведите число -1. Иначе выведите целое число T — минимальное суммарное время, необходимое для прочтения подходящего множества книг.

Примеры

входные данные	Скопировать
<pre> 8 4 7 1 1 2 1 1 4 0 1 8 1 1 1 0 1 1 1 1 1 0 1 3 0 0 </pre>	
выходные данные	Скопировать
<pre> 18 </pre>	
входные данные	Скопировать
<pre> 5 2 6 0 0 9 0 0 1 0 1 2 1 1 5 1 0 </pre>	
выходные данные	Скопировать

8	
<div>входные данные</div> <div>5 3 3 0 0 2 1 0 3 1 0 5 0 1 3 0 1</div>	<div>Скопировать</div>
<div>выходные данные</div> <div>-1</div>	<div>Скопировать</div>

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 12.02.2025 16:51:50^{UTC+5} (13).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке



[ЗАДАЧИ](#) [ОТΟΣЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

А. Мафия

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

Как-то раз собрались n друзей, чтобы сыграть в «Мафию». В каждом раунде игры «Мафия» должен быть назначен ведущий (кто-то один из ребят), остальные $n - 1$ человек принимают участие в игре. Для каждого человека известно, в каком количестве раундов он хочет принять участие как игрок, а не как ведущий: i -ый человек хочет сыграть a_i раундов. Какое минимальное количество раундов игры «Мафия» нужно сыграть, чтобы каждый человек сыграл как минимум столько раундов, сколько хочет?

Входные данные

В первой строке записано целое число n ($3 \leq n \leq 10^5$). Во второй строке через пробел записаны n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — i -ое число в списке обозначает количество раундов, которое хочет сыграть i -ый человек.

Выходные данные

В единственную строку выведите целое число — минимальное количество раундов игры, которое нужно сыграть, чтобы i -ый человек сыграл как минимум a_i раундов.

Пожалуйста, не используйте спецификатор `%lld` для чтения или записи 64-битных чисел на C++. Рекомендуется использовать потоки `cin`, `cout` или спецификатор `%I64d`.

Примеры

входные данные	Скопировать
3 3 2 2	
выходные данные	Скопировать
4	
входные данные	Скопировать
4 2 2 2 2	
выходные данные	Скопировать
3	

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 12.02.2025 16:51:57^{UTC+5} (13).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке

**ИТМО**

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Уничтожение массива

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 256 мегабайт

Вы нашли бесполезный массив целых чисел a длины $2n$. Так как он оказался Вам не нужен, Вы решили выкинуть все элементы a .

Это было бы просто, но оказалось, что выкидывать числа надо, соблюдая следующие правила:

1. Сначала Вы выбираете натуральное число x .
2. Затем Вы n раз повторите следующую операцию:
 - выбрать два числа с суммой x
 - выкинуть их из a и заменить x на максимальное из этих двух чисел.

Например, если изначально $a = [3, 5, 1, 2]$, то Вы можете выбрать $x = 6$ и выкинуть второе и третье число в a , сумма которых равна $5 + 1 = 6$. После этого x станет равно 5, а в массиве останутся числа 3 и 2, которые Вы можете выкинуть на следующей операции.

Обратите внимание, что Вы выбираете x только в начале и не можете изменить его между операциями.

Напишите программу, чтобы определить, как выбросить все элементы a .

Входные данные

В первой строке задано одно целое число t ($1 \leq t \leq 1000$) — количество наборов входных данных. Далее следуют описания самих наборов.

В первой строке каждого набора задано одно целое число n ($1 \leq n \leq 1000$).

Во второй строке каждого набора задано $2n$ натуральных чисел a_1, a_2, \dots, a_{2n} ($1 \leq a_i \leq 10^6$) — изначальный массив a .

Гарантируется, что сумма n по всем наборам входных данных не превосходит 1000.

Выходные данные

Для каждого набора входных данных в первой строке выведите YES если возможно выкинуть все элементы массива. Иначе выведите NO.

Если Вы вывели YES в следующей строке выведите значение x , которое Вы выбираете в начале. В следующих n строках выведите по два числа — числа, которые Вы выбираете на очередной операции.

Пример

входные данные	Скопировать
4 2 3 5 1 2 3 1 1 8 8 64 64 2 1 1 2 4 5 1 2 3 4 5 6 7 14 3 11	
выходные данные	Скопировать
YES 6 1 5 2 3 NO NO YES 21 14 7 3 11 5 6 2 4 3 1	

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D. Цветные прямоугольники

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Даны три мультимножества пар цветных палок:

- R пар красных палок, у первой пары длины равны r_1 , у второй пары длины равны r_2, \dots , у R -й пары длины равны r_R ;
- G пар зеленых палок, у первой пары длины равны g_1 , у второй пары длины равны g_2, \dots , у G -й пары длины равны g_G ;
- B пар синих палок, у первой пары длины равны b_1 , у второй пары длины равны b_2, \dots , у B -й пары длины равны b_B .

Вы собираете прямоугольники из этих пар палок следующим образом:

1. взять пару палок одного цвета;
2. взять пару палок другого цвета, отличного от первого;
3. прибавить площадь полученного прямоугольника к суммарной площади.

В итоге получатся такие прямоугольники, что противоположные стороны у них одного цвета, а соседние стороны различных цветов.

Каждая пара палок может быть использована не более одного раза, некоторые пары можно не использовать. Не разрешается разбивать пару палок на отдельные палки.

Какую максимальную суммарную площадь можно получить?

Входные данные

В первой строке записаны три целых числа R, G, B ($1 \leq R, G, B \leq 200$) — количество пар красных палок, количество пар зеленых палок и количество пар синих палок.

Во второй строке записаны R целых чисел r_1, r_2, \dots, r_R ($1 \leq r_i \leq 2000$) — длины палок в каждой паре красных палок.

В третьей строке записаны G целых чисел g_1, g_2, \dots, g_G ($1 \leq g_i \leq 2000$) — длины палок в каждой паре зеленых палок.

В четвертой строке записаны B целых чисел b_1, b_2, \dots, b_B ($1 \leq b_i \leq 2000$) — длины палок в каждой паре синих палок.

Выходные данные

Выведите максимально возможную суммарную площадь построенных прямоугольников.

Примеры

входные данные	Скопировать
1 1 1 3 5 4	
выходные данные	Скопировать
20	
входные данные	Скопировать
2 1 3 9 5 1 2 8 5	
выходные данные	Скопировать
99	
входные данные	Скопировать
10 1 1 11 7 20 15 19 14 2 4 13 14 8 11	
выходные данные	Скопировать
372	

Примечание

В первом примере можно построить один из следующих прямоугольников: красный и зеленый со сторонами 3 и 5, красный и синий со сторонами 3 и 4 и зеленый и синий со сторонами 5 и 4. Лучшая площадь из них равна $4 \times 5 = 20$.

Во втором примере лучшие прямоугольники: красный/синий 9×8 , красный/синий 5×5 , зеленый/синий 2×1 . Суммарная площадь равна $72 + 25 + 2 = 99$.

В третьем примере лучшие прямоугольники: красный/зеленый 19×8 и красный/синий 20×11 . Суммарная площадь равна $152 + 220 = 372$. Обратите внимание, что больше прямоугольников нельзя построить, потому что не разрешается иметь обе взятые пары одного цвета.

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 12.02.2025 16:52:12^{UTC+5} (13).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке

**ИТМО**

[ЗАДАЧИ](#) [ОТΟΣЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D2. Приравнивание делением (сложная версия)

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

Единственное отличие между легкой и сложной версиями — количество элементов в массиве.

Вам задан массив a , состоящий из n целых чисел. За один ход вы можете выбрать любое a_i и разделить его на 2 с округлением вниз (иными словами, за один ход вы можете присвоить $a_i := \lfloor \frac{a_i}{2} \rfloor$).

Вы можете совершать эту операцию **любое** (возможно, нулевое) количество раз с **любым** a_i .

Ваша задача — посчитать минимально возможное количество операций, необходимое для того, чтобы получить хотя бы k равных чисел в массиве.

Не забудьте, что допустимо иметь $a_i = 0$ после каких-то операций, таким образом, ответ всегда существует.

Входные данные

Первая строка входных данных содержит два целых числа n и k ($1 \leq k \leq n \leq 2 \cdot 10^5$) — количество элементов в массиве и необходимое количество равных элементов.

Вторая строка входных данных содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 2 \cdot 10^5$), где a_i равно i -му элементу a .

Выходные данные

Выведите одно целое число — минимально возможное количество операций, необходимое для того, чтобы получить хотя бы k равных элементов в массиве.

Примеры

входные данные	Скопировать
5 3 1 2 2 4 5	
выходные данные	Скопировать
1	
входные данные	Скопировать
5 3 1 2 3 4 5	
выходные данные	Скопировать
2	
входные данные	Скопировать
5 3 1 2 3 3 3	
выходные данные	Скопировать
0	



[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

Е. Перестановка строк и столбцов

ограничение по времени на тест: 3 секунды

ограничение по памяти на тест: 256 мегабайт

У вас была матрица a размера n на m , содержащая перестановку целых чисел от 1 до $n \cdot m$.

Перестановкой из n целых чисел называется массив, содержащий все числа от 1 до n ровно один раз. Например, массивы $[1]$, $[2, 1, 3]$, $[5, 4, 3, 2, 1]$ являются перестановками, а массивы $[1, 1]$, $[100]$, $[1, 2, 4, 5]$ — нет.

Матрица содержит перестановку, если при выписывании всех её элементов полученный массив является перестановкой.

Матрицы $[[1, 2], [3, 4]]$, $[[1]]$, $[[1, 5, 3], [2, 6, 4]]$ содержат перестановки, а матрицы $[[2]]$, $[[1, 1], [2, 2]]$, $[[1, 2], [100, 200]]$ — нет.

За одну операцию вы можете совершить одно из двух следующих действий:

- выбрать столбец c и столбец d ($1 \leq c, d \leq m$, $c \neq d$) и поменять эти столбцы местами;
- выбрать строку c и строку d ($1 \leq c, d \leq n$, $c \neq d$) и поменять эти строки местами.

Вы можете совершить любое количество операций.

Вам даны исходная матрица a и матрица b . Ваша задача — определить, можно ли с помощью данных операций сделать из матрицы a матрицу b .

Входные данные

Первая строка содержит целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных. Далее следуют описания наборов входных данных.

Первая строка описания каждого набора входных данных содержит 2 целых числа n и m ($1 \leq n, m \leq n \cdot m \leq 2 \cdot 10^5$) — размеры матрицы.

Следующие n строк содержат по m целых чисел a_{ij} каждая ($1 \leq a_{ij} \leq n \cdot m$). Гарантируется, что матрица a является перестановкой.

Следующие n строк содержат по m целых чисел b_{ij} каждая ($1 \leq b_{ij} \leq n \cdot m$). Гарантируется, что матрица b является перестановкой.

Гарантируется, что сумма значений $n \cdot m$ по всем наборам входных данных не превосходит $2 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите «YES», если вторая матрица может быть получена из первой, и «NO» в противном случае.

Вы можете выводить каждую букву в любом регистре (строчную или заглавную). Например, строки «yEs», «yes», «Yes» и «YES» будут приняты как положительный ответ.

Пример

входные данные	Скопировать
<pre> 7 1 1 1 1 2 2 1 2 3 4 4 3 2 1 2 2 1 2 3 4 4 3 1 2 3 4 1 5 9 6 12 10 4 8 7 11 3 2 1 5 9 6 12 10 4 8 7 11 3 2 </pre>	

```
3 3
1 5 9
6 4 2
3 8 7
9 5 1
2 4 6
7 8 3
2 3
1 2 6
5 4 3
6 1 2
3 4 5
1 5
5 1 2 3 4
4 2 5 1 3
```

выходные данные[Скопировать](#)

```
YES
YES
NO
YES
YES
NO
YES
```

Примечание

Во втором примере исходная матрица выглядит так:

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$$

При обмене строк 1 и 2 местами она станет такой:

$$\begin{pmatrix} 3 & 4 \\ 1 & 2 \end{pmatrix}$$

При обмене столбцов 1 и 2 местами она станет равна матрице b :

$$\begin{pmatrix} 4 & 3 \\ 2 & 1 \end{pmatrix}$$

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 12.02.2025 16:52:27^{UTC+5} (13).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке

**ИТМО**

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

В. Эквивалентные строки

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

Сегодня на спецкурсе «Строки» Геральд узнал новое определение эквивалентности строк. Две строки a и b равной длины называются *эквивалентными* в одном из двух случаев:

- Они равны.
- Если разделить строку a на две одинаковые по длине половины a_1 и a_2 , а строку b — на две одинаковые по длине половины b_1 и b_2 , то окажется, что верно одно из двух:

а. a_1 эквивалентно b_1 , а a_2 эквивалентно b_2 б. a_1 эквивалентно b_2 , а a_2 эквивалентно b_1

В качестве домашнего задания преподаватель выдал ученикам две строки и попросил определить, эквиваленты ли они.

Геральд уже сделал это домашнее задание. Сделайте и вы!

Входные данные

В первых двух строках входных данных даны две строки, которые выдал преподаватель. Каждая из них имеет длину от 1 до 200 000 и состоит из строчных букв английского алфавита. Строки имеют одинаковую длину.

Выходные данные

Выведите «YES» (без кавычек), если эти две строки эквивалентны, и «NO» (без кавычек) в противном случае.

Примеры

входные данные	Скопировать
aaba abaa	
выходные данные	Скопировать
YES	

входные данные	Скопировать
aabb abab	
выходные данные	Скопировать
NO	

Примечание

В первом примере первую строку можно разделить на строки "aa" и "ba", вторую — на строки "ab" и "aa". "aa" эквивалентно "aa", "ab" эквивалентно "ba", так как "ab" = "a" + "b", "ba" = "b" + "a".

Во втором примере первую строку можно разделить на строки "aa" и "bb", которые эквивалентны только сами себе. Поэтому строка "aabb" эквивалентна только сама себе и строке "bbaa".



[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Пинки Пай поедает пирожные

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Пинки Пай купила пакет пирожных с разными начинками! Но оказалось, что не все пирожные различаются начинкой между собой, то есть в пакете есть какие-то пирожные с одинаковой начинкой.

Пинки Пай ест пирожные по одному подряд. Она любит веселиться, поэтому решила не просто поедать пирожные, а стараться не есть пирожные с одной начинкой слишком часто. Для этого она хочет, чтобы минимальное расстояние между съеденными пирожными с одинаковой начинкой было как можно больше. При этом расстоянием между двумя пирожными Пинки Пай решила назвать количество съеденных пирожных между ними.

Пинки Пай может поедать пирожные в любом порядке. Ей не терпится скорее съесть все пирожные, поэтому она просит вас помочь посчитать наибольшее минимальное расстояние между съеденными пирожными с одинаковой начинкой среди всех возможных порядков поедания!

Пинки Пай собирается покупать еще пакеты с пирожными, поэтому просит вас решить задачу для нескольких пакетов!

Входные данные

Первая строка содержит одно целое число T ($1 \leq T \leq 100$) — количество пакетов, для которых нужно решить задачу.

Первая строка описания пакета содержит одно целое число n ($2 \leq n \leq 10^5$) — количество пирожных в нем. Вторая строка описания пакета содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) — информация о начинках пирожных: одинаковые начинки обозначены одинаковым числом, разные начинки — разным. Гарантируется, что в каждом пакете есть как минимум два пирожных с одинаковой начинкой.

Гарантируется, что сумма значений n по всем пакетам не превосходит 10^5 .

Выходные данные

Для каждого пакета в отдельной выведите одно целое число — наибольшее минимальное расстояние между съеденными пирожными с одинаковой начинкой среди всех возможных порядков поедания пирожных из этого пакета.

Пример

входные данные	Скопировать
4 7 1 7 1 6 4 4 6 8 1 1 4 6 4 6 4 7 3 3 3 3 6 2 5 2 3 1 4	
выходные данные	Скопировать
3 2 0 4	

Примечание

Для первого пакета Пинки Пай может есть пирожные в следующем порядке (по начинкам): 1, 6, 4, 7, 1, 6, 4 (таким образом, минимальное расстояние будет равно 3).

Для второго пакета Пинки Пай может есть пирожные в следующем порядке (по начинкам): 1, 4, 6, 7, 4, 1, 6, 4 (таким образом, минимальное расстояние будет равно 2).

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D. Загадочная посылка

ограничение по времени на тест: 1 second

ограничение по памяти на тест: 64 megabytes

Петя решил поздравить своего друга из Австралии с предстоящим днем рождения, отправив ему открытку. Чтобы сделать свой сюрприз более загадочным, он решил создать цепь. Цепью назовем такую последовательность конвертов $A = \{a_1, a_2, \dots, a_n\}$, что ширина и высота i -го конверта строго больше ширины и высоты $(i - 1)$ -го соответственно, а размером цепи — количество конвертов в ней. Из имеющихся у Пети конвертов он хочет создать наибольшую по размеру цепь, в которую можно было бы вложить открытку. Открытку можно вложить в цепь, если ширина и высота открытки меньше ширины и высоты меньшего конверта в цепи соответственно. Поворачивать конверты или открытку запрещено. Поскольку у Пети очень много конвертов и очень мало времени, то эта нелегкая задача возлагается на Вас.

Входные данные

Первая строка содержит целые числа n, w, h ($1 \leq n \leq 5000, 1 \leq w, h \leq 10^6$) — количество конвертов, имеющихся у Пети в распоряжении, ширину и высоту открытки соответственно. Далее содержится n строк, в каждой из которых находится два целых числа w_i и h_i — ширина и высота i -го конверта ($1 \leq w_i, h_i \leq 10^6$).

Выходные данные

В первую строку выведите размер максимальной цепи. Во вторую строку выведите через пробелы номера конвертов, образующих такую цепь, начиная с номера наименьшего конверта. Помните, что в наименьший конверт должна поместиться открытка. Если существует несколько цепей максимального размера, выведите любую.

Если открытка не помещается ни в один конверт, то выведите единственную строку, содержащую число 0.

Примеры

входные данные	Скопировать
2 1 1 2 2 2 2	
выходные данные	Скопировать
1 1	

входные данные	Скопировать
3 3 3 5 4 12 11 9 8	
выходные данные	Скопировать
3 1 3 2	



[ЗАДАЧИ](#) [ОТΟΣЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D. Задача Пашмака и Пармиды

ограничение по времени на тест: 3 секунды

ограничение по памяти на тест: 256 мегабайт

Пармида — умная девушка, в этом году она хочет принять участие в олимпиаде. Конечно, она хочет, чтобы ее парень тоже был умным! Девушка приготовила следующую тестовую задачу для Пашмака.

Задана последовательность a , состоящая из n целых чисел: a_1, a_2, \dots, a_n . Обозначим за $f(l, r, x)$ количество таких индексов k , что: $l \leq k \leq r$ и $x = a_k$. Требуется посчитать количество таких пар индексов i, j ($1 \leq i < j \leq n$), что $f(1, i, a_i) > f(j, n, a_j)$.

Помогите Пашмаку с этой задачей.

Входные данные

В первой строке записано целое число n ($1 \leq n \leq 10^6$). Во второй строке записано n целых чисел через пробел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Выходные данные

Выведите единственное целое число — ответ на задачу.

Примеры

входные данные	Скопировать
7 1 2 1 1 2 2 1	
выходные данные	Скопировать
8	
входные данные	Скопировать
3 1 1 1	
выходные данные	Скопировать
1	
входные данные	Скопировать
5 1 2 3 4 5	
выходные данные	Скопировать
0	

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 12.02.2025 16:52:52^{UTC+5} (13).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке

**ИТМО**

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D. Ловушки

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 256 мегабайт

Как известно, в мире всегда есть недобросовестные люди. К сожалению, иногда такие люди делают ревью, и самый умный человек на земле (естественно, Тётя Люсине) придумала хитрейший способ с ними бороться: ханипоты! Но Тётя Люсине пошла ещё дальше: она замаскировала все ханипоты под «ловушки»! Теперь вам, как самому лучшему ревьюверу, нужно через них пробраться.

Вам нужно преодолеть n ловушек, пронумерованных от 1 до n . Вы будете проходить через ловушки по очереди. i -я ловушка наносит вам a_i базового урона.

Вместо того, чтобы проходить через ловушку, вы можете перепрыгнуть ее. Всего вы можете перепрыгнуть не более k ловушек. В случае, если вы перепрыгиваете ловушку, вы не получаете от неё урона. Однако если вы перепрыгиваете какую-то ловушку, то это увеличивает урон каждой следующей ловушки на 1 (это бонусный урон).

Обратите внимание, что если вы перепрыгиваете ловушку, то вы не получаете от неё урона (ни базовый, ни бонусный), а что также бонусные уроны от прыжков складываются. Например, если вы проходите через i -ю ловушку с базовым уроном a_i , а до этого вы перепрыгнули 3 другие ловушки, то вы получите от неё $(a_i + 3)$ урона.

Определите, какое минимальный суммарный урон вы можете получить, если разрешается перепрыгнуть не больше k любых ловушек.

Входные данные

Каждый тест состоит из нескольких наборов входных данных. В первой строке находится единственное целое число t ($1 \leq t \leq 100$) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит два целых числа n и k ($1 \leq n \leq 2 \cdot 10^5$, $1 \leq k \leq n$) — количество ловушек и максимальное количество ловушек, которые вы можете перепрыгнуть.

Следующая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) — базовые уроны ловушек.

Гарантируется, что сумма n по всем наборам входных данных не превышает $2 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите одно число — минимальный суммарный урон, который можно получить, если можно перепрыгнуть не больше k любых ловушек.

Пример

входные данные	Скопировать
5 4 4 8 7 1 4 4 1 5 10 11 5 7 5 8 2 5 15 11 2 8 6 3 1 2 3 4 5 6 1 1 7	
выходные данные	Скопировать
0 21 9 6 0	

Примечание

В первом наборе входных данных можно перепрыгнуть все ловушки и получить 0 урона.

Во втором наборе входных данных есть 5 способов перепрыгнуть ловушки:

1. Не перепрыгивать никакие ловушки.
Суммарный урон: $5 + 10 + 11 + 5 = 31$.

2. Перепрыгнуть 1-ю ловушку.

Суммарный урон: $\underline{0} + (10 + 1) + (11 + 1) + (5 + 1) = 29$.

3. Перепрыгнуть 2-ю ловушку.

Суммарный урон: $5 + \underline{0} + (11 + 1) + (5 + 1) = 23$.

4. Перепрыгнуть 3-ю ловушку.

Суммарный урон: $5 + 10 + \underline{0} + (5 + 1) = 21$.

5. Перепрыгнуть 4-ю ловушку.

Суммарный урон: $5 + 10 + 11 + \underline{0} = 26$.

Чтобы получить минимальный урон, необходимо перепрыгнуть 3-ю ловушку, тогда ответ будет 21.

В третьем наборе входных данных оптимально перепрыгнуть через ловушки с номерами 1, 3, 4, 5, 7:

Суммарный урон: $0 + (2 + 1) + 0 + 0 + 0 + (2 + 4) + 0 = 9$.

[Codeforces](#) (c) Copyright 2010-2025 Михаил Мирзаянов

Соревнования по программированию 2.0

Время на сервере: 12.02.2025 16:52:58^{UTC+5} (13).

Мобильная версия, переключиться на [десктопную](#).

[Privacy Policy](#)

При поддержке



ІТМО