

Задача А. Долгая игра

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

Леша очень любит играть с кубиками. Иногда он строит из них стены, иногда рисует на них буквы, а затем составляет слова, но сегодня особенный день, ведь к Леше в гости пришла Полина, поэтому Леша придумал новую игру.

Перед началом игры Леша достал из шкафа N кубиков, после чего на каждом кубике написал число от 1 до N . Разумеется, ни на каких двух кубиках Леша не написал одно и то же число.

После успешной подготовки к игре Леша надел на глаза повязку, благодаря которой он больше не видит кубики и числа, написанные на них. Далее Леша и Полина по очереди делают ходы, причем первым ходит Леша, ведь кубики принадлежат ему!

Своим первым ходом Леша случайно равномерно перемешивает все лежащие перед ним кубики и выкладывает их в ряд. После этого Полина во время своего хода сообщает ему, какие кубики лежат на своем месте. Будем говорить, что кубик с написанным на нем числом A лежит на своем месте, если слева от него находятся ровно $A - 1$ кубиков. Во время всех следующих ходов Леша не будет трогать кубики, которые лежат на своем месте.

Далее Леша снова перемешивает все кубики, которые лежат не на своих местах, а Полина сообщает ему, какие из них оказались на своем месте. Игра продолжается до тех пор, пока все кубики не окажутся на своих местах.

Перед началом игры Леше стало интересно, насколько много ходов ему придется сделать. Посчитайте математическое ожидание количества ходов Леша, с учетом первого хода.

Формат входных данных

В единственной строке записано целое число N ($1 \leq N \leq 10^6$) — количество кубиков, которые есть у Леша.

Формат выходных данных

Нетрудно показать, что ответ можно представить в виде несократимой дроби $\frac{P}{Q}$.

В качестве ответа выведите $P \cdot Q^{-1} \pmod{998\,244\,353}$.

Примеры

стандартный ввод	стандартный вывод
1	1
2	2

Замечание

В первом примере у Леша есть всего один кубик, который после его первого хода с вероятностью 1 окажется на своем месте. Значит, математическое ожидание количества ходов Леша равно 1.

Во втором примере у Леша есть два кубика. С вероятностью $\frac{1}{2}$ после первого же хода они оба окажутся на своих местах и игра закончится, и с вероятностью $\frac{1}{2}$ ни один из кубиков не окажется на своем месте, и Леша окажется в том же положении, что и в начале игры. Тогда можно понять, что вероятность того, что игра закончится ровно через k ходов, равна $\frac{1}{2^k}$. Можно показать, что математическое ожидание этой случайной величины равно 2.

Напомним, что математическое ожидание случайной величины X равно

$$\sum_i x_i \cdot P(X = x_i)$$

Здесь $P(X = x_i)$ — вероятность того, что значение случайной величины равно x_i , а x_i — все возможные значения случайной величины.

Также напомним, что $Q \cdot Q^{-1} \equiv 1 \pmod{998\,244\,353}$.

Задача В. Липецкое метро

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

Шел 2120-й год, и в Липецке решили воплотить в жизнь проект вековой давности — воздушное метро! Для разработки схемы метро пригласили лучших архитекторов и инженеров со всей страны, и спустя месяц работы проект уже был готов.

Было решено построить N станций, каждая из которых не будет похожа на остальные и станет настоящим произведением искусства. Затем было решено построить на некоторых станциях по одному перегону до некоторой другой станции: так, если $p_i \neq -1$, то перегон соединяет станции i и p_i . Конечно, поезда по перегону могут ездить в обоих направлениях: как от станции i к станции p_i , так и наоборот.

Вам в руки случайным образом попала карта перегонов будущего метро, и вам стало очень интересно: можно ли, начав путь на какой-либо станции метро, объехать все станции, посетив каждую ровно один раз. Обратите внимание, что начать и закончить маршрут можно на любых станциях.

Формат входных данных

В первой строке записано число N ($1 \leq N \leq 2 \cdot 10^5$) — количество станций.

Во второй строке записаны N чисел p_i ($1 \leq p_i \leq N$, либо $p_i = -1$), где $p_i \neq -1$ означает, что станции с номерами i и p_i соединены перегонном.

Гарантируется, что $p_i \neq i$. Также гарантируется, что если $p_i = j$, то $p_j \neq i$.

Формат выходных данных

В первой строке выведите «YES» (без кавычек), если можно построить описанный маршрут, либо «NO» (без кавычек) в противном случае.

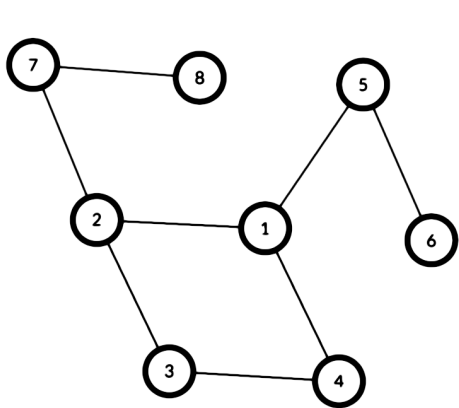
В случае, если построить необходимый маршрут возможно, во второй строке выведите N чисел — номера станций метро в порядке их посещения.

Примеры

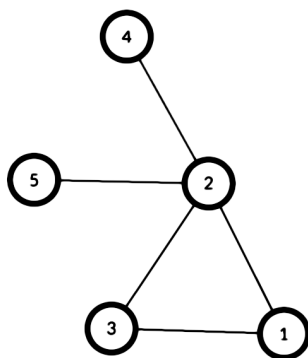
стандартный ввод	стандартный вывод
8 2 3 4 1 1 5 2 7	YES 6 5 1 4 3 2 7 8
5 2 3 1 2 2	NO
3 2 3 1	YES 1 3 2
3 2 3 -1	YES 1 2 3
4 2 3 -1 2	NO
3 -1 -1 -1	NO

Замечание

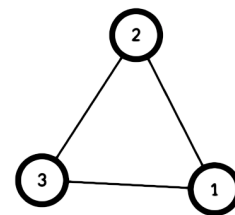
Ниже приведены рисунки схем Липецкого метро для каждого из шести примеров.



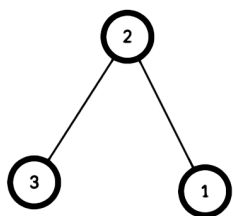
Пример 1



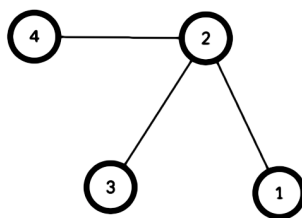
Пример 2



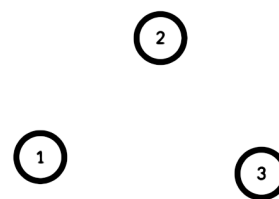
Пример 3



Пример 4



Пример 5



Пример 6

Задача С. Начинающий маг

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

Марк недавно решил попробовать нового персонажа в своей любимой RPG-игре — мага. Но перед тем, как начать играть, необходимо снарядить своего персонажа. Марк выбрал отличный набор предметов, и осталось самое главное — определиться с набором магических навыков.

Всего в любимой игре Марка есть N магических навыков. Пронумеруем навыки числами от 1 до N . Основной навык — возможность сотворения заклинаний — имеет номер 1 и автоматически имеется изначально у каждого мага. Чтобы изучить любой другой навык, нужно потратить одно очко умений. К слову, у Марка есть K таких очков, и он может потратить их полностью или частично для создания лучшего персонажа.

Однако на изучение навыков наложено еще одно ограничение. Для изучения сильных навыков требуется другие, базовые. Чтобы изучить не основной навык с номером i необходимо уже знать навык p_i . При этом $p_i < i$.

Для навыка i определим уровень навыка l_i следующим образом:

- $l_1 = 0$;
- $l_i = l_{p_i} + 1$

Марк подумал, что не хочет, чтобы у его мага была только одна сильная сторона. Поэтому он решил, что ему нужно изучить два навыка одного и того же уровня, при этом этот уровень должен быть максимальным. И с проблемой выбора этих двух навыков он решил обратиться к вам. Помогите ему!

Формат входных данных

В первой строке заданы два числа N и K ($1 \leq N, K \leq 2 \cdot 10^5$) — число магических навыков и количество доступных очков навыков.

Во второй строке содержится $N - 1$ число p_2, p_3, \dots, p_N ($1 \leq p_i < i$) — требования для изучения навыков.

Формат выходных данных

Если выбрать два подходящих навыка невозможно, выведите одно слово — «NO» (без кавычек).

В противном случае на первой строке выведите «YES» (без кавычек). На второй строке выведите уровень выбранных навыков, а на третьей — номера выбранных навыков через пробел. Если существует несколько вариантов выбора, выведите любой из них.

Примеры

стандартный ввод	стандартный вывод
4 3 1 2 2	YES 2 3 4
4 2 1 2 2	NO
8 3 1 1 1 3 4 3 7	YES 2 5 7

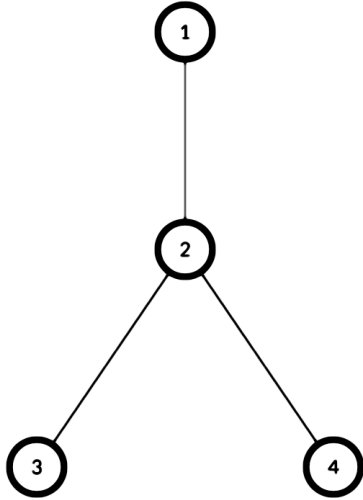
Замечание

Первый и второй примеры отличаются лишь количеством доступных очков навыков. В первом примере это количество равно 3, чего достаточно, чтобы выучить навыки 2, 3 и 4, выбрав после

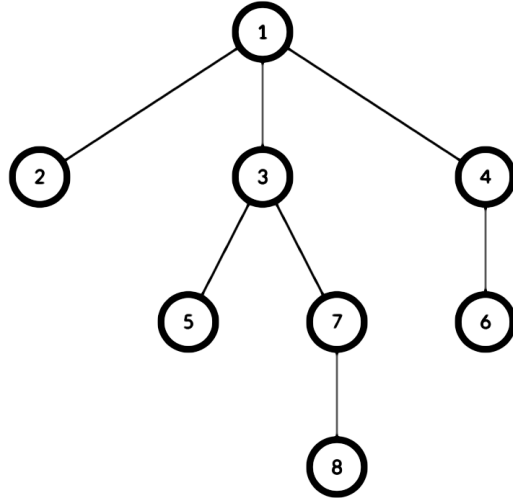
этого навыки с номерами 3 и 4. Уровень этих навыков равен 2. Во втором примере очков навыков недостаточно, чтобы изучить два навыка одного уровня.

В третьем примере можно выучить навыки 3 и 4 и тем самым получить два навыка первого уровня. Однако, данное решение не является оптимальным. Можно потратить больше очков, изучив навыки 3, 5 и 7, получив два навыка второго уровня.

Ниже приведены рисунки, описывающие примеры.



Примеры 1 и 2



Пример 3

Задача D. Юные следопыты

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Отряд юных следопытов отправился в учебную экспедицию навстречу своим первым приключениям. И возглавляет их старший следопыт Рассел. Вот герои зашли в лес, разбили лагерь и дальше решили разделить на группы, чтобы исследовать как можно больше интересных мест. Рассел должен был выбрать состав групп, но столкнулся с одной проблемой. . .

Многие юные следопыты неопытны, и отправлять их маленькими группами — не всегда хорошая идея. Даже сам Рассел недавно стал старшим следопытом и нечасто бывал в экспедициях. Каждый следопыт характеризуется своей неопытностью — целым положительным числом e_i . Рассел решил, что юный следопыт с неопытностью e может идти лишь в группе, количество следопытов в которой не меньше e .

Теперь задача Рассела — определить, какое наибольшее число групп следопытов он сможет организовать. При этом может получиться, что некоторые следопыты не войдут в состав ни одной группы, это не страшно, ведь и в лагере для них найдется работа. Рассел очень переживает за успех экспедиции, и потому попросил вас помочь ему.

Формат входных данных

В первой строке записано число T ($1 \leq T \leq 2 \cdot 10^5$) — количество независимых тестовых случаев. В следующих $2T$ строках следует описание тестовых случаев.

В первой строке описания каждого теста задано целое число юных следопытов N ($1 \leq N \leq 2 \cdot 10^5$).

В следующей строке записаны N целых чисел e_1, e_2, \dots, e_N ($1 \leq e_i \leq N$), где e_i — неопытность i -го следопыта.

Гарантируется, что сумма N по всем тестовым случаям не превосходит $3 \cdot 10^5$.

Формат выходных данных

Выведите T чисел, каждое на отдельной строке.

В i -й строке выведите наибольшее число групп, которое можно организовать в i -м тестовом случае.

Пример

стандартный ввод	стандартный вывод
2	3
3	2
1 1 1	
5	
2 3 1 2 2	

Замечание

В первом примере можно сформировать три группы, в каждой из которых будет один следопыт. Это возможно, так как неопытность всех трех следопытов равна 1, что не меньше, чем размер их групп.

Во втором примере можно сформировать две группы. В первой группе окажутся следопыты с неопытностью 1, 2 и 3, а во второй группе — два следопыта с неопытностью 2.

Этот способ — не единственный возможный. Можно, например, сформировать одну группу из трех следопытов с неопытностью 2, а также еще одну группу, в которой будет всего один следопыт с неопытностью 1. При таком разбиении на группы следопыт с неопытностью 3 не войдет в состав ни одной группы.

Задача Е. М — многомерность

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

Многие в детстве играют с кубиками, затем все в школе изучают геометрию и встречаются с такими простыми объектами, как параллелепипеды. Но ведь изучать геометрию в трехмерном пространстве — это так скучно! Даже четырехмерным пространством уже никого не удивишь! Поэтому в этой задаче мы предлагаем вам изучить параллелепипеды в M -мерном пространстве. Чувствуете, как интересно?

Определим M -мерный параллелепипед как набор отрезков $[a_1, b_1], [a_2, b_2], \dots, [a_M, b_M]$, где $a_i < b_i$ для всех $i = 1 \dots M$. Для простоты будем считать, что все a_i и b_i являются целыми.

Скажем, что точка с координатами (x_1, x_2, \dots, x_M) лежит внутри параллелепипеда, если выполнены неравенства:

$$\begin{aligned} a_1 &\leq x_1 \leq b_1, \\ a_2 &\leq x_2 \leq b_2, \\ &\dots \\ a_M &\leq x_M \leq b_M. \end{aligned}$$

Вам даны N M -мерных параллелепипедов. Требуется посчитать, сколько точек с целочисленными координатами лежат внутри ровно $N - 1$ параллелепипеда. Так как ответ может быть большим, выведите остаток от деления количества точек на число 998 244 353.

Формат входных данных

В первой строке записаны два числа N и M ($2 \leq N \leq 2 \cdot 10^5, 1 \leq M \leq 2 \cdot 10^5, 2 \leq N \cdot M \leq 2 \cdot 10^5$) — количество параллелепипедов и размерность пространства соответственно.

Каждая из следующих N строк задает параллелепипед. В каждой строке через пробел записаны $2 \cdot M$ чисел в следующем порядке: $a_1, b_1, a_2, b_2, \dots, a_M, b_M$ ($-10^6 \leq a_i < b_i \leq 10^6$ для всех i).

Формат выходных данных

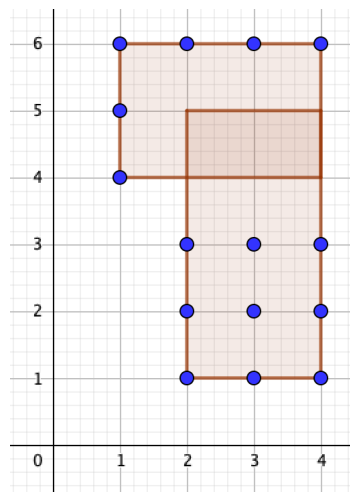
Выведите одно число — остаток от деления количества точек, лежащих внутри ровно $N - 1$ параллелепипеда, на число 998 244 353.

Примеры

стандартный ввод	стандартный вывод
2 2 2 4 1 5 1 4 4 6	15
4 1 1 6 2 4 6 7 2 9	4

Замечание

Рисунок к первому примеру, в котором даны два прямоугольника. Необходимо посчитать все целочисленные точки, которые лежат внутри (или на границе) ровно одного прямоугольника. В данном примере таких точек 15. Все эти точки отмечены на рисунке.



Во втором примере $M = 1$, значит параллелепипеды являются обыкновенными отрезками на прямой.

Задача F. Игра с массивом

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

Петя и Вася, как всегда, заняты тем, что соревнуются между собой в новой интересной игре.

Сначала Петя должен придумать массив длины N , состоящий из целых положительных чисел, сумма которых равна S , а также целое число K такое, что $0 \leq K \leq S$. Чтобы выиграть, Вася должен найти такой непустой подотрезок в массиве Пети, что сумма его элементов равна K или $S - K$. В противном случае Васе придётся признать своё поражение.

Зная N и S , определите, может ли Петя одержать победу при условии, что Вася будет играть наилучшим для себя способом. Если Петя может выиграть, подскажите, как ему следует сыграть.

Формат входных данных

В единственной строке содержатся два целых числа N и S ($1 \leq N \leq S \leq 10^6$) — необходимая длина массива и сумма его элементов.

Формат выходных данных

Если Петя может победить, то в первой строке следует вывести «YES» (без кавычек), во второй строке нужно вывести необходимый массив, состоящий из N натуральных чисел, сумма которых равна S , а в третьей строке вывести одно целое число K . Если подходящих ответов несколько, то можно вывести любой из них.

В противном случае в единственной строке выведите «NO» (без кавычек).

Вы можете выводить каждую букву в любом регистре (строчную или заглавную).

Примеры

стандартный ввод	стандартный вывод
1 4	YES 4 2
3 4	NO
3 8	YES 2 1 5 4

Задача G. Последовательность с цифрами

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

Определим рекуррентную последовательность следующим образом:

$$a_{n+1} = a_n + \minDigit(a_n) \cdot \maxDigit(a_n).$$

Здесь $\minDigit(x)$ и $\maxDigit(x)$ — минимальная и максимальная цифры в десятичной записи числа x без ведущих нулей соответственно. Для примеров обратитесь к примечаниям.

Ваша задача — по заданным a_1 и K вычислить a_K .

Формат входных данных

В первой строке записано единственное число t ($1 \leq t \leq 1000$) — количество независимых наборов входных данных.

Каждый набор входных данных состоит из двух целых чисел a_1 и K ($1 \leq a_1 \leq 10^{18}$, $1 \leq K \leq 10^{16}$), записанных через пробел на отдельной строке.

Формат выходных данных

Для каждого набора входных данных выведите одно число a_K на отдельной строке.

Пример

стандартный ввод	стандартный вывод
8	42
1 4	487
487 1	519
487 2	528
487 3	544
487 4	564
487 5	588
487 6	628
487 7	

Замечание

$$a_1 = 487$$

$$a_2 = a_1 + \minDigit(a_1) \cdot \maxDigit(a_1) = 487 + \min(4, 8, 7) \cdot \max(4, 8, 7) = 487 + 4 \cdot 8 = 519$$

$$a_3 = a_2 + \minDigit(a_2) \cdot \maxDigit(a_2) = 519 + \min(5, 1, 9) \cdot \max(5, 1, 9) = 519 + 1 \cdot 9 = 528$$

$$a_4 = a_3 + \minDigit(a_3) \cdot \maxDigit(a_3) = 528 + \min(5, 2, 8) \cdot \max(5, 2, 8) = 528 + 2 \cdot 8 = 544$$

$$a_5 = a_4 + \minDigit(a_4) \cdot \maxDigit(a_4) = 544 + \min(5, 4, 4) \cdot \max(5, 4, 4) = 544 + 4 \cdot 5 = 564$$

$$a_6 = a_5 + \minDigit(a_5) \cdot \maxDigit(a_5) = 564 + \min(5, 6, 4) \cdot \max(5, 6, 4) = 564 + 4 \cdot 6 = 588$$

$$a_7 = a_6 + \minDigit(a_6) \cdot \maxDigit(a_6) = 588 + \min(5, 8, 8) \cdot \max(5, 8, 8) = 588 + 5 \cdot 8 = 628$$

Задача Н. Карантин

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

Вокруг живописного круглого озера расположено N домов на одинаковом расстоянии D друг от друга. Михаил живет в одном из этих домов, во всех остальных домах живут его друзья. Все нравится Михаилу: и прекрасная природа, и добрые соседи, и мягкий климат, но и в эти места добрался коронавирус... И вот теперь жители самоизолировались, и могут выходить на прогулку только по одному человеку, при этом все остальные находятся в своих домах.

Михаил хочет во время своей прогулки пройти мимо каждого дома, чтобы оставить в почтовом ящике каждого из домов письмо-приветствие своим друзьям. При этом перемещаться между домами он будет по кратчайшему пути. Изначально Михаил выбирает некоторый порядок, в котором он будет посещать дома друзей, после чего начинает свою прогулку. Оставив письмо в почтовом ящике одного дома, он перемещается к другому дому, и так до тех пор, пока не пройдет все дома, после чего зайдет в гости к другу, который живет в последнем посещенном доме.

Михаил, пользуясь возможностью прогуляться, хочет пройти во время своей прогулки как можно большее расстояние. Это расстояние ему нужно сообщить для получения пропуска на выход из дома. В таком важном деле ошибаться Михаилу нельзя. Помогите Михаилу верно вычислить максимальное расстояние, которое он может пройти во время прогулки.

Формат входных данных

В единственной строке через пробел записаны два числа: N и D ($3 \leq N \leq 10^6, 1 \leq D \leq 10^6$) — количество домов и расстояние между соседними домами соответственно.

Формат выходных данных

Выведите одно число — максимальное расстояние, которое сможет пройти Михаил во время прогулки.

Примеры

стандартный ввод	стандартный вывод
3 5	10
6 10	130

Задача I. Сосчитайте треугольники

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

Как и у любого неизвестного математика, у Юрия есть любимые числа: A , B , C и D , причем $A \leq B \leq C \leq D$. Также Юрий очень любит треугольники, поэтому в один день он задумался: сколько существует невырожденных треугольников с целочисленными длинами сторон x , y и z таких, что $A \leq x \leq B \leq y \leq C \leq z \leq D$?

Сейчас Юрий очень занят подготовкой задач для очередной олимпиады, поэтому он просит вас помочь посчитать количество интересующих его треугольников.

Треугольник называется невырожденным тогда и только тогда, когда его вершины не лежат на одной прямой.

Формат входных данных

В единственной строке через пробел записаны четыре целых числа A , B , C и D ($1 \leq A \leq B \leq C \leq D \leq 5 \cdot 10^5$) — любимые числа Юрия.

Формат выходных данных

Выведите одно число — количество невырожденных треугольников с целочисленными длинами сторон x , y и z , для которых выполнено неравенство: $A \leq x \leq B \leq y \leq C \leq z \leq D$.

Примеры

стандартный ввод	стандартный вывод
1 2 3 4	4
1 2 2 5	3
500000 500000 500000 500000	1

Замечание

В первом примере можно составить треугольники со следующими длинами сторон: $(1, 3, 3)$, $(2, 2, 3)$, $(2, 3, 3)$ и $(2, 3, 4)$.

Во втором примере можно составить треугольники: $(1, 2, 2)$, $(2, 2, 2)$ и $(2, 2, 3)$.

В третьем примере можно составить лишь один равносторонний треугольник. Длины всех сторон будут равны $5 \cdot 10^5$.

Задача J. Реставрационное расстояние

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

Вам нужно отреставрировать стену. Стена состоит из N столбиков кирпичей, высота i -го столбика изначально равна h_i , высота измеряется количеством кирпичей. После реставрации все N столбиков должны иметь одинаковую высоту.

Вам доступны следующие операции:

- положить кирпич на верх одного из столбиков, стоимость этой операции равна A ;
- убрать кирпич с верха одного из непустых столбиков, стоимость этой операции равна R ;
- переложить один кирпич с верха одного из непустых столбиков на верх другого столбика, стоимость этой операции равна M .

Вы не можете создавать дополнительные столбики или игнорировать какие-то из существующих столбиков, даже если их высота стала равна 0.

За какую минимальную суммарную стоимость возможно провести реставрацию, то есть сделать высоты всех столбиков равными?

Формат входных данных

В первой строке записаны четыре целых числа N, A, R, M ($1 \leq N \leq 10^5, 0 \leq A, R, M \leq 10^4$) — количество столбиков кирпичей в стене, а также стоимости доступных операций.

Во второй строке записаны N целых чисел h_i ($0 \leq h_i \leq 10^9$) — начальные высоты столбиков.

Формат выходных данных

Выведите одно целое число — минимальную суммарную стоимость реставрации стены.

Примеры

стандартный ввод	стандартный вывод
3 1 100 100 1 3 8	12
3 100 1 100 1 3 8	9
3 100 100 1 1 3 8	4
5 1 2 4 5 5 3 6 5	4
5 1 2 2 5 5 3 6 5	3

Задача К. Угадать количество делителей

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Это интерактивная задача.

Загадано число $1 \leq X \leq 10^9$. Вам **не нужно** угадывать это число. Вам нужно **определить количество делителей** этого числа, и даже это вам **не нужно делать точно**: ваш ответ будет считаться верным, если его абсолютная погрешность не превышает 7 **или** его относительная погрешность не превышает 0.5. Формально, пусть ваш ответ равен ans , а количество делителей X равно d , тогда ваш ответ будет считаться правильным, если выполнено **хотя бы одно** из следующих двух условий:

- $|ans - d| \leq 7$
- $\frac{1}{2} \leq \frac{ans}{d} \leq 2$

Вы можете не более 22 раз сделать запрос. Запрос состоит из одного числа $1 \leq Q \leq 10^{18}$. В ответ на запрос вы получите $gcd(X, Q)$ — наибольший общий делитель X и Q .

Число X зафиксировано до всех запросов. Иными словами, **интерактор не является адаптивным**.

Назовём процесс отгадывания количества делителей числа X *игрой*. В рамках одного теста вам нужно будет сыграть T независимых игр, то есть отгадать количество делителей T раз для T независимых чисел X .

Формат входных данных

На первой строке записано одно целое число T ($1 \leq T \leq 100$) — количество игр.

Протокол взаимодействия

Чтобы сделать запрос, выведите строку вида «? Q» ($1 \leq Q \leq 10^{18}$). После запроса считайте одно число — $gcd(X, Q)$. Вы можете сделать не более 22 таких запросов в рамках одной игры.

Если вы считаете, что знаете количество делителей X с достаточной точностью, выведите ваш ответ в формате «! ans». ans должно быть целым числом. Если это последняя игра, то вы должны завершить выполнение программы, иначе вы должны начать следующую игру. Обратите внимание, что интерактор не выводит ничего в ответ на вывод ответа.

После вывода запроса или ответа не забудьте вывести перевод строки и сбросить буфер вывода. Для сброса буфера вывода используйте:

- `fflush(stdout)` или `cout.flush()` в C++;
- `System.out.flush()` в Java;
- `flush(output)` в Pascal;
- `stdout.flush()` в Python;
- смотрите документацию для других языков.

Пример

стандартный ввод	стандартный вывод
2	? 982306799268821872
1	? 230856864650023977
1	? 134690134760714371
1	! 5 ? 1024
1024	? 1048576
1048576	? 1073741824
4194304	! 42

Замечание

Почему ограничение на запросы именно 22? Возможно, автор задачи — фанат Тейлор Свифт. Рассмотрим пример из условия.

В первой игре загадано число $X = 998\,244\,353$. Было бы сложно это угадать, правда? Это число является простым, то есть количество его делителей равно 2. Решение сделало запросы с несколькими случайными числами, и ответы на все запросы оказались равны 1 (удивительно, что ни один из трёх запросов не оказался кратным $998\,244\,353$). Логично предположить, что у загаданного числа не очень много делителей, поэтому решение ответило 5. Почему бы и не 5. Этот ответ будет засчитан, так как $|5 - 2| = 3 \leq 7$.

Во второй игре загадано число $X = 4\,194\,304 = 2^{22}$, количество его делителей равно 23. Решение сделало запросы $1024 = 2^{10}$, $1\,048\,576 = 2^{20}$, $1\,073\,741\,824 = 2^{30}$ и получило ответы $1024 = 2^{10}$, $1\,048\,576 = 2^{20}$, $4\,194\,304 = 2^{22}$, соответственно. Затем решение окончательно запуталось и выдало ответ на Главный вопрос жизни, Вселенной и всего такого. Этот ответ будет засчитан, так как $\frac{1}{2} \leq \frac{42}{23} \leq 2$.

Задача L. Стековая машина

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

Дано $0 \leq N \leq 100$ целых неотрицательных чисел, выведите их сумму.

Эх, как было бы хорошо, если бы вам нужно было написать программу для решения такой задачи на C++ или Python...

У вас есть машина, память которой — это стек, элементы которого являются целыми числами. Количество элементов стека в данный момент мы будем обозначать как sz .

Напомним, что стек — это абстрактная структура данных, хранящая коллекцию элементов, и работающая по принципу LIFO (последним пришёл — первым вышел). Стек поддерживает две базовые операции: положить элемент на вершину стека, взять элемент с вершины стека. В качестве аналогии можно представить стопку тарелок: вы не можете оперировать с тарелками не на вершине стопки.

Список доступных инструкций приведен ниже. RTE (Run-Time Error) обозначает, что выполнение программы завершилось с ошибкой, вам следует избегать таких ситуаций.

- PUSHZ — положить на стек 0;
- POP — взять со стека x (если $sz < 1$, то RTE);
- SWAP2 — взять со стека x , взять со стека y , положить на стек x , положить на стек y (если $sz < 2$, то RTE);
- SWAP3 — взять со стека x , взять со стека y , взять со стека z , положить на стек y , положить на стек x , положить на стек z (если $sz < 3$, то RTE);
- COPY — взять со стека x , положить на стек x , положить на стек x (если $sz < 1$, то RTE);
- INC — взять со стека x , положить на стек $x + 1$ (если $sz < 1$, то RTE);
- DEC — взять со стека x , положить на стек $x - 1$ (если $sz < 1$, то RTE);
- ADD — взять со стека x , взять со стека y , положить на стек $x + y$ (если $sz < 2$, то RTE);
- SUB — взять со стека x , взять со стека y , положить на стек $x - y$ (если $sz < 2$, то RTE);

Также доступны условный оператор и цикл while. Чтобы их использовать, сначала нужно научиться задавать какие-то условия:

- EZ — TRUE если на вершине стека 0 (если $sz < 1$, то RTE);
- GZ — TRUE если на вершине стека положительное число (если $sz < 1$, то RTE);
- HAVE1, HAVE2 — TRUE если $sz \geq k$ (обратите внимание, что $k \in \{1, 2\}$, HAVE3 не является условием)

К любому условию можно добавить NOT в начале, например NOT GZ (обратите внимание на пробел) TRUE если на вершине стека 0 или отрицательное число.

Условный оператор:

```
IF cond THEN
```

```
BEGIN
```

```
    body
```

```
END
```

Цикл while:


```
WHILE cond DO
BEGIN
    body
END
```

В обоих случаях `cond` — одно из условий, описанных выше, `body` — тело условного оператора или цикла — последовательность команд, которые будут выполнены, если `cond=TRUE` (один раз в случае условного оператора, или много раз в случае цикла). Циклы и условные операторы могут быть вложенными.

Вы должны написать программу для этой машины, которая решает следующую задачу:

Изначально стек был пуст. Числа N, a_1, a_2, \dots, a_N положили на стек в таком порядке (наверху стека лежит a_N , N лежит в самом низу). После выполнения программы на вершине стека должно лежать число $a_1 + a_2 + \dots + a_N$ (также в стеке могут лежать другие числа).

Гарантируется, что $0 \leq N \leq 100$, $0 \leq a_i$. Каждая инструкция (а также проверки условия для условного оператора и цикла) выполняется 1 такт. Выполнение программы должно занимать не более $(N + 22)^2$ тактов. В процессе выполнения программы не должно происходить RTE.

Формат входных данных

В этой задаче нет входных данных.

Формат выходных данных

Выведите программу для стековой машины, решающую описанную задачу.

Вы не обязаны использовать отступы. Любой пробельный символ может быть заменён на любое ненулевое количество пробельных символов, вставлять пробельные символы внутрь инструкций нельзя.

Все инструкции должны быть записаны именно так, как они записаны в тексте условия.

Пример

стандартный ввод	стандартный вывод
	<pre>PUSHZ SWAP2 WHILE NOT EZ DO BEGIN COPY SWAP3 ADD SWAP2 DEC END POP WHILE NOT EZ DO BEGIN COPY DEC END WHILE HAVE2 DO BEGIN ADD END</pre>

Замечание

Ответ на пример из условия содержит две программы, которые вычисляют сумму чисел от 1 до N для $N \geq 0$.

В стеке могут храниться числа произвольного размера. И ваша программа должна работать для a_i произвольного размера.

Намеренные попытки дестабилизировать работу проверяющей системы могут повлечь дисквалификацию команды. Жюри оставляет за собой право решать, что является невинной ошибкой, а что — намеренной попыткой дестабилизировать работу проверяющей системы.