

Задача А. Подстава от Кевина

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

Кевин остался один дома. Он понимает, что этот год не будет особенным, и жулики обязательно придут. Так что Кевин решил подготовить для них ловушки заранее.

Любой путь в дом лежит через ограду, поэтому Кевин решил изготовить первую ловушку именно из нее. Ограда состоит из n досок. Про каждую доску Кевин знает сколько она весит. Он считает, что ограда станет ловушкой, если переставить в ней две доски. При этом, если суммарный вес этих досок будет больше чем x , то у Кевина не получится их переставить, а если разность весов переставляемых досок будет меньше чем y , то ловушка не произведёт должного эффекта. Так например, при $x = 5$ и $y = 2$, перестановка досок с весами 1 и 4 устроит Кевина, а 2 и 3 нет.

Кевину нужно разработать планы остальных ловушек, поэтому он просит вас помочь ему узнать может ли он из ограды сделать ловушку.

Формат входного файла

В первой строке входного файла содержится три целых числа n, x, y ($1 \leq n \leq 100\,000; 1 \leq y \leq x \leq 100\,000$) — количество досок в ограде, а также числа, которые задают ловушку, соответственно.

Во второй строке содержится n целых чисел a_i ($1 \leq a_i \leq 100\,000$), характеризующих вес досок в ограде.

Формат выходного файла

В единственной строке выходного файла выведите два числа — номера досок, которые Кевин может переставить, чтобы получить ловушку с должным эффектом. В случае если ответов несколько, выведите любой. Если ответа не существует, выведите 0.

Пример

<code>trap.in</code>	<code>trap.out</code>
5 5 2 1 2 3 4 5	1 4
4 5 2 2 3 4 5	0
3 20 1 1 10 5	3 2

Система оценивания

Первая группа тестов состоит из тестов, для которых выполняется ограничение $1 \leq n \leq 1000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 50 баллов.

Вторая группа тестов состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 50 баллов.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».

Задача В. Кевин и машинка

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

За помощь в уборке снега миссис Хесс подарила Кевину радиоуправляемую машинку, которую случайно забрала из аэропорта.

Кевин, вдоволь наигравшись, решил посмотреть, что же находится внутри машинки. Но разобрать ее оказалось не так просто, как он думал. Внутри машинки оказался хитрый электронный замок и две строки из цифр. Прочитав указание, Кевин понял, что для того, чтобы разобрать машинку, нужно решить непростую головоломку. Замок позволяет взять любые две соседние цифры в первой строке и заменить их на одну цифру, равную остатку от деления их суммы на десять. Если первая строка в точности совпадет со второй, замок откроется. Поскольку после применения любой операции длина первого числа уменьшается, и нет возможности отменить свои действия, у Кевина есть всего одна попытка.

Помогите Кевину понять, можно ли хотя бы теоретически открыть замок.

Формат входного файла

В первой строке дана одна строка a ($1 \leq |a| \leq 200\,000$) — первое число.

В второй строке дана одна строка b ($1 \leq |b| \leq 200\,000$) — второе число.

Обратите внимание, что строки могут содержать ведущие нули.

Формат выходного файла

Выведите «YES» если можно получить из первого числа второе, и «NO» иначе.

Пример

<code>numbers.in</code>	<code>numbers.out</code>
123456 326	YES
<code>numbers.in</code>	<code>numbers.out</code>
123456 12345	NO

Система оценивания

Первая группа тестов состоит из тестов, для которых выполняется ограничение $1 \leq |a|, |b| \leq 10$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 20 баллов.

Вторая группа тестов состоит из тестов, для которых выполняется ограничение $1 \leq |a|, |b| \leq 500$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 30 баллов.

Третья группа тестов состоит из тестов, для которых выполняется ограничение $1 \leq |a|, |b| \leq 200\,000$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 50 баллов.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».

Задача С. Регистры для Кевина

Имя входного файла:	<code>registers.in</code>
Имя выходного файла:	<code>registers.out</code>
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Кевин снова хочет проучить грабителей. Для этого он решил собрать электронную ловушку. Одна из ее частей — это арифметическое логическое устройство (АЛУ). АЛУ производит все арифметические операции, которыми мы привыкли пользоваться в наших любимых языках программирования.

В этой задаче с помощью АЛУ потребуется складывать два n -битных числа. АЛУ состоит из регистров. Регистр — это устройство, которое хранит в себе число.

Есть несколько типов регистров. Регистр типа x может содержать в себе числа от 0 до $2^x - 1$. АЛУ должно состоять из регистров одного типа t , причем t не задано и выбирается Кевином. Кевин может купить регистры любого типа в любом количестве.

Сложение происходит следующим образом: n бит в порядке от младших к старшим обоих чисел разбиваются на отрезки длины 2^t , кроме, возможно, последнего. Всего получается $\lceil \frac{n}{2^t} \rceil$ отрезков. Биты одного отрезка обоих чисел образуют два новых числа. Сложение этих чисел происходит в одном регистре. Регистры нумеруются от младших к старшим. Сначала происходит сложение в первом регистре, затем во втором и так далее. Если произошло переполнение некоторого регистра, то есть результат превышает $2^t - 1$, то происходит перенос в следующий регистр. Если с учетом этого переноса в следующем регистре произошло переполнение, то происходит перенос в следующий регистр, и так далее.

Перенос — это самая затратная по времени операция. Для каждого типа регистров известно время, которое он тратит на перенос бита. Можно считать, что сложение в одном регистре происходит мгновенно.

В продаже имеются регистры k типов, неограниченное количество каждого типа. От вас требуется определить, регистры какого типа следует установить на АЛУ, чтобы достичь минимального времени исполнения.

Задача также осложняется тем, что эти числа могут меняться в зависимости от конструкции устройства.

В результате сборки устройства происходит m изменений: некоторый бит некоторого числа меняется на противоположный.

После каждого изменения бита, от вас требуется посчитать минимальное время, которое потребуется на сложение двух новых чисел при оптимальном выборе типа регистра.

Обратите внимание, что после ответа на запрос измененный бит не меняет свое значение обратно.

Формат входного файла

В первой строке находятся два натуральных числа n, k ($1 \leq n \leq 10^5, 1 \leq k \leq 17$) — количество бит в обоих числах и количество типов регистров, имеющихся в продаже.

В следующих двух строках находятся два числа в двоичной системе счисления, возможно с ведущими нулями. Каждое состоит ровно из n бит. Каждое число дано от старших битов к младшим.

В следующих k строках находится описание типов регистров, имеющихся в продаже — два целых числа h_i, p_i ($1 \leq 2^{h_i} \leq n, 1 \leq p_i \leq 10\,000$) — тип регистра и время, которое занимает перенос при использовании этого типа регистров. Все h_i различны.

В следующей строке находится m ($1 \leq m \leq 10^5$) — количество запросов на изменение.

В следующих m строках находятся запросы на изменение бит чисел — два целых числа a_j, b_j ($1 \leq a_j \leq 2, 0 \leq b_j < n$) — номер числа, и номер разряда числа. Самый младший разряд имеет номер 0.

Формат выходного файла

На каждый запрос изменения бита выведите минимальное время, которое потребуется для сложения двух чисел, при оптимальном выборе типа регистра.

Пример

registers.in	registers.out
10 3	5
1011110010	5
0010010010	5
0 1	6
1 3	7
3 15	
5	
1 9	
2 9	
1 0	
2 2	
2 3	
12 3	0
101111111111	0
010000000000	0
0 1	3
1 1	1
3 7	2
6	
1 10	
2 10	
1 0	
2 7	
1 9	
2 10	

Система оценивания

Первая группа тестов 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 40 баллов.

Вторая группа тестов состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 60 баллов.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».

Задача D. Плеер Кевина

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

Кевин решил как следует отдохнуть. Для начала он хочет послушать музыку на своем плеере. На его плеере сохранено n песен. Известен порядок, в котором будут воспроизводиться композиции.

В некоторых песнях есть особенно классные отрезки, которые нравятся Кевину. Для каждого из них известно, сколько радости приносит прослушивание одной секунды этого отрезка. Оставшиеся моменты песни, которые не вошли ни в один классный отрезок, не приносят радости.

Радость Кевина выражается целым неотрицательным числом. Перед началом прослушивания она равна 0. Кевин хочет, чтобы его радость достигла F .

Также в плеере доступна возможность ускорить воспроизведение. При ускорении за одну секунду реального времени проходит v секунд песни. Если во время ускорения песня заканчивается, то ускорение продолжается с начала следующей. Если во время ускорения встречается некоторая часть классного отрезка, то радости она не приносит.

Если количество радости, доставляемое от прослушивания некоторого отрезка равно f , и Кевин прослушивал его в течении t секунд ($t \geq 0$, t вещественно), то Кевин получит $f \cdot t$ радости.

Ускорять воспроизведение также можно в течении любого вещественного количества секунд.

Ускорение можно начать и закончить в любой момент времени. Включение и выключение ускорения происходят мгновенно.

Плейлист не зациклен, то есть после того как все песни из плейлиста воспроизведены, плеер заканчивает свою работу.

Как только радость Кевина достигает F , он сразу же прекращает слушать музыку. Помогите ему определить, какое наименьшее время придется провести с плеером, чтобы достичь радости F .

Формат входного файла

В первой строке находятся три натуральных числа n, v, F ($1 \leq n \leq 10^5, 1 \leq v, F \leq 10^9$) — количество песен, коэффициент ускорения и радость, которой хочет достигнуть Кевин.

В следующих n строках идет описание классных отрезков в песне: в i -й из них содержатся два целых числа t_i, k_i ($1 \leq t_i \leq 10^9, 0 \leq k_i$) — длина i -й песни, количество классных отрезков в ней, а затем k_i троек чисел $l_{i,j}, r_{i,j}, f_{i,j}$ ($0 \leq l_{i,j} \leq r_{i,j} \leq t_i, r_{i,j} \leq l_{i,j+1}, 1 \leq f_{i,j} \leq 10^9$) — с какой секунды по какую находится классный отрезок и количество радости, которое доставляет прослушивание одной секунды этого отрезка. Все l_{ij}, r_{ij}, f_{ij} целые.

Песни заданы в порядке воспроизведения.

Сумма всех k_i не превосходит 10^5 .

Формат выходного файла

Выведите единственное вещественное число — количество секунд, которое нужно провести с плеером, чтобы достичь радости F . Если сделать этого невозможно, выведите -1 .

Ответ будет считаться правильным, если относительная или абсолютная погрешность не будет превосходить 10^{-8} .

Пример

player.in	player.out
3 2 5 4 2 0 1 1 2 4 1 6 2 0 1 1 1 5 4 3 1 1 3 2	3.7500000000
2 2 10 3 2 0 1 1 1 3 1 2 1 0 2 3	-1
4 1 8 5 1 2 4 2 4 1 1 3 1 3 1 0 1 3 6 2 0 2 10 3 5 9	9.6666666667

Система оценивания

Первая группа тестов состоит из тестов, для которых выполняется $v = 1$. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 20 баллов.

Вторая группа тестов состоит из тестов, для которых выполняется $n \leq 1000$. Сумма всех k_i не превосходит 1000. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 30 баллов.

Третья группа тестов состоит из тестов, для которых выполняются полные ограничения. Баллы за эту группу начисляются только при прохождении всех тестов группы. Стоимость группы составляет 50 баллов.

Обратите внимание на возможность узнать результат проверки вашего решения на всех тестах, нажав на ссылку «Request feedback» на вкладке «Runs».