

Задача А. Простое произведение

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

Для контрольных работы по квазиматике Иван Васильевич каждый раз готовит n интереснейших задач. Все задачи пронумерованы натуральными числами от 1 до n . Его ученики, Витя и Сережа, давно заметили, что для получения зачета нужно решить всего две из них. Также жизненный опыт мальчиков подсказывает им, что две самые простые задачи обладают следующим свойством: произведение их номеров должно быть простым числом.

Узнать число n перед контрольной не составляет труда для наших героев. Однако, Витя и Сережа не хотят тратить много времени на выбор номеров самых простых задач. Поэтому они решили написать программу для решения данной непростой задачи. Но, как это часто бывает, что-то пошло не так.

Помогите мальчикам по известному числу n быстро научиться находить две задачи, произведение номеров которых является простым числом.

Формат входных данных

Вводится одно натуральное число n ($2 \leq n \leq 2147483647$) — количество задач.

Формат выходных данных

Выведите два натуральных числа в одной строке — искомые номера задач. Числа можно выводить в любом порядке. Если пар простых задач несколько, выведите любую. Если найти пару простых задач не представляется возможным, ваша программа должна вывести -1.

Пример

стандартный ввод	стандартный вывод
6	5 1

Замечание

Простое число — натуральное (целое положительное) число, имеющее ровно два различных натуральных делителя — единицу и самого себя. Другими словами, число x является простым, если оно больше 1 и при этом делится без остатка только на 1 и на x . К примеру, 5 — простое число, а 6 не является простым числом, так как, помимо 1 и 6, также делится на 2 и на 3.

Задача В. Карточки с цифрами

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

У Миши в руках оказались две карточки, на каждой из которых нарисована цифра. Эти карточки обладают следующим свойством: если её перевернуть, то все равно будет образована цифра (возможно другая). Так, если перевернуть карточку с единицей или восьмеркой, то она останется единицей или восьмеркой соответственно. А если перевернуть шестерку или девятку, то они превратятся в девятку или шестерку соответственно. Других цифр на карточках нарисовано быть не может.

Теперь Миша хочет из двух этих карточек составить наибольшее возможное число, возможно перевернув их и/или поменяв местами. Помогите Мише определить, какое наибольшее двузначное число может быть составлено таким образом.

Формат входных данных

Даны две цифры через пробел. Каждая из цифр может быть единицей, шестеркой, восьмеркой или девяткой.

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
1 6	91

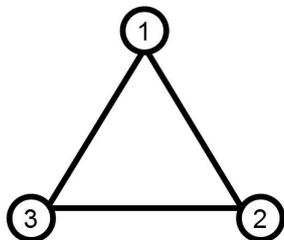
Замечание

В примере шестерку поставили первой и перевернули, получилась девятка. Единицу поставили второй, поэтому получилось число 91.

Задача С. Самое большое число

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

В каждой вершине равностороннего треугольника записано по одной цифре. Из этих цифр мы можем составить трехзначное число, начиная с любой, выбранной нами, вершины треугольника и двигаясь в одном из двух возможных направлений – по часовой стрелке или против часовой стрелки.



Напишите программу, которая из цифр, записанных в вершинах треугольника, составляет наибольшее трехзначное число, которое мы можем сформировать по описанным выше правилам.

Формат входных данных

В единственной строке стандартного ввода содержатся три целых числа a, b, c ($0 \leq a, b, c \leq 9$), разделенные пробелами. По крайней мере, одна из введенных цифр не равна нулю.

Формат выходных данных

Выведите одно целое трехзначное число.

Примеры

стандартный ввод	стандартный вывод
1 2 3	321
4 2 3	432
1 3 3	331

Задача D. Бочки

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

На складе хранятся бочки двух типов: с бензином и с керосином. Все бочки выставлены в один ряд, при этом сначала идут бочки одного типа, затем бочки другого типа. Скоро подъедет машина, в которую нужно будет загрузить k бочек.

Петя первый день работает грузчиком на этом складе, поэтому он просто возьмет, и загрузит самые левые k бочек со склада в машину. Требуется определить, сколько бочек с бензином при этом будет загружено в машину.

Формат входных данных

В первой строке дано целое число n_1 и символ c_1 ($1 \leq n_1 \leq 100$), которые описывают бочки, которые стоят слева. Символ A обозначает бочки с бензином, символ B бочки с керосином. Число n_1 задаёт сколько бочек этого типа стоят слева в ряду бочек.

Во второй строке даны целое число n_2 и символ c_2 ($1 \leq n_2 \leq 100$), описывающие бочки справа. Символ A обозначает бочки с бензином, символ B бочки с керосином. Число n_2 задаёт сколько бочек этого типа стоят справа в ряду бочек.

В третьей строке дано целое число k ($1 \leq k \leq n_1 + n_2$). Гарантируется, что есть бочки обоих типов, то есть $c_1 \neq c_2$.

Формат выходных данных

Выведите одно число — сколько бочек с бензином будет загружено в машину.

Примеры

стандартный ввод	стандартный вывод
5 A 6 B 7	5
5 B 6 A 7	2
5 A 6 B 3	3
5 B 6 A 3	0

Замечание

В первом примере в машину загрузят все бочки с бензином и еще две бочки с керосином.

Во втором примере в машину загрузят все бочки с керосином (они стоят левее) и еще две бочки с бензином.

Задача Е. Подготовка к контесту

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

При подготовке к этому контесту, Миша решил прорешать несколько других соревнований с похожим по сложности набором задач. Сегодня день очередной тренировки.

Тренировка состоит из n задач и длится t часов. В первый час Миша всегда решает a задач, а затем тратит на каждую из оставшихся задач b минут. Требуется определить, успеет ли Миша решить все задачи тренировки.

Если Миша успеет решить все задачи, нужно вывести время, которое у него останется в запасе. Если Миша не успеет решить все задачи, то нужно вывести количество нерешенных им задач и сколько времени ему не хватит.

Формат входных данных

В единственной строке даны четыре натуральных числа n, t, a и b ($5 \leq n \leq 100$, $2 \leq t \leq 24$, $1 \leq a < n$, $10 \leq b \leq 60$).

Формат выходных данных

В первой строке выведите «YES» (без кавычек), если Миша решит все задачи и у него останется время. Выведите «NO» (без кавычек), если Миша не успеет решить все задачи. Выведите «ЕХАСТ!» (без кавычек), если длительности контеста хватит ровно на n задач, и времени в запасе не останется.

Если у Миши останется время, то во второй строке нужно вывести сколько минут у него останется после решения всех задач. Если Миша не успеет решить все задачи, на второй строке нужно вывести количество минут, которых не хватило Мише, а на третьей строке вывести количество задач, которые не успеет решить Миша.

Примеры

стандартный ввод	стандартный вывод
12 5 7 45	YES 15
12 3 7 45	NO 105 3
10 3 6 30	ЕХАСТ!

Задача F. Игра на повышение

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

У вас имеется три кучки камней. За один ход вы можете сделать одно из следующих действий:

- Выбрать одну кучку, и доложить в неё два камня
- Выбрать две кучки, и доложить в каждую из них по одному камню

Ваша задача уровнять количество камней в кучках. Нужно определить возможно ли это, и если возможно, какое минимальное количество ходов вам для этого потребуется?

Формат входных данных

Даны три целых числа a, b, c ($0 \leq a, b, c \leq 50$) — число камней в каждой из кучек.

Формат выходных данных

Выведите одно число — искомое количество ходов. Если уровнять кучки невозможно, выведите -1 .

Пример

стандартный ввод	стандартный вывод
1 4 3	2

Замечание

В примере можно первым ходом доложить в первую кучку два камня, а вторым ходом добавить в первую и третью кучки по одному камню. В результате во всех кучках будет по 4 камня.

Задача G. Хитмейкер

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

Шерали Штенморген решил записать очередной хит. Трек состоит из некоторого количества звуков, для простоты звуки пронумерованы натуральными числами. Для размещения трека на платформе mtunes сумма звуков должна быть равна заданному числу n . Исследования показали, что трек набирает тем больше прослушиваний, чем больше произведение звуков в треке.

Помогите бедному артисту написать генератор хитов на все времена. По заданному числу n ваша программа должна выдавать номера звуков, входящих в состав композиции, которая наберет максимальное возможное количество прослушиваний.

Формат входных данных

Дано одно натуральное число n ($1 \leq n \leq 200$).

Формат выходных данных

Выведите натуральные числа через пробел в одной строке — искомый набор звуков, дающий максимальное число прослушиваний. Если правильных ответов несколько, выведите любой.

Примеры

стандартный ввод	стандартный вывод
4	2 2
5	2 3

Замечание

В первом примере $2 + 2 = 4$ и $2 * 2 = 4$ — максимальное достижимое произведение.

Во втором примере $2 + 3 = 5$ и $2 * 3 = 6$ — максимальное достижимое произведение.

Задача Н. Тривиальность

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

Для одного университетского проекта Сергею потребовалось реализовать алгоритм оценки числа на «тривиальность». «Тривиальность» числа N — это отношение суммы всех его делителей меньших N к самому числу N . Например, тривиальность числа $T(20) = \frac{1+2+4+5+10}{20} = 1.1$

С этой задачей Сергей справился легко, поэтому он ее чуть-чуть усложнил. Сергей хотел бы среди всех чисел A таких, что $L \leq A \leq R$, найти числа с максимальной и минимальной тривиальностью. Так как все такие числа находить — задача трудоёмкая, среди чисел с одинаковой минимальной или максимальной тривиальностью нужно найти наименьшие.

Формат входных данных

Вводятся два натуральных числа L и R ($1 \leq L \leq R \leq 100\,000$) — границы отрезка, на котором нужно рассматривать числа.

Формат выходных данных

Выведите через пробел два целых числа X и Y , где X — минимальное число из всех чисел с минимальной тривиальностью, а Y — минимальное число из всех чисел с максимальной тривиальностью.

Примеры

стандартный ввод	стандартный вывод
24 28	25 24
12 453	449 360

Замечание

Если вы пишете на языке Python, то отправляйте решение под PyPy.

Тривиальность числа 25 равна $T(25) = \frac{1+5}{25} = 0.24$.

Тривиальность числа 24 равна $T(24) = \frac{1+2+3+4+6+8+12}{24} = 1.5$.

Задача I. Трудно запомнить дни рождения

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

Миша хочет запомнить дни рождения всех своих n друзей, чтобы знать кого поздравлять каждый день. К сожалению, иногда возникают ситуации, когда у нескольких друзей день рождения в один день. Это может сбить Мишу с толку, поэтому он решил запомнить день рождения друга, который ему больше всего нравится, в случае если у нескольких друзей день рождения в один день.

Имея список дней рождения для каждого из друзей, и насколько Миша любит каждого друга, выведите, дни рождения каких друзей Миша запомнит.

Формат входных данных

В первой строке дано одно целое число n ($1 \leq n \leq 2000$) — количество друзей Миши.

Далее в n строках идут описания друзей. Сначала идет имя — строка s начинающаяся с заглавной буквы английского алфавита, за которой следуют строчные буквы английского алфавита ($1 \leq |s| \leq 10$). Далее идет одно целое число c_i ($0 \leq c_i \leq 10^5$) — характеристика отношения Миши к этому другу. Чем больше это число, тем больше ему нравится этот друг. Затем идет дата рождения этого друга в формате DD/MM (DD — день рождения, MM — месяц рождения, каждое число состоит ровно из двух цифр).

Все дни рождения — реальные даты для 2020 (високосного) года. Все c_i различны. Все имена друзей различны.

Формат выходных данных

В первой строке выведите одно число k — количество друзей, дни рождения которых запомнит Миша. Далее выведите k строк — имена этих детей в алфавитном порядке.

Примеры

стандартный ввод	стандартный вывод
3 Sanna 1 16/03 Simon 2 16/03 Saga 3 14/10	2 Saga Simon
10 Oden 78 03/12 Tor 132 14/05 Freja 10000 14/05 Loke 512 12/10 Hel 14 04/05 Fjorgynn 532 13/05 Hildegun 500 13/05 Vindsval 17 03/12 Snotra 20 04/05 Kvaser 420 03/12	5 Fjorgynn Freja Kvaser Loke Snotra

Задача J. Богдан и подстроки

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

На свой день рождения Богдан получил строку, в которой было не больше двух различных букв английского алфавита. Богдан придумал функцию *maxdiff* для таких строк: она возвращает разницу между количеством этих двух букв. Если в строке все буквы одинаковые, функция, соответственно, возвращает длину строки.

Богдан обожает подстроки. Теперь Богдан хочет узнать максимальное значение функции *maxdiff* для всех подстрок подаренной ему строки. Он запустил эту функцию для всех подстрок, но так и не дождался ответа. Помогите ему узнать максимальное значение его функции.

Формат входных данных

В первой строке задано число n ($1 \leq n \leq 10^5$) — количество символов в строке. Во второй строке задана сама строка. Строка состоит из строчных букв английского алфавита и содержит не более двух различных букв.

Формат выходных данных

Требуется вывести одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
5 aaaaa	5
6 хухуху	1
15 pddpddpdddppdppd	4

Замечание

Подстрока — некоторая непустая подпоследовательность подряд идущих символов строки.

В первом примере нужно взять подстроку, состоящую из всей строки. Во втором примере можно взять, например, подстроку с 2 по 6 символ (ухуху). В третьем примере максимальное значение дает подстрока с 9 по 14 символ (pprdpp).

Задача К. Маленький читер

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

Сегодня у Миши очень важная контрольная по математике. Работа состоит из большого числа простых заданий, на которые нужно только выписать ответ. Дальше учитель проверит эти ответы и выставит оценку.

В самом конце работы учитель намекнул, что оценка напрямую будет зависеть от наибольшего общего делителя ответов: чем больше он будет, тем лучше будет оценка. И тут же принялся собирать работы. У Миши совсем мало времени, он может успеть исправить не более одного ответа. Помогите ему определить, какой наибольший общий делитель ответов он таким образом может получить?

Формат входных данных

В первой строке дано натурально число n ($2 \leq n \leq 100\,000$) — количество заданий в контрольной.

Во второй строке даны n натуральных чисел a_i ($1 \leq a_i \leq 10^9$) — ответы Миши на каждое из заданий.

Формат выходных данных

Выведите одно число — максимальный наибольший общий делитель, который может получить Миша, исправив не более одного ответа.

Пример

стандартный ввод	стандартный вывод
3 30 2 15	15

Замечание

В примере Миша может заменить 2 на 45, тогда $\text{НОД}(30, 45, 15)$ будет равен 15.

Задача L. Стрельба по телевизорам

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

Вам только что привезли на склад новые прямоугольные телевизоры в коробках. Вы старательно составляете эти коробки вдоль стенки, чтобы между коробками не было щелей. Телевизоры довольно дорогие, поэтому их нельзя ставить друг на друга. В результате, у вас получается цепочка из коробок, где каждый следующий телевизор стоит вплотную к предыдущему.

Скоро на склад нападут разбойники, и ваш шпион узнал, в какие точки стены, вдоль которой стоят телевизоры, разбойники будут стрелять. Для удобства, вы ввели систему координат по этой стене, обозначив за начало координат нижний угол склада.

Теперь, по заданному расположению коробок с телевизорами и траекториям пуль, вы хотите определить, сколько пуль попадут или хотя бы заденут телевизоры. Телевизоры настолько тонкие, что можно считать будто траектория каждой пули перпендикулярна плоскости экрана телевизора.

Формат входных данных

В первой строке даны два натуральных числа n и m ($1 \leq n, m \leq 100\,000$) — количество телевизоров и пуль соответственно.

Следующие n строк содержат по два натуральных числа w_i и h_i ($1 \leq w_i, h_i \leq 10\,000$) — ширину и высоту коробок в том порядке, в котором они составлены вдоль стены. Нижний угол первого телевизора располагается в начале координат стены склада.

Следующие m строк содержат по два целых числа x_j и y_j ($0 \leq x_j, y_j \leq 10^9$) — координаты точек, в которых траектория пули пересекается с плоскостью телевизоров.

Формат выходных данных

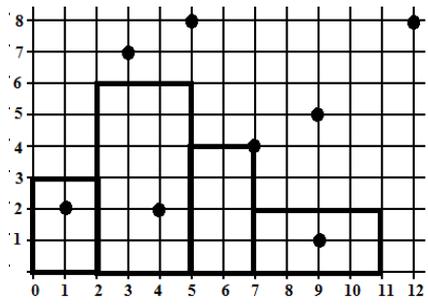
Выведите одно число — количество пуль, которые попадут или хотя бы заденут коробки с телевизорами.

Примеры

стандартный ввод	стандартный вывод
4 8 2 3 3 6 2 4 4 2 1 2 3 7 4 2 5 8 7 4 9 1 9 5 12 8	4
1 4 2 3 1 2 3 7 1 2 0 3	3

Замечание

На картинке показано расположение коробок и пуль из первого примера



Задача М. Урок физкультуры

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 256 мегабайт

Однажды Дима пришел на урок физкультуры. На этом уроке присутствуют дети из нескольких классов.

Как известно, физкультура всегда начинается с построения. Диме надоело строиться в шеренгу по росту, поэтому он предложил новую фигуру. К счастью, учеников на физкультуре ровно $n \times m$, поэтому они могут составить прямоугольник размером n на m .

Преподаватель согласился на такое нововведение при том, что следующие условия будут выполнены:

- Каждый ученик наденет колпак определенного цвета.
- Цвета колпаков учеников из одного класса должны совпадать.
- Цвета колпаков учеников, стоящих в одном столбце, должны совпадать.

Ученики уже построились в прямоугольник, и Дима хочет сделать построение как можно более красочным. Помогите Диме и определите максимальное возможное количество различных цветов колпаков.

Формат входных данных

В первой строке входных данных содержится три целых числа n , m и k ($1 \leq n, m \leq 700$, $1 \leq k \leq 26$) — высота прямоугольника, ширина прямоугольника и количество различных классов.

В последующих n строках содержится по m символов, каждый из которых является одной из первых k заглавных букв латинского алфавита. Символ в строке i на позиции j означает букву класса ученика, стоящего на этом месте.

Формат выходных данных

Выведите единственное целое число — максимальное возможное число различных цветов колпаков.

Примеры

стандартный ввод	стандартный вывод
2 3 2 AAB ABB	1
2 2 3 AC BC	2
2 3 3 ABC ABC	3
3 5 5 ABECE BCDAE CADBD	2

Замечание

В первом примере все ученики должны надеть колпак одного цвета.

Во втором примере ученики классов «А» и «В» должны надеть колпак одного цвета, а ученики класса «С» могут надеть колпак другого цвета.

Задача N. Вова на даче

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

Вова собирался утром поехать с дачи домой, но, к своему удивлению, обнаружил, что рейсовый автобус до его дачи больше не ходит вследствие оптимизации рейсовых автобусов. Не зная, чем себя занять, Вова решил вскопать ещё одну грядку.

Всё шло своим чередом, пока лопата не лязгнула обо что-то твёрдое. Подкопав со всех сторон подозрительный объект, Вова вытащил на свет сундук с сокровищами. Всё было бы замечательно и прекрасно, если бы на сундуке не стоял замок. Чтобы замок открыть, требовалось ответить на вопрос «сколько чисел от L до R включительно имеют в битовом представлении без лидирующих нулей ровно k нулевых битов?».

Вова в замешательстве: с одной стороны, хочется рассказать всем о находке, но с другой, надо бы решить загадку. Поколебавшись, Вова доверил решение задачи вам, а сам побежал рассказывать соседям радостную новость. Помогите Вове открыть сундук.

Формат входных данных

В единственной строке содержатся три целых числа L , R и k ($1 \leq L \leq R \leq 10^{18}$, $0 \leq k \leq 60$) — границы рассматриваемого отрезка чисел и количество нулевых битов в записи.

Формат выходных данных

Выведите единственное число — количество чисел от L до R включительно, в двоичной записи которых без лидирующих нулей ровно k нулевых битов.

Пример

стандартный ввод	стандартный вывод
8 23 2	6

Замечание

Двоичные записи чисел от 8 до 23: 1000, **1001**, **1010**, 1011, **1100**, 1101, 1110, 1111, 10000, 10001, 10010, **10011**, 10100, **10101**, **10110**, 10111. Жирным выделены числа с двумя нулями в битовой записи (двоичной системе счисления).

Задача О. Заснеженные дороги

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

Страну Берляндию можно представить как неориентированный граф с n вершинами с m ребрами, где вершины соответствуют городам, а ребра – двусторонним дорогам, соединяющим города. Дорога с номером i ($1 \leq i \leq m$) соединяет города с номерами a_i и b_i ($1 \leq a_i, b_i \leq n$) и имеет *эффективность* c_i . Эффективность дороги – численная характеристика, описывающая сколько миллиметров снега может выпасть на нее прежде, чем по ней невозможно будет проехать. Берляндия очень маленькая, поэтому когда идет снег, можно считать, что на все дороги выпадает одинаковое количество миллиметров снега.

Назовем город i *опасным* при выпадении на дороги C миллиметров снега, если выполнены следующие два условия:

- После удаления из графа ребер с номерами j , такими что $c_j \leq C$, он остается связным. Назовем такой граф *обновленным*.
- После удаления из обновленного графа вершины i он перестает быть связным.

Совсем недавно в Берляндии вышел указ, в котором сообщалось, что если в какой-то момент времени одновременно будет k или более опасных городов, то на обсуждение будет вынесен вопрос о реструктуризации системы дорог. После этой новости городским активистам стало интересно, какое минимальное количество снега должно выпасть, чтобы по крайней мере k городов стали опасными. Именно на этот вопрос Вам и предстоит ответить.

Известно, что изначально все города образуют связный граф.

Формат входных данных

Первая строка содержит три целых числа n , m и k ($2 \leq n \leq 5 \cdot 10^4$, $1 \leq m \leq 2 \cdot 10^5$, $1 \leq k \leq n$) – количество городов, количество дорог и минимальное количество опасных городов, которое необходимо для проведения реструктуризации системы дорог.

Следующие m строк содержат описания дорог – три целых числа a_i , b_i , c_i ($1 \leq a_i, b_i \leq n$, $1 \leq c_i \leq 10^9$) – номера городов, соединенных дорогой, и эффективность дороги. Обратите внимание, что между одной парой городов может быть более одной дороги, а также дорога может соединять город с самим собой.

Формат выходных данных

В первой строке выведите два целых числа X и Y , разделенных пробелом – минимальное количество миллиметров снега, которое должно выпасть, чтобы по крайней мере k городов стали опасными, и сколько на самом деле городов опасны при таком количестве снега.

В следующей строке выведите Y целых чисел, отсортированных в порядке возрастания, – номера городов, представляющих опасность при условии, что выпало X миллиметров снега. Если никакое количество снега не приводит к ситуации, когда k или более городов являются опасными, выведите $X = -1$ и $Y = 0$.

Примеры

стандартный ввод	стандартный вывод
8 13 3 1 2 19 1 3 13 2 3 5 4 2 9 2 5 7 3 5 17 6 3 8 3 8 12 4 7 11 5 7 5 7 2 10 7 6 13 8 5 16	8 4 1 2 3 7
3 3 1 1 2 42 2 3 42 1 3 666	-1 0

Замечание

В первом примере, если на дороги выпадет 8 миллиметров снега, то города все еще будут представлять связный граф. При удалении одного любого города среди городов с номерами 1, 2, 3, 7 граф перестанет быть связным.

Во втором примере, если на дороги выпадет 42 миллиметров снега, то города с номерами 1 и 2 перестанут быть достижимыми друг из друга. По первому условию определения опасности города получаем, что никакой город не является опасным.