

Задача А. Автобусы

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

В Берляндии плачевная ситуация с междугородним автобусным сообщением. Во всей стране есть всего три автобусных маршрута, по каждому из которых курсирует лишь один автобус. В первый день нового года ровно в полночь все три автобуса отправляются по своим маршрутам из столицы Берляндии. Известно, что первому автобусу на то, чтобы проехать весь маршрут и вернуться в столицу требуется a минут, второму — b минут, а третьему — c минут. Таким образом, первый автобус отправляется из столицы Берляндии в моменты времени $0, a, 2a, 3a, \dots$, второй — в моменты времени $0, b, 2b, 3b, \dots$, а третий в моменты времени $0, c, 2c, 3c, \dots$.

Момент времени называется *подходящим для пересадки*, если в этот момент все три автобуса отправляются из столицы Берляндии. Например если $a = 1, b = 2, c = 1$, то моменты времени 0 и 2 являются подходящими для пересадки, а момент времени 1 не является, потому что в этот момент времени второй автобус находится в пути. Берляндия — особая страна с особым измерением времени, поэтому в берляндских сутках ровно t минут. Это означает, что в первый день происходят все моменты времени с 0 -го по $(t-1)$ -й включительно, во второй день — с t -го по $(2t-1)$ -й включительно, в третий — с $2t$ -го по $(3t-1)$ -й включительно и так далее.

Министерство транспорта Берляндии заинтересовалось, сколько подходящих для пересадки моментов времени произойдет в d -й день в Берляндии. К сожалению, местные чиновники заняты другими делами, поэтому ответить на этот вопрос было поручено вам.

Формат входных данных

В пяти строках заданы пять целых чисел a, b, c, t и d ($1 \leq a, b, c \leq 10^6, 1 \leq t, d \leq 10^9$) — время полного прохождения маршрута первым, вторым и третьим автобусами, соответственно, количество минут в сутках и номер дня, которым интересуется министерство транспорта Берляндии.

Формат выходных данных

Выведите одно целое число — количество подходящих для пересадки моментов времени в d -й день.

Примеры

стандартный ввод	стандартный вывод
1 2 1 3 1	2
2 3 4 7 2	1
2 3 4 3 3	0

Замечание

В первом примере сутки делятся на 3 минуты, поэтому все моменты времени в день с номером 1 — это $0, 1, 2$, из них моменты времени 0 и 2 являются подходящими для пересадки.

Во втором примере рассматриваются вторые сутки с моментами времени 7, 8, 9, 10, 11, 12, 13. Первый автобус отправляется в моменты времени 8, 10, 12, второй автобус — в моменты времени 9 и 12, а третий — в моменты времени 8 и 12. Таким образом, только момент времени 12 является подходящим для пересадки.

В третьем примере нет ни одного подходящего для пересадки момента времени.

Система оценки

В данной задаче 50 тестов, помимо тестов из условия, каждый из них оценивается в 2 балла. Результаты работы ваших решений на первых 35 тестах будут доступны во время соревнования. Результаты работы на остальных 15 будут доступны после окончания соревнования.

Решения, корректно работающие при $1 \leq a, b, c, t, d \leq 100$, наберут не менее 20 баллов.

Решения, корректно работающие при $t = 1$, наберут не менее 10 баллов.

Решения, корректно работающие при $1 \leq t \leq 10^6$, наберут не менее 50 баллов.

Решения, корректно работающие при $d = 1$, наберут не менее 20 баллов.

Задача В. День рождения

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

У ковбоя Влада день рождения! На праздник собрались n детей. Чтобы поздравить ковбоя, дети решили водить вокруг Влада хоровод. Среди детей, пришедших к Владу, есть и высокие, и низкие, поэтому если они встанут в хороводе как угодно, многим из них может быть неудобно, потому что если в хороводе рядом стоят очень высокий и очень низкий ребёнок, им трудно держаться за руки. Поэтому дети решили встать в хоровод так, чтобы максимальная разность ростов двух соседних детей была минимальной.

Более формально, пусть n детей выстроились в хоровод. Пронумеруем их целыми числами от 1 до n так, чтобы справа от ребёнка с номером i стоял ребёнок с номером $i + 1$, а справа от ребёнка с номером n стоял ребёнок с номером 1. Тогда *неудобством* этого хоровода назовём максимальную разность между ростом детей, которые стоят рядом. Обратите внимание, что разностью в росте двух детей называется разность между ростом более высокого и более низкого ребёнка, таким образом, разность в росте двух детей всегда неотрицательна.

Помогите детям и определите, в каком порядке им надо выстроиться в круг, чтобы минимизировать неудобство получившегося хоровода. Обратите внимание, что все n детей должны оказаться в хороводе.

Формат входных данных

В первой строке содержится одно целое число n ($2 \leq n \leq 10^5$) — количество детей, которые пришли на день рождения ковбоя Влада.

Во второй строке заданы n целых чисел a_i ($1 \leq a_i \leq 10^9$) — рост каждого из детей. Рост детей задан в нанометрах и уменьшен на 10^9 , таким образом, рост ребёнка с $a_i = 1$ чуть выше метра, а рост ребёнка с $a_i = 10^9$ составляет два метра.

Формат выходных данных

Выведите n целых чисел — значения роста детей в порядке, в котором они должны встать в хоровод. В этом порядке соседними будут дети с номерами i и $i + 1$, а также дети с номерами 1 и n . Если оптимальных хороводов несколько, то выведите любой из них.

Примеры

стандартный ввод	стандартный вывод
5 2 1 1 3 2	1 2 3 2 1
3 30 10 20	10 20 30

Замечание

В первом примере неудобство хоровода равно 1, так как разность в росте между соседними детьми равна 1, 1, 1 и 0 соответственно. Обратите внимание, что последовательности $[2, 3, 2, 1, 1]$, $[3, 2, 1, 1, 2]$ задают те же хороводы и отличаются только выбором ребёнка с номером 1.

Во втором примере неудобство хоровода равно 20, так как разность в росте детей высотой 10 и 30 равна 20.

Система оценки

В данной задаче 25 тестов, помимо тестов из условия. Каждый из них оценивается в 4 балла. Результаты работы ваших решений на первых 15 тестах будут доступны во время соревнования. Результаты работы на остальных 10 будут доступны после окончания соревнования.

Решения, корректно работающие при $1 \leq n \leq 5$, наберут не менее 20 баллов.

Решения, корректно работающие при $a_i = i$, наберут не менее 40 баллов.

Задача С. Произведение строк

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

Рома и Денис отправились на соревнование по программированию. В долгой дороге ребята вспоминали операции над строками. Денис сказал, что в Python строки можно умножать на число, тогда Рома, программирующий на C++, решил придумать операцию перемножения строк. По версии Ромы, умножение строки s длины n на строку t обозначается как $s \cdot t$ и равно строке $t + s_1 + t + s_2 + \dots + t + s_n + t$, где s_i обозначает i -й символ строки s , а знаком «+» обозначено сложение (конкатенация) строк. Например, произведением строк «abc» и «de» является строка «deadebdecde», а произведением строк «z» и «ab» является строка «abzab». Обратите внимание, что, в отличие от умножения чисел, произведение строк s и t , вообще говоря, не равно произведению строк t и s .

Денис решил продолжить мысль Ромы — он, как ценитель прекрасного, решил определить *красоту* строки как максимальную длину подряд идущей группы одинаковых букв. Например, красота строки «хаууааабса» равна 3, так как самая длинная группа подряд идущих одинаковых букв — это «ааа», а красота строки «qwewqwer» равна 1, потому что все соседние буквы в ней различны.

Чтобы развлечь Дениса, Рома написал ему на листочке n строк $p_1, p_2, p_3, \dots, p_n$ и попросил его вычислить красоту строки $(\dots((p_1 \cdot p_2) \cdot p_3) \cdot \dots) \cdot p_n$. Денис не до конца понял, как работает умножение Ромы, но не хочет признаваться в этом, поэтому просит посчитать красоту этой строки вас. Рома знает, что Денис слишком впечатлительный, поэтому гарантирует, что красота полученной строки не превосходит 10^9 .

Формат входных данных

В первой строке содержится число n ($1 \leq n \leq 100\,000$) — количество строк, которые написал Рома.

В следующих n строках содержатся непустые строки p_1, p_2, \dots, p_n , состоящие из маленьких букв английского алфавита.

Гарантируется, что суммарная длина строк не превосходит 100 000, а также, что красота произведения всех строк не превосходит 10^9 .

Формат выходных данных

Выведите одно целое число — красоту произведения строк.

Примеры

стандартный ввод	стандартный вывод
3 a b a	3
2 bnn a	1

Замечание

В первом примере произведение трёх строк равно «abaaaba».

Во втором примере произведение двух строк равно «abanana».

Система оценки

В данной задаче 50 тестов, помимо тестов из условия, каждый из них оценивается в 2 балла. Результаты работы ваших решений на первых 30 тестах будут доступны во время соревнования. Результаты работы на остальных 20 будут доступны после окончания соревнования.

Обозначим за L длину произведения всех строк.

Решения, корректно работающие при $1 \leq n \leq 7$, $1 \leq L \leq 100$ наберут не менее 20 баллов.

Решения, корректно работающие при $1 \leq n \leq 15$, $1 \leq L \leq 100\,000$, наберут не менее 40 баллов.

Решения, корректно работающие при $n = 2$, наберут не менее 20 баллов.

Задача D. Выбор гурмана

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

Участникам, использующим языки Python2/Python3, рекомендуется отправлять решения на проверку с использованием интерпретаторов PyPy2/PyPy3.

Гурман Яблочков работает редактором известного гастрономического издания. Он разъезжает по всему миру, дегустируя новые изыски именитых шеф-поваров самых фешенебельных ресторанов высокой кухни. У Яблочкова есть свой фирменный способ ревью: в каждом заведении Яблочков заказывает два набора блюд в разные дни. Все заказанные блюда разные, так как Яблочков не любит есть одно и то же. Для каждой пары блюд из разных наборов он точно помнит, какое из них лучше по его ощущениям, или что они одинаковы по вкусовым качествам. Затем гурман оценивает каждое блюдо положительным целым числом.

Однажды, во время очередной ревизии, ресторан кельтской средневековой кухни «Пуассон», подающий суп из каштанов с пихтой, теплый содовый хлеб, пряный лимонный пирог и другую народную еду, очень приятно удивил гурмана своим разнообразным меню, и эксперт заказал слишком много. Поэтому ему нужна помощь в оценке блюд.

Гурман в первый день дегустировал набор из n блюд, во второй день набор из m блюд. Он составил таблицу ощущений a размером $n \times m$, в которой описал свои впечатления. Если по мнению эксперта блюдо i из первого набора лучше блюда j из второго набора, то a_{ij} равно «>», в противоположном случае a_{ij} равно «<». Блюда также могут быть равны по уровню, тогда a_{ij} равно «=».

Теперь Яблочков хочет, чтобы вы помогли ему оценить каждое блюдо в наборах целым положительным числом. Так как Яблочков очень строгий дегустатор, то постарается оценить блюда так, чтобы максимальная из оценок была как можно меньше. В то же время Яблочков очень справедливый, поэтому никогда не оценивает блюда так, чтобы это шло вразрез с его ощущениями. Другими словами, если a_{ij} равно «<», то оценка блюда i из первого набора должна быть меньше оценки блюда j из второго набора, если a_{ij} равно «>», то должна быть больше, а если a_{ij} равно «=», то должна совпадать.

Помогите Яблочкову выставить оценки каждому блюду из обоих наборов так, чтобы это согласовывалось с его ощущениями, или определите, что это невозможно.

Формат входных данных

В первой строке заданы два целых числа n и m ($1 \leq n, m \leq 1000$) — количества блюд в каждый из двух дней.

Каждая из следующих n строк содержит строку из m символов. j -й символ в i -й строке задаёт значение a_{ij} . Все строки состоят только из символов «<», «>» и «=».

Формат выходных данных

В первой строке выведите «Yes» (без кавычек), если можно сделать корректную оценку всем блюдам и «No» (без кавычек), если иначе.

В случае, если сделать корректную оценку можно, во второй строке выведите n целых чисел — оценки блюд первого набора, а в третьей строке m целых чисел — оценки блюд второго набора.

Примеры

стандартный ввод	стандартный вывод
3 4 >>>> >>>> >>>>	Yes 2 2 2 1 1 1 1
3 3 >>> <<< >>>	Yes 3 1 3 2 2 2
3 2 == =< ==	No

Замечание

В первом примере все блюда первого дня лучше всех блюд второго дня. Значит, самой высокой оценкой будет 2 для всех блюд первого дня.

В третьем примере таблица противоречива: нет ни одного набора оценок, который удовлетворял бы ей.

Система оценки

В данной задаче 50 тестов, помимо тестов из условия, каждый из них оценивается в 2 балла. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при $1 \leq n, m \leq 3$, наберут не менее 20 баллов.

Решения, корректно работающие при $1 \leq n, m \leq 50$, никакой a_{ij} не равен «=», наберут не менее 10 баллов.

Решения, корректно работающие при $1 \leq n, m \leq 50$, наберут не менее 50 баллов.

Решения, корректно работающие при $1 \leq n, m \leq 1000$, никакой a_{ij} не равен «=», наберут не менее 30 баллов.

Решения, корректно работающие только на тестах из условия и тестах с ответом «No», оцениваются в 0 баллов.

Задача Е. Ася и котят

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

Ася очень любит животных. Недавно она приобрела n котят, дала им числовые идентификаторы от 1 до n и поселила в вольере. Вольер представляет собой ряд из n ячеек, также пронумерованных от 1 до n . Соседние ячейки разделены сетчатыми перегородками, всего в вольере $n - 1$ перегородок. Изначально в каждой ячейке поселился ровно один котёнок с некоторым номером.

Наблюдая за котятками, Ася заметила, что они очень дружелюбны и некоторые пары живущих в соседних ячейках котят очень хотят играть друг с другом. Чтобы не лишать их этого удовольствия, Ася стала вынимать перегородки между соседними ячейками, делая их более крупными.

В i -й день Ася делала следующее.

- Обращала внимание, что какие-то котята x_i и y_i , в i -й день живущие в соседних ячейках, хотят играть.
- Удаляла перегородку между этими ячейками, превращая их в одну, в которой оказывались все котята из двух прежних ячеек.

Поскольку Ася не возвращала перегородки, через $n - 1$ день вольер стал единой ячейкой, в которой обитали все котята. Будучи очень педантичной, Ася записывала в специальный журнал идентификаторы котят x_i и y_i для каждого из $n - 1$ дней.

Вам в руки попал журнал с этой информацией, однако вам неизвестно, как котята были поселены в ячейки изначально. Найдите любое расселение котят по n исходным ячейкам, не противоречащее данным в журнале.

Формат входных данных

В первой строке задано целое число n ($2 \leq n \leq 150\,000$) — количество котят.

В следующих $n - 1$ строках заданы пары целых чисел x_i, y_i ($1 \leq x_i, y_i \leq n, x_i \neq y_i$) — идентификаторы котят, между ячейками которых была удалена перегородка в день i . Гарантируется, что котята x_i и y_i не находятся в одной ячейке по итогам предыдущих объединений ячеек.

Формат выходных данных

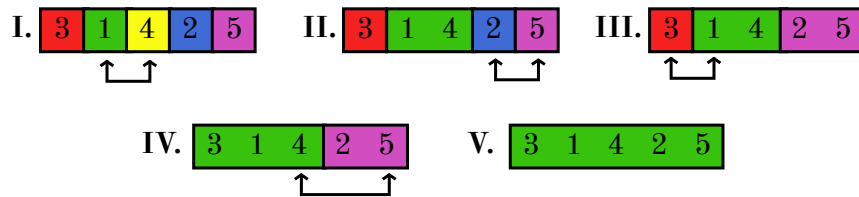
Выведите n различных целых чисел p_i ($1 \leq p_i \leq n$), где p_i — идентификатор котёнка, который изначально жил в ячейке с номером i . Если возможных вариантов ответа несколько, выведите любой из них.

Пример

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

Замечание

В ответе к примеру приведено одно из возможных изначальных расселений котят, существуют и другие варианты ответа. На изображении ниже показано, как происходило объединение ячеек для этого варианта исходного размещения котят. Обратите внимание, что при таком размещении котят, подружившиеся в каждый из дней согласно журналу Аси, находятся в **соседних** ячейках.



Система оценки

В данной задаче 50 тестов, помимо тестов из условия, каждый из них оценивается в 2 балла. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при $1 \leq n \leq 10$, наберут не менее 20 баллов.

Решения, корректно работающие при $1 \leq n \leq 100$, наберут не менее 40 баллов.

Решения, корректно работающие при $1 \leq n \leq 2000$, наберут не менее 70 баллов.

Задача F. Самая опасная акула

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	768 мегабайт

Как известно, не только люди любят принимать участие в различных соревнованиях. Среди рыб и моллюсков также проводится множество состязаний на дне океана. Акула Семён участвует в самом престижном соревновании Мирового океана на звание самой опасной акулы. Во время этого состязания акулы соревнуются в различных дисциплинах: плавание на скорость, маскировка, навигация по картам и многим другим. Сейчас Семён проходит испытание под названием «разрушение».

Во время этого испытания перед акулой ставят m доминошек. Все доминошки стоят на одной прямой, однако высоты доминошек могут различаться. Расстояние между соседними доминошками равно 1. Кроме того, у каждой доминошки есть своя стоимость, выраженная целым числом. Цель акулы — уронить все доминошки. Для этого акула может толкнуть любую доминошку влево или вправо, после чего она начнёт падать в этом направлении. Если во время падения доминошка задевает другие, они также начинают падать в ту же сторону, в которую падала исходная, таким образом начинается цепная реакция, в результате которой может упасть множество доминошек. Падающая доминошка задевает другую, если и только если расстояние между ними **строго меньше** высоты падающей доминошки, причем доминошки не обязательно должны быть соседними.

Разумеется, любая акула справится с тем, чтобы уронить все доминошки таким образом, поэтому оценивается не факт разрушения, а его стоимость. Стоимостью разрушения называется сумма стоимостей доминошек, которые акуле придётся толкнуть, чтобы все доминошки упали.

Семён уже победил в прошлых испытаниях, однако недостаточно умён, чтобы победить в этом. Помогите Семёну и определите, какая минимальная суммарная стоимость доминошек, которые ему придётся толкнуть, чтобы все доминошки упали.

Формат входных данных

Для уменьшения размера ввода высоты и стоимости доминошек задаются при помощи блоков.

В первой строке заданы два целых числа n и m ($1 \leq n \leq 250\,000$, $1 \leq m \leq 10^7$) — количество блоков и суммарное количество доминошек, которые надо уронить Семёну, соответственно.

Затем следует описание n блоков. Описание каждого блока состоит из трёх строк.

В первой строке описания блока с номером i содержится целое число k_i ($1 \leq k_i \leq 250\,000$, $\sum_{i=1}^n k_i \leq 250\,000$) — количество доминошек в блоке.

Во второй строке описания блока с номером i содержатся k_i целых чисел $a_{i,j}$ ($1 \leq a_{i,j} \leq m$) — высоты доминошек в блоке.

В третьей строке описания блока с номером i содержатся k_i целых чисел $c_{i,j}$ ($1 \leq c_{i,j} \leq 100\,000$) — стоимости доминошек в блоке.

Далее следует описание последовательности доминошек, которые надо уронить Семёну, в порядке слева направо.

В первой строке описания последовательности задано целое число q ($n \leq q \leq 250\,000$) — количество блоков в последовательности доминошек, которые надо уронить.

В следующих q строках содержатся пары целых чисел id_i mul_i ($1 \leq id_i \leq n$, $1 \leq mul_i \leq 100\,000$), обозначающие, что очередные k_{id_i} в порядке слева направо доминошек — это доминошки блока id_i , чьи стоимости были умножены на число mul_i .

Гарантируется, что $\sum_{i=1}^q k_{id_i} = m$, а также, что каждый блок встречается хотя бы один раз в последовательности доминошек, которые требуется уронить, то есть для всех i от 1 до n найдется j такое, что $id_j = i$.

Формат выходных данных

Выведите одно целое число — минимальную суммарную стоимость доминошек, которые надо толкнуть, чтобы все доминошки упали.

Примеры

стандартный ввод	стандартный вывод
<pre> 2 7 3 1 2 2 1 2 1 1 3 2 3 2 2 1 3 1 1 </pre>	5
<pre> 1 1 1 1 100000 1 1 100000 </pre>	10000000000

Замечание

В первом примере перед Семёном стоят 7 доминошек. Их высоты равны [3, 1, 2, 2, 1, 2, 2], а стоимости равны [4, 3, 6, 3, 1, 2, 1]. Сначала Семёну следует уронить доминошку с номером 7 влево, она упадёт и заденет доминошку с номером 6. Доминошка 6, падая, заденет доминошку с номером 5, которая упадёт, но не заденет другие доминошки. Затем Семён должен уронить доминошку с номером 1 вправо, она, падая, заденет доминошки с номерами 2 и 3, а доминошка 3, падая, заденет доминошку 4, таким образом все доминошки упадут.

Во втором примере перед Семёном стоит одна доминошка стоимостью 10000000000.

Система оценки

Тесты к этой задаче состоят из семи групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов **необходимых** групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Дополнительные ограничения		Необх. группы	Комментарий
		m	$c_{i,j}, mul_i$		
0	0	—	—	—	Тесты из условия.
1	11	$1 \leq m \leq 10$	—	0	—
2	9	$1 \leq m \leq 500$	—	0, 1	—
3	22	$1 \leq m \leq 5000$	—	0, 1, 2	—
4	12	$1 \leq m \leq 100\,000$	—	0, 1, 2, 3	—
5	13	$1 \leq m \leq 10^6$	—	0, 1, 2, 3, 4	—
6	11	—	$c_{i,j} = 1, mul_i = 1$	—	—
7	22	—	—	0, 1, 2, 3, 4, 5, 6	Offline-проверка.