

Задача А. НОПроблемо

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Даны две последовательности. Найдите длину их наибольшей общей подпоследовательности (подпоследовательность — это то, что можно получить из данной последовательности вычёркиванием некоторых элементов).

Формат входных данных

В первой строке входного файла записано число N — длина первой последовательности ($1 \leq N \leq 10^3$). Во второй строке записаны члены первой последовательности (через пробел) — целые числа, не превосходящие 10^4 по модулю. В третьей строке записано число M — длина второй последовательности ($1 \leq M \leq 10^3$). В четвёртой строке записаны члены второй последовательности (через пробел) — целые числа, не превосходящие 10^4 по модулю.

Формат выходных данных

В выходной файл требуется вывести единственное целое число: длину наибольшей общей подпоследовательности, или число 0, если такой не существует.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 4 2 1 3 5	2
3 1 2 3 3 1001 1002 1003	0

Задача В. НВП

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Числовая последовательность задана рекуррентной формулой: $a_{i+1} = (k \cdot a_i + b) \bmod m$. Найдите её наибольшую возрастающую подпоследовательность. Если таких последовательностей несколько, можно вывести любую.

Формат входных данных

Программа получает на вход пять целых чисел: длину последовательности n ($1 \leq n \leq 10^5$), начальный элемент последовательности a_1 , параметры k, b, m для вычисления последующих членов последовательности ($1 \leq m \leq 10^4, 0 \leq k < m, 0 \leq b < m, 0 \leq a_1 < m$).

Формат выходных данных

На первой строке выходного файла вы должны вывести количество чисел в найденной вами наибольшей возрастающей подпоследовательности. На следующей строке выведите элементы подпоследовательности, разделяя их пробелами.

Примеры

стандартный ввод	стандартный вывод
5 41 2 1 100	3 41 67 71
7 1 2 1 10	4 1 3 5 7
7 2 2 1 10	3 1 3 5

Задача С. Расстояние Дамерау-Левенштейна

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

Одной из наиболее распространенных опечаток при наборе текста является перестановка двух соседних символов, например, вместо слова «программа» набрано слово «прогармма». Расстояние Левенштейна не учитывает такие опечатки: при вычислении расстояния Левенштейна одна перестановка будет считаться за два редактирования (например, удаление и вставка символа).

При вычислении расстояния Дамерау-Левенштейна, помимо операций замены, вставки и удаления символа допускается еще операция перестановки двух соседних символов. При этом между переставленными символами нельзя вставлять другие символы.

Определите расстояние Дамерау-Левенштейна для двух данных строк.

Формат входных данных

Программа получает на вход две строки, длина каждой из которых не превосходит 1000 символов, строки состоят только из заглавных латинских букв.

Формат выходных данных

Требуется вывести одно число – расстояние Дамерау-Левенштейна для данных строк.

Пример

стандартный ввод	стандартный вывод
XABCDE ACBYDF	4

Задача D. Пилообразная последовательность

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 4 мегабайта

Назовем последовательность пилообразной, если каждый её элемент либо строго больше, либо строго меньше своих соседей. По данным числам n и k определите количество пилообразных последовательностей длины n , составленных из чисел $1, \dots, k$.

Формат входных данных

Программа получает на вход два натуральных числа n и k , $1 \leq n \leq 1000$, $1 \leq k \leq 1000$.

Формат выходных данных

Необходимо вывести остаток от деления количества искомых последовательностей на $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
3 3	10
20 3	35422

Задача Е. Рюкзак максимальной стоимости

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

Дано n предметов массой m_1, \dots, m_n и стоимостью c_1, \dots, c_n соответственно. Ими наполняют рюкзак, который выдерживает вес не более w . Определите набор предметов, который можно унести в рюкзаке, имеющий наибольшую стоимость.

Формат входных данных

В первой строке вводится натуральное число n , не превышающее 100, и натуральное число w , не превышающее 10000.

Во второй строке вводятся n натуральных чисел m_i , не превышающих 100.

В третьей строке вводятся n натуральных чисел c_i , не превышающих 100.

Формат выходных данных

В первой строке выведите число k — количество элементов в рюкзаке наибольшей стоимости. Во второй строке выведите k различных чисел от 1 до n — номера предметов.

Пример

стандартный ввод	стандартный вывод
4 6	3
2 4 1 2	4 3 1
7 2 5 1	

Задача F. Большой рюкзак

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

У вас есть N предметов с весами w_1, w_2, \dots, w_N . Требуется проверить, можно ли выбрать некоторые предметы, суммарный вес которых равен W .

Формат входных данных

В первой строке записаны два целых числа N и W ($1 \leq N \leq 2\,500$, $1 \leq W \leq 6\,250\,000$).

Во второй строке через пробел записаны N целых чисел w_1, w_2, \dots, w_N ($1 \leq w_i \leq 2\,500$) — веса предметов.

Формат выходных данных

В единственной строке выведите «YES» (без кавычек), если, используя данные предметы, можно набрать вес W , либо «NO» в противном случае.

Примеры

стандартный ввод	стандартный вывод
5 10 1 2 3 4 5	YES
2 10 4 5	NO

Задача G. Максимальное подмножество вершин

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

Надо бы всё-таки написать нормальную легенду, а то как-то не очень. И без легенды непонятно, почему задача так называется

Но пока легенды нет, вот формальное условие:

Вам дано дерево на n вершинах. В вершинах записаны числа a_n .

Требуется выбрать подмножество вершин с максимальной суммой a_n , чтобы никакие две соседние вершины не лежали одновременно в этом подмножестве.

Формат входных данных

В первой строке записано число n — количество вершин дерева ($1 \leq n \leq 10^6$). Во второй строке записаны через пробел n чисел v_i ($|v_i| < 10^9$), задающие значения в вершинах. В следующих $n - 1$ строках описаны ребра дерева. В $(i + 2)$ -й строке записаны номера вершин a_i, b_i ($1 \leq a_i, b_i \leq n$), означающие, что в дереве есть ребро из вершины a_i в вершину b_i .

Формат выходных данных

Выведите единственное число — максимальную искомую величину.

Пример

стандартный ввод	стандартный вывод
7 0 7 2 7 1 4 8 3 6 7 6 1 3 5 1 4 6 2 1	25

Задача Н. Максимальная тройка

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

Дано дерево на n вершинах. Требуется выбрать из них три так, чтобы сумма расстояний между ними была максимальна.

Формат входных данных

Первая строка каждого теста содержит натуральное число n — количество вершин в дереве ($3 \leq n \leq 1\,000\,000$). Следующие $n - 1$ строк содержат по 2 натуральных числа v, u и описывают ребро дерева, соединяющее две вершины v и u ($1 \leq v, u \leq n$).

Формат выходных данных

Выведите единственное число — максимальную сумму расстояний.

Система оценки

Решения, правильно работающие на тестах, в которых $n \leq 50$, будут оцениваться в 25 баллов.

Решения, правильно работающие на тестах, в которых $n \leq 500$, будут оцениваться в 50 баллов.

Решения, правильно работающие на тестах, в которых $n \leq 5000$, будут оцениваться в 75 баллов.

Решения, правильно работающие на тестах, в которых $n \leq 10^6$, будут оцениваться в 100 баллов.

Примеры

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

Задача I. Дерево на Манхеттене

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Участникам, использующим язык Python3, рекомендуется отправлять решения на проверку с использованием интерпретатора PyPy3.

Однажды мальчик Вова нашел корневое дерево[†] из n вершин, i -я вершина которого была соединена со своим предком p_i ($p_i < i$) ребром, с записанным на нём числом c_i . Корнем является вершина 1. Увидев его, Вова сразу же придумал интереснейшую задачу, которую вам предстоит решить.

Выкладыванием дерева на прямую назовём сопоставление каждой вершине v целого числа x_v ($1 \leq x_v \leq n$), при котором разным вершинам соответствуют разные числа, а каждое поддерево является последовательным подотрезком.

Проще говоря, выкладывание — это такая перестановка[‡] на вершинах, что для любой вершины v , если отсортировать значения x_u всех вершин u в поддереве вершины v , то образуется отрезок целых чисел $[l, r]$ для каких-то $1 \leq l \leq r \leq n$.

Стоимостью выкладывания назовём величину $\sum_{v=2}^n |x_v - x_{p_v}| \cdot c_v$.

Посчитайте минимальную стоимость выкладывания для данного вам дерева.

[†] *Дерево* — это связный неориентированный граф без циклов. *Корневое дерево* — дерево с выделенной вершиной, которую называют корнем.

[‡] *Перестановкой* длины n является массив, состоящий из n различных целых чисел от 1 до n в произвольном порядке. Например, $[2, 3, 1, 5, 4]$ — перестановка, но $[1, 2, 2]$ не перестановка (2 встречается в массиве дважды) и $[1, 3, 4]$ тоже не перестановка ($n = 3$, но в массиве встречается 4).

Формат входных данных

Первая строка содержит одно целое число n ($2 \leq n \leq 5000$) — количество вершин в дереве.

Затем следуют $n - 1$ строк, i -я из которых содержит два целых числа p_{i+1} и c_{i+1} ($1 \leq p_{i+1} \leq i, 0 \leq c_{i+1} \leq 10^{11}$) — предок $(i + 1)$ -й вершины и число, записанное на ребре из $i + 1$ в p_{i+1} .

Формат выходных данных

В единственной строке выведите одно число — минимальную стоимость выкладывания данного вам дерева.

Примеры

стандартный ввод	стандартный вывод
5 1 6 1 5 2 4 2 3	21
8 1 6 1 9 2 2 2 9 5 4 3 9 6 11	56

Замечание

Можно показать, что добиться на первом тесте из условия стоимости выкладывания меньше, чем 21 нельзя. Данная стоимость получается с помощью выкладывания $x = \{4, 3, 5, 2, 1\}$.

Аналогично, для второго теста стоимости выкладывания меньше, чем 56 получить нельзя. Пример выкладывания с данной стоимостью $x = \{6, 5, 7, 1, 4, 3, 8, 2\}$.

Задача J. Экспедиция

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Планируется отправить экспедицию к соседней звёздной системе. Были отобраны n кандидатов, пронумерованных от 1 до n , среди которых необходимо выбрать участников экспедиции. Организаторы хотят отправить в экспедицию как можно больше кандидатов.

Среди кандидатов был проведён опрос, в процессе которого каждый мог указать не более, чем одного из остальных кандидатов, с которым он не готов отправиться в экспедицию. Результатом опроса для i -го кандидата является целое число a_i , которое равно номеру кандидата, с которым i -й кандидат не готов отправиться в экспедицию, либо -1 , если i -й кандидат готов отправиться в экспедицию в любом составе.

Теперь организаторы должны выбрать, кто из кандидатов отправится в экспедицию. Решено было выбрать участников экспедиции так, что если туда входит некоторый кандидат i , и $a_i \neq -1$, то туда не входит кандидат a_i . Организаторы хотят выбрать максимальное количество участников экспедиции.

Требуется написать программу, которая по заданным результатам опроса кандидатов определяет максимальное количество кандидатов, которых можно отправить в экспедицию.

Формат входных данных

В первой строке входных данных находится целое число n — количество кандидатов ($1 \leq n \leq 300\,000$).

В следующих n строках даны результаты опроса, i -я из этих строк содержит результат опроса i -го кандидата, целое число a_i ($a_i = -1$ или $1 \leq a_i \leq n$, $a_i \neq i$).

Формат выходных данных

В единственной строке выведите одно целое число — максимальное количество кандидатов, которых можно отправить в экспедицию.

Примеры

стандартный ввод	стандартный вывод
4 2 4 2 1	2
3 2 -1 2	2

Задача К. Польшар и Подарки

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

Рождество! Польшар и его друзья будут дарить друг другу подарки. Всего шаров n . Каждый шар должен подарить подарок ровно одному другому шару в соответствии с некоторой перестановкой p , $p_i \neq i$ для всех i .

К сожалению, шары забывчивы. Мы знаем, что ровно k шаров забудут принести свои подарки. Шар номер i получит подарок, если будут выполнены следующие два условия:

1. Шар номер i должен принести свой подарок.
2. Шар x такой, что $p_x = i$, должен принести свой подарок.

Какое минимально и максимально возможное число шаров, которые **не** получают свой подарок, если ровно k шаров забудут принести свой подарок?

Формат входных данных

В первой строке находится два целых числа n и k ($2 \leq n \leq 10^6$, $0 \leq k \leq n$) — общее число шаров и число шаров, которые забудут подарки.

Во второй строке находится перестановка p целых чисел от 1 до n , где p_i — номер шара, которому должен дать подарок шар номер i . Для всех i выполняется $p_i \neq i$.

Формат выходных данных

Выведите два числа — минимально и максимально возможное число шаров, которые **не** получают подарков, соответственно.

Примеры

стандартный ввод	стандартный вывод
5 2 3 4 1 5 2	2 4
10 1 2 3 4 5 6 7 8 9 10 1	2 2

Замечание

В первом примере, если первый и третий шары забудут принести подарок, то они же и будут единственными, кто не получит подарка. Поэтому минимальный ответ равен 2. Однако, если первый и второй шары забудут, то только пятый шар получит подарок. Поэтому максимальный ответ равен 4.