

Задача А. Тамада

Имя входного файла:	sadgegrdzelo.in
Имя выходного файла:	sadgegrdzelo.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайт

По мнению тех, кто знал видного советского математика Илью Несторовича Векуа, он был интересным собеседником и хорошим тамадой.

Одним из показателей, по которым можно оценивать качество руководства застольем, является коэффициент тостирования. Коэффициент тостирования застолья равен минимуму индивидуальных коэффициентов участников данного застолья. Индивидуальный коэффициент - это количество тостов, объектами которых оказался данный участник на протяжении всего периода данного застолья. Например, если застолье посвящено именинам, то участник, являющийся другом именинника, будет одним из объектов тоста за друзей именинника, но если он еще и одноклассник именинника, то будет объектом тоста за одноклассников именинника, разумеется, если таковой будет провозглашён. Говорят, что в свое время существовала Всесоюзная Федерация Проведения Застольий, располагавшаяся в Сигнахи (Грузия, Кахетия). И говорят, что для получения звания Тамады Всесоюзной Категории необходимо было провести 3 квалификационных застолья со значением коэффициента тостирования не меньшим, чем 2. Та же федерация, оказывается, присуждала звание Тамады Республиканской Категории за проведение 5 квалификационных застолий с коэффициентом тостирования, не меньшим, чем 3. О нормативах для получения звания Тамады Кахетинской Категории молва умалчивает. По данным о тостах, провозглашённых во время заданного застолья, Вам необходимо определить, получил ли тамада этого застолья балл для присуждения ему почётного звания Тамады и если да, то какой категории соответствует этот балл.

Формат входного файла

В первой строке входного файла задаются два целых числа $1 \leq n \leq 25$ и $2 \leq m \leq 10^8$ — соответственно количество тостов, провозглашённых тамадой на данном застолье и общее количество участников застолья. Затем расположены n строк. Если первым символом i -й строки является +, то далее расположены номера тех участников застолья, которые подпадают под i -ый тост, а если первым символом строки является -, то далее расположены номера тех участников застолья, которые не подпадают под i -ый тост. Знак отделяется от первого номера участника пробелом. Участники перенумерованы натуральными числами от 1 до m , при этом если сразу после знака вместо номера участника идёт 0, то это значит, что, в зависимости от знака, в тосте или не упоминался ни один участник застолья (в случае + 0), или упоминались все его участники (в случае - 0). При этом длина каждой строки входного файла не превышает 1000 символов.

Формат выходного файла

В выходной файл выведите текст NO, если тамада не смог заслужить балл для почётного звания, TUK, если тамада заслужил балл для присвоения звания Тамады Всесоюзной Категории и TRK, если тамада заслужил как балл для присвоения звания Тамады Всесоюзной Категории, так и балл для присвоения звания Тамады Республиканской Категории.

Пример

sadgegrdzelo.in	sadgegrdzelo.out
1 3 + 2 3	NO
2 3 + 2 3 - 3	NO
2 3 + 1 2 3 - 0	TUK
5 4 + 2 3 + 2 4 1 - 0 - 1 + 1	TRK

Задача В. Футбол

Имя входного файла: `soc.in`
Имя выходного файла: `soc.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

Футбол в Грузии всегда был очень популярен. Не зря во времена СССР, когда тбилиское «Динамо» стало обладателем Кубка Кубков УЕФА, грузинских футболистов называли «советскими бразильцами». И дело не только в высокой технике грузинских футболистов — как и в Бразилии, в Грузии в футбол на любительском уровне играли практически все. Проходила масса любительских турниров, при этом одна и та же команда могла заявляться и играть сразу в нескольких турнирах.

И вот в одном грузинском городе решили составить рейтинг городских команд. Для этого по окончании сезона были собраны итоговые таблицы всех турниров, проводимых в городе, и введены следующие правила оценки: если команда заняла в таблице с l участниками m -е место, она набирает $l + 1 - m$ локальных очков; если команда не участвовала в данном турнире и в таблице отсутствует, она набирает 0 локальных очков. Расстоянием между двумя таблицами называется сумма абсолютных величин разностей локальных очков, набранных каждой из команд города в этих таблицах. Общий рейтинг-лист строится как таблица с наименьшим суммарным расстоянием от итоговых таблиц всех турниров.

Ваша задача — по заданным итоговым таблицам игр всех турниров, проведённых в городе, построить общий рейтинг-лист и вывести его суммарное расстояние от итоговых таблиц этих турниров.

Формат входного файла

В первой строке заданы два целых числа n и k ($2 \leq n \leq 500, 1 \leq k \leq 500$), обозначающие количество команд в городе и количество сыгранных турниров. Далее в k строках идут итоговые таблицы турниров. Таблица с номером i начинается с целого числа m_i ($2 \leq m_i \leq n$) — количества команд в i -м турнире, далее идёт список из m_i команд в порядке занятых в турнире мест, начиная с первого (каждая команда обозначается целым числом от 1 до n). При этом ситуация, когда в какой-то таблице две команды «делят» место, является невозможной.

Формат выходного файла

В выходной файл выведите одно целое число p — суммарное расстояние итогового рейтинг-листа, построенного по указанным правилам, от всех заданных во входном файле итоговых таблиц турниров.

Пример

<code>soc.in</code>	<code>soc.out</code>
4 2 3 1 2 3 2 4 3	11

Задача С. Гости с Тау Кита

Имя входного файла: `tau1.in`
Имя выходного файла: `tau1.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

Организаторы Кубка Векуа были изрядно удивлены, когда среди заявок на участие в соревнованиях оказалась заявка команды с Тау Кита. Однако заявка была оформлена по всем правилам, и таукитяне были включены в списки. В Батуми они собирались прибыть на своём космическом корабле. Для коммуникации с приближающимся кораблём таукитяне предложили установить стационарный лазер и использовать некий аналог азбуки Морзе.

Однако в день прибытия возникла неожиданная сложность: на небе появились плотные облака, которые, проходя над установкой, закрывали луч лазера. Для того, чтобы сделать поправку на вызванные облаками ошибки, гости подали срочный запрос в жюри соревнований: какое наибольшее число раз луч лазера будет закрыт? К сожалению, жюри не знает место установки лазера — это дело оргкомитета, да и из прогноза погоды известно только, что ветер будет дуть с постоянной скоростью. Но положение облаков на небе жюри определить может. Поэтому было принято решение сообщить наибольшее для всех возможных расположений лазера и направлений ветра количество закрытий луча. Вам поручено написать программу, вычисляющую это количество.

Для упрощения задачи поверхность земли считается плоской, проекция каждого облака на землю представляется в виде многоугольника с целыми вершинами, а ветер с постоянной скоростью смещает все имеющиеся облака вдоль некоторого вектора параллельно поверхности земли. Лазер представляется точкой на плоскости. Считается, что облако закрывает лазер, если точка, его представляющая, лежит на границе или внутри проекции облака.

Формат входного файла

В первой строке входного файла записано число n — количество облаков на небе. Далее идёт n строк, описывающих проекции отдельных облаков: в i -й строке сначала идёт количество $n_i > 3$ вершин в многоугольнике, представляющем i -е облако, затем $2n_i$ координат этих вершин $-10^9 \leq x_j, y_j \leq 10^9$. При этом суммарное количество вершин многоугольников для всех облаков, заданных во входном файле не превышает 2000.

Формат выходного файла

Одно число k - максимальное количество перебоев в видимости лазера, вызванных облаками.

В случае, показанном в примере, максимальный ответ 3 достигается, например, при размещении лазера в точке $(0,4)$ и ветре, направленном вдоль вектора $[-1,0]$;

Пример

<code>tau1.in</code>	<code>tau1.out</code>
2 5 1 1 5 1 5 4 3 2 1 4 3 2 4 3 5 4 4	3

Задача D. Общение с организаторами

Имя входного файла: tau2.in
Имя выходного файла: tau2.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

Получив от жюри информацию о возможных помехах при передаче сигнала с помощью лазера, экипаж тау-китянского корабля должен учесть эти помехи при чтении полученных сообщений. В тау-китянском языке отсутствует понятие «начало слова» и «конец слова», то есть любые два слова, переводимые друг в друга последовательностью из некоторого числа циклических сдвигов, эквивалентны. Циклическим сдвигом называется преобразование слова, заключающееся в том, что первый символ слова переставляется в его конец.

Помехи, создаваемые облаками, по расчётам бортового связиста, могут привести к тому, что ровно одна буква в каждом слове будет принята неправильно. Так что для решения задачи коррекции ошибок экипажу корабля важно знать, может ли принятое слово после замены ровно одного символа на другой, с ним не совпадающий, оказаться эквивалентным словарному слову той же длины.

Напишите программу, выполняющую такую проверку.

Формат входного файла

В первой строке входного файла задана длина $1 \leq n \leq 1000000$ принятого слова. Во второй строке входного файла находится принятое слово, состоящее из n строчных латинских букв. В третьей строке находится словарное слово, также состоящее из n строчных латинских букв

Формат выходного файла

В выходной файл выведите слово NO, если принятое слово или уже является эквивалентным в смысле тау-китянского языка словарному, или не может оказаться эквивалентным словарному при замене ровно одной буквы. В противном случае в первую строку выходного файла выведите YES, а во вторую — упорядоченный по возрастанию список всех порядковых номеров циклических сдвигов принятого слова, после которых строка, содержащая принятое слово отличается от строки, содержащей словарное ровно в одной позиции (определим порядковый номер циклического сдвига, состоящего из n единичных циклических сдвигов равным n . Порядковый номер «пустого» сдвига равен 0).

Пример

tau2.in	tau2.out
5 ababa aaaab	YES 2 4
5 ababa bbbbbb	NO

Задача Е. Церемония открытия

Имя входного файла: `open.in`
Имя выходного файла: `open.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

Режиссёр церемонии открытия Кубка Векуа предложил следующий вариант представления приехавших на Кубок n команд разных вузов: на сцену выходит цепочка из $2n$ человек. Каждый вуз символизируют два участника цепочки: первый несёт табличку с названием вуза на языке той страны, которую вуз представляет, а второй — по замыслу режиссёра это должен быть ребёнок — несёт воздушный шарик (по некоторой аналогии с ACM ICPC). Всего в цепочке n человек с табличками и n детей с шариками.

Участники церемонии уже выстроились в ряд, как вдруг оказалось, что последовательность нарушена: где-то рядом стояли два участника с табличками, где-то дети стояли по двое и по трое... Так как участников много, то попытка отдавать команды всем сразу приведёт к хаосу, и режиссёр может только попросить двух человек поменяться между собой. Так как порядок представления вузов в цепочке особого значения не имеет (в конце концов, это открытие, а не оглашение итогов), то режиссёру достаточно добиться ситуации, в которой первым в цепочке будет участник с табличкой, и при этом нигде в цепочке не будут стоять рядом два участника с табличками или два ребёнка с воздушными шариками. Режиссёр хочет узнать, за какое минимальное количество перестановок он может добиться такой ситуации.

Формат входного файла

В первой строке входного файла задано натуральное число $n \leq 10^6$ — количество прибывших команд. В последующих $2n$ строках идут целые числа, описывающие цепочку, начиная с её «головы»: 0, если на данной позиции в цепочке стоит ребёнок с воздушным шариком, и $0 < k \leq 10^9$ (условный код языка, на котором написана табличка) в случае, если на данной позиции в цепочке стоит человек с табличкой, содержащей название команды.

Формат выходного файла

Выведите одно целое число — минимальное количество перестановок, которое потребуется сделать режиссёру, чтобы добиться необходимого чередования.

Пример

<code>open.in</code>	<code>open.out</code>
5	3
5	
1	
0	
0	
2	
4	
0	
3	
0	
0	

Задача F. Множества

Имя входного файла: `set.in`
Имя выходного файла: `set.out`
Ограничение по времени: 30 секунд
Ограничение по памяти: 64 мегабайт

Для распределения участников Кубка Векуа по аудиториям разрабатывается система многопараметрической жеребьёвки. Вам досталась задача написать для системы отдельный модуль, используемый для определения параметров жеребьёвки.

Для заданного натурального $x > 1$ построим все множества, состоящие из k различных натуральных чисел, меньших или равных n , такие, что для любого натурального числа y как минимум одно из чисел y и $x * y$ не принадлежит данному множеству. Ваша задача — вычислить остаток от деления общего количества таких множества на заданное натуральное число m .

Формат входного файла

Во входном файле записаны четыре целых числа n , m , k , x ($1 \leq n \leq 10^{18}$, $2 \leq m \leq 10^6$, $0 \leq k \leq 1000$, $2 \leq x \leq 10$).

Формат выходного файла

В выходной файл выведите остаток от деления общего количества описанных в условии множеств на m .

Пример

<code>set.in</code>	<code>set.out</code>
6 1234 3 2	9

Задача G. Дождь

Имя входного файла: `rain.in`
Имя выходного файла: `rain.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

В Батуми не всегда бывает солнечно: море есть море, и иногда на город неожиданно может налететь настоящий шторм. Резо, один из организаторов Кубка Векуа, ждал на автостанции прибытия гостей на междугороднем автобусе, когда пошёл сильный ливень. На улице стоять нельзя, а в здании автостанции затеяли ремонт. И тут Резо обратил внимание на полное расписание городского автобуса по всем остановкам, висевшее на стене автостанции: если сесть на станции в автобус этого маршрута, проехать на нём несколько остановок, потом выйти и сесть на встречный автобус того же маршрута так, чтобы успеть на автостанцию к прибытию гостей. При этом Резо хочет провести под дождём как можно меньше времени. Ваша задача - посчитать это время.

Формат входного файла

В первой строке входного файла заданы 5 чисел t_1, t_2, m, n_1, n_2 ($0 \leq t_1 \leq t_2 \leq 10^9$, $2 \leq m \leq 1000$, $n_1, n_2 \geq 1$, $m * (n_1 + n_2) \leq 10^6$). t_1 — время начала ливня, t_2 — время прибытия гостей на автостанцию, m — количество остановок на маршруте внутригородского автобуса (включающее автостанцию), n_1 и n_2 — количество рейсов внутригородского автобуса, идущих с автостанции и до автостанции соответственно. Конечными пунктами маршрута являются автостанция и остановка с номером m , оставаться в автобусе на конечном пункте нельзя. Далее в m строках записаны расписания автобусов по каждой из остановок, перечисленных последовательно в направлении от автостанции к конечной остановке — $n_1 + n_2$ неотрицательных целых чисел, не превосходящих 10^9 . Первые n_1 чисел обозначают время отправления автобусов, идущих от автостанции: i -е число обозначает время прибытия и отправления i -го «прямого» рейса с данной остановки, следующие n_2 — время прибытия и отправления автобусов, идущих к автостанции: $n_1 + i$ -е число обозначает время отправления i -го «обратного» рейса с данной остановки. Считается, что автобус прибывает и отправляется с остановки в ту же единицу времени, в частности, Резо может пересесть с автобуса на автобус, только если время прибытия первого автобуса на остановку не превосходит времени прибытия второго автобуса.

Формат выходного файла

Выведите целое число — минимальное время, которое Резо вынужден будет провести под дождём до прибытия команды.

Пример

<code>rain.in</code>	<code>rain.out</code>
0 10 3 1 2 0 9 10 3 4 8 4 3 7	2

Задача N. Кредитная карта

Имя входного файла: `card.in`
Имя выходного файла: `card.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

Победителю индивидуального Кубка Векуа от спонсоров достался крупный денежный приз. Для перечисления этой суммы участник должен назвать номер своей кредитной карты. Проблема в том, что он забыл этот номер, но помнит одно из его свойств: в n -ричной системе это натуральное число состоит из k попарно различных цифр, при этом если число умножить на 2, то оно будет циклической перестановкой первоначального, то же самое будет, если его умножить на 3... и так далее до k . Ваша задача - вычислить, сколько таких чисел имеется для заданного n .

Формат входного файла

Во входном файле задано единственное число $3 \leq n \leq 1000$ — основание системы счисления.

Формат выходного файла

Выведите целое число — количество в n -ричной системе чисел с указанными свойствами.

Пример

<code>card.in</code>	<code>card.out</code>
3	0
10	1

Задача I. Анализ логов

Имя входного файла: `logs.in`
Имя выходного файла: `logs.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайт

После проведения очередного Чемпионата Южного Кавказа тренеры команды Тбилисского университета решили проанализировать, сколько попыток по каким задачам на дорешивании Чемпионата делали команды. У тренеров имеется лог, в котором указана последовательность посылок по каждой задаче. Всего в дорешивании предлагалось 26 различных задач, каждая задача обозначается в логе одной латинской буквой. Тренеры хотят составить программу, которая бы подсчитывала количество разных групп команд по «набору попыток» и количество команд в каждой из таких групп. Команды относятся к одной группе, если по каждой задаче количество сделанных ими попыток совпадает.

Формат входного файла

В первой строке входного записано одно число $1 \leq N \leq 10000$. Далее до конца файла по одному слову в строке записаны N слов, состоящих из строчных латинских букв, описывающих последовательность посылок по задачам. Одинаковые буквы обозначают посылки по одной и той же задаче, разные — по разным. Известно, что ни одна команда не сделала более 15 посылок.

Формат выходного файла

В выходной файл нужно вывести несколько строк, по два числа в каждой: количество команд в группе и количество групп такого размера. Строки необходимо расположить в порядке убывания размера групп.

Пример

<code>logs.in</code>	<code>logs.out</code>
5	2 1
abcef	1 3
acebf	
a	
b	
ac	