

Задача А. Водохранилище Пумбы

Имя входного файла: `water.in`
Имя выходного файла: `water.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайта

Пумба очень запасливый. В связи с возможной засухой, он решил запастись водой. У Пумбы есть n деревянных пронумерованных тазиков. Объем кадого тазика равен m литров. Запасаться водой Пумба будет на протяжении k дней. В каждый день заготовки Пумба может сделать ровно одно из следующих действий:

- долить c_i литров воды во все тазики с номерами от l_i до r_i включительно. При этом если в тазик не поместится c_i литров, Пумба его полностью заполнит водой.
- спросить у Тимона суммарное количество литров воды в тазиках с номерами от l_i до r_i включительно.

Тимон очень ленивый, поэтому он поручил вам отвечать на вопросы Пумбы. Перед заготовкой в i -м тазике уже содержалось a_i литров воды.

Формат входного файла

В первой строке находятся три целых числа n , m , k — количество тазиков, объем одного тазика и количество дней, на протяжении которых Пумба будет запасаться водой ($1 \leq n \leq 200000$, $1 \leq m \leq 10^9$, $1 \leq k \leq 200000$).

Во второй строке содержатся n чисел a_i — объем воды в каждом тазике перед заготовкой ($0 \leq a_i \leq m$).

Каждая из k следующих строк содержит несколько целых чисел, описывающих действие Пумбы в соответствующий день.

Если в i -й день Пумба доликает воду, первое число i -й строки будет равно одному. За ним будут идти три целых числа l_i , r_i , c_i — границы отрезка номеров тазиков, куда Пумба будет доликать воду и объем доликаемой воды ($1 \leq l_i \leq r_i \leq n$, $1 \leq c_i \leq 10^9$).

Если же в i -й день Пумба спросит суммарный объем воды в тазиках с номерами от l_i до r_i включительно, первое число i -й строки будет равно двум. За ним будут следовать два целых числа l_i и r_i соответственно ($1 \leq l_i \leq r_i \leq n$).

Формат выходного файла

На каждый вопрос Пумбы выведите ответ в отдельной строке. Ответы выведите в том порядке, в котором следуют запросы.

Примеры

water.in	water.out
3 3 7	6
1 2 1	7
1 1 2 1	2
2 1 3	3
1 2 3 1	2
2 1 3	
2 1 1	
2 2 2	
2 3 3	

Задача В. Математика

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

К сожалению, даже в дикой природе без математики никуда. Чтобы быть хорошим королем саванны, нужно быть не только сильным и ловким, но и уметь правильно распределить ресурсы, а тут никак не обойтись без сложения, умножения и деления. Это знал и Муфаса, и поэтому в один прекрасный день он заставил Симбу читать книгу по математике. Книга была очень сложной, а сложнее всего Симбе было со знаком \wedge .

Дело в том, что в этой книге таким знаком обозначено две совсем разные вещи. Во-первых, таким знаком обозначили возведение в степень. Во-вторых — сноски. Сноски — это ссылки на объяснения некоторых кусков текста, которые приведены в конце книги. В этой книге после каждого символа \wedge следует некоторое число, после которого может быть либо знак препинания, либо пробел. Симба знает, что числа, которые стоят в сносках после символа \wedge , являются последовательным натуральными числами. Число возле самой первой сноски равно единице. Про каждый символ \wedge можно сказать, является ли он возведением в степень или сноской.

Рассмотрим все возможные варианты разбиения знаков \wedge на степени и сноски. Будем считать, что авторы учебника постарались объяснить все как можно более подробно, а поэтому, корректными будем считать только те варианты, в которых количество сносок максимально.

Про каждый символ \wedge Симба хочет знать, может ли он являться возведением в степень в каком-нибудь из корректных разбиений. Аналогично, он хочет знать, может ли он являться сноской в каком-нибудь из корректных разбиений. Помогите ему!

Формат входного файла

В первой и единственной строке задан текст книги, который содержит не более 10^6 символов. Текст может состоять из строчных и заглавных латинских букв, цифр, символов \wedge , а также точек, запятых, вопросительных знаков, пробелов и двоеточий. Гарантируется, что в тексте после каждого символа \wedge следует натуральное число (возможно, довольно большое). После каждого такого числа следует либо пробел, либо знак препинания (точка, запятая, двоеточие или вопросительный знак).

Формат выходного файла

В первой строке выведите количество символов \wedge в тексте. Во второй строке для каждого символа \wedge выведите Y , если он может являться степенью, и N в противном случае. Аналогично в третьей строке для каждого символа \wedge выведите Y , если он может являться сноской и N в противном случае.

Примеры

<code>math.in</code>	<code>math.out</code>
<code>sample: 2 plus 2 equals 2\wedge2.\wedge1 easy, \wedge2 isnt it? 1\wedge1 equals 1.</code>	<code>4 YNNY NYYN</code>

Комментарий

Обратите внимание на то, что символы, стоящие перед \wedge , никак не влияют на то, может ли этот символ являться сноской или возведением в степень.

Задача С. Родственные связи

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

Еще недавно королевство было совсем маленьким, все всех знали, и определение родственных связей не представляло никакой проблемы. Так сложилось, что в последние годы население резко выросло, и стало сложно определить кто кому кем приходится.

Мало кто знает, но у каждого зверя есть паспорт, в котором указан его номер — целое неотрицательное число. Паспортная система королевства хороша, но не идеальна, поэтому эти номера совсем не обязательно уникальны. Хороша она, собственно, тем, что если в десятичной записи номеров паспортов двух зверей есть хотя бы одна общая цифра, то они являются родственниками. Например, звери с номерами 47 и 107 — родственники, а с номерами 74 и 931 — нет.

Король Лев, как настоящий правитель, хочет знать все о своих подданных, поэтому просит вас посчитать количество различных пар зверей, которые являются родственниками.

Формат входного файла

Первая строка входного файла содержит число N ($1 \leq N \leq 500000$) - количество зверей в королевстве.

Вторая строка содержит n целых неотрицательных чисел, не превышающих 10^9 , разделенных пробелами — номера зверей.

Формат выходного файла

Выведите единственное число — ответ на задачу.

Примеры

<code>relatives.in</code>	<code>relatives.out</code>
5 10 74 47 77 301	4

Комментарий

Родственные связи есть у зверей 10 и 301, 74 и 47, 74 и 77, 47 и 77. Всего - 4 пары.

Задача D. Полупалиндромы

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

Тимон и Пумба придумали себе очередное развлечение! В этот раз они надумали коллекционировать строки особого вида. Однако все уже придуманные другими строки им наскучили, потому они придумали полупалиндромы!

Полупалиндромом называется строка $X[1..n]$, удовлетворяющая следующим условиям:

- Любая строка длины не более 1 — полупалиндром.
- Строка $X[1.. \lfloor \frac{n}{2} \rfloor]$ является полупалиндромом.
- Строка $X[n - \lfloor \frac{n}{2} \rfloor .. n]$ является полупалиндромом.
- Обозначим $X_1 = X[1.. \lfloor \frac{n}{2} \rfloor]$, $X_2 = X^R[1.. \lfloor \frac{n}{2} \rfloor]$. Будем говорить, что символ $X_1[i]$ соответствует символу $X_2[i]$. Тогда одинаковым символам из X_1 должны соответствовать одинаковые символы из X_2 , и наоборот.

Строка $X[i..j]$ обозначает подстроку строки X с индексами от i до j включительно. Строка X^R обозначает развернутую строку X . $\lceil a \rceil$ означает число a , округленное вверх, $\lfloor a \rfloor$ — число a , округленное вниз.

У Пумбы есть длинная строка S . Тимон хочет найти самую длинную подстроку S так, чтобы она была полупалиндромом. Помогите ему!

Формат входного файла

В первой строке находится целое число n — длина строки ($1 \leq n \leq 3000$). Во второй строке содержится строка S , состоящая из n строчных букв английского алфавита.

Формат выходного файла

В первой строке выходного файла выведите подстроку T строки S максимальной длины, являющейся полупалиндромом. Если таких подстрок несколько, то выведите любую.

Примеры

<code>semipal.in</code>	<code>semipal.out</code>
7 aaacaba	acaba
9 aabbaccaa	aabbaccaa

Задача Е. Пришло время праздника

Имя входного файла: `rgb.in`
Имя выходного файла: `rgb.out`
Ограничение по времени: 6 секунд
Ограничение по памяти: 256 мегабайта

Вот и пришло время. На землях прайда пройдет первый в истории карнавал. Праздник пройдет рядом со Скалой Прайда. Все должно быть по высшему разряду, поэтому жители решили украсить часть скалы в честь праздника. Так получилось, что единственными кандидатами на эту работу остались Тимон и Пумба. Им и поручили разрисовать часть Скалы Прайда.

Они еще не придумали, что они будут рисовать, но уже поняли, что картина должна быть необычайно красивой. Поэтому они решили, для начала, заказать специальные краски. Краски имеют цвет, он задается тремя числами r, g и b ($0 \leq r, g, b < 256$). Тимон и Пумба — очень странные ребята. Они считают картину, в которой много очень разных цветов, некрасивой. Поэтому, чем меньше $\max_{i,j} \max\{|r_i - r_j|, |g_i - g_j|, |b_i - b_j|\}$ (где r_i, g_i, b_i — цвета, характеризующие краску, которая была использована во время разрисовки), тем более, на их взгляд, картина красива.

Тимон и Пумба вынуждены экономить, поэтому они посчитали, что им нужно ровно k различных красок, чтобы раскрасить скалу.

Нужно выбрать такие k цветов, чтобы нарисованная картина была как можно красивее.

Формат входного файла

В первой строке заданы два натуральных числа n и k ($1 \leq k, n \leq 10^5, k \leq n$).

В следующих n строках заданы цвета красок в тюбиках, по три числа в строке r_i, g_i и b_i ($0 \leq r_i, g_i, b_i < 256$).

Формат выходного файла

В единственной строке через пробел выведите k различных чисел — номера тюбиков, которые требуется выбрать, чтобы нарисовать как можно более красивую картину.

Тюбики нумеруются в том порядке, в котором они заданы во входном файле.

Если есть несколько ответов на задачу, выведите любой.

Примеры

<code>rgb.in</code>	<code>rgb.out</code>
2 2 255 255 255 0 0 0	1 2
4 2 255 255 0 255 0 255 0 255 255 255 255 255	1 4
3 2 255 255 255 255 255 255 255 255 255	1 3

Задача F. Испытание Рафики

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Не я запутался, а ты запутался.

Рафики

Это интерактивная задача.

Каждый львенок должен пройти ряд испытаний, чтобы стать Королем Львом. Наставник Симбы, Рафики, известен своим нежеланием четко формулировать задачи, поэтому вам придется самим задавать наводящие вопросы. А задача заключается в том, что бы определить базу и модуль полиномиального хэша.

Рафики не хочет давать Симбе никаких начальных данных. При этом, он готов несколько раз сказать Симбе, каков будет хэш строки, названной Симбой.

Полиномиальный хэш Рафики считает по формуле $(s_0 \cdot x^0 + s_1 \cdot x^1 + s_2 \cdot x^2 + \dots + s_n \cdot x^n) \bmod p$, где s_i — номер в английском алфавите i -й буквы строки, если символы в строке нумеруются с нуля, x и p — натуральные числа называемое базой и модулем хэша соответственно, \bmod — остаток от деления.

К сожалению, Симба прогулял занятие Зазу, на котором тот рассказывал хэши, и теперь он попросил вас помочь ему.

Протокол взаимодействия с программой жюри:

Программа жюри выводит слово `start` в отдельной строке. После этого не более чем 10 раз повторяются следующие действия.

Ваша программа выводит в отдельной строке строку, для которой вы хотите посчитать хэш. Все буквы должны быть английскими строчными, длина строки не должна превышать 1000 символов. После этого программа жюри выводит одно натуральное число — полиномиальный хэш этой строки, посчитанный с базой x ($2 \leq x \leq 100$) и модулем p ($10^9 \leq p \leq 2 \cdot 10^9$).

Когда вы найдете базу и модуль хэша, вашей программе необходимо вывести

`answer: x p`

где x и p — база и модуль хэша соответственно.

Примеры

stdin	stdout
<code>start</code>	<code>helloworld</code>
<code>392578150</code>	<code>ihuckyou</code>
<code>803444726</code>	<code>aaaaa</code>
<code>954305</code>	<code>panic</code>
<code>3052183</code>	<code>stupid</code>
<code>123325769</code>	<code>ahahahahahahahahahahahaha</code>
<code>410304507</code>	<code>qwerty</code>
<code>734740968</code>	<code>wasd</code>
<code>137477</code>	<code>answer: 31 1000000007</code>

Комментарий

Для корректной работы программы после каждой операции вывода данных вам необходимо делать следующие операции:

- В языке Pascal: `flush(output);`
- В C/C++: `fflush(stdout);`
- В Java: `System.out.flush();`
- В Python: `sys.stdout.flush();`

Кроме этого, не забывайте после каждой выведенной строки ставить перевод строки.

Задача G. Бутерброды из жуков

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

Сколько... Но питательно!

Король лев

Тимон и Пумба очень любят есть жуков, особенно бутерброды из них. А их лучший друг Симба относится к этому лакомству равнодушно. Поэтому Тимон и Пумба хотят доказать Симбе, что вкуснее бутербродов из жуков ничего нет. В дупле одного из деревьев они нашли n жуков и k кусков хлеба. Теперь они хотят сделать несколько бутербродов для Симбы.

Бутерброд делается следующим образом: кладется один кусок хлеба, сверху на него кладется один жук, на жука кладется еще один кусок хлеба и т.д. в итоге получится конструкция, в которой снизу лежит один кусок хлеба, дальше жуки и куски хлеба чередуются, причем наверху всей конструкции может лежать как жук, так и кусок хлеба. Тимон и Пумба считают, что если Симба съест бутерброд, в котором будет t жуков, то его удовлетворение увеличится на a_t . Они хотят, чтобы Симба получил от их бутербродов как можно большее удовлетворение, причем задействовать нужно всех жуков и все куски хлеба. Помогите им в этом нелегком деле!

Формат входного файла

В первой строке дано два целых числа n и k , где n — количество жуков, k — количество кусков хлеба ($1 \leq n \leq 500, 1 \leq k \leq 10^9$). Во второй строке содержится n чисел a_i ($1 \leq a_i \leq 10^7$).

Формат выходного файла

В единственной строке выходного файла выведите максимальное удовлетворение, которое Симба может получить, съев бутерброды. Если собрать бутерброды, использовав при этом всех жуков и хлеб, невозможно, выведите «Impossible».

Пример

sandwiches.in	sandwiches.out
3 3 1 3 2	4
3 2 1 2 3	Impossible

Задача Н. Побег

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайта

Это интерактивная задача.

Шрам хочет остаться единственным претендентом на титул короля Саванны. Ради этого он коварно расправился с Муфасой, после чего приказал своим приспешникам-гиенам уничтожить Симбу. К счастью, Симбе улыбнулась удача, и ему предоставился шанс спастись от стаи разъяренных гиен.

Гиены преследуют Симбу на квадратном клетчатом поле, длина и ширина которого равны k клеткам. Как столбцы, так и строки этого поля пронумерованы числами от одного до k . Симба находится в одной из клеток поля, находящейся в первой строке. Стая гиен находится в некоторой клетке поля, не совпадающей с той, в которой находится Симба.

Симба и стая по очереди перемещаются по полю. За один ход Симба должен перебежать в одну из клеток, соседних по вертикали, горизонтали, или диагонали с той, в которой он находится. После этого гиены могут переместиться в любую клетку, расстояние до которой по горизонтали совпадает с расстоянием по вертикали. Проще говоря, гиены могут переместиться в любую клетку, находящуюся на одной диагонали с той, в которой они располагаются в данный момент. Заметим, что из этого определения следует, что они могут вообще не перемещаться и остаться в той же клетке.

Если после чьего-либо хода гиены оказываются в одной клетке с Симбой, будущий король погибает в неравной схватке. Если же после очередного хода львенка он оказывается в любой из клеток k -й строки, в которой нет гиен, он благополучно пересекает поле и скрывается в джунглях. Помогите Симбе сбежать от своих преследователей.

Протокол взаимодействия с программой жюри:

В первой строке программа жюри передает вашей программе одно целое число k ($3 \leq k \leq 100$) — размер поля, которое необходимо пересечь Симбе. В следующей строке программа жюри передает одно целое число b ($1 \leq b \leq k$) — номер столбца, в котором находится клетка с Симбой. В следующей строке программа жюри передает два целых числа x и y ($1 \leq x, y \leq k$) — номера строки и столбца клетки, в которой находится стая гиен. Гарантируется, что эта клетка не является клеткой с Симбой. После этого несколько раз повторяются следующие действия.

Ваша программа передает два целых положительных числа, не превышающих k — номера строки и столбца клетки, в которую необходимо переместиться Симбе. После этого программа жюри передает вашей программе два целых положительных числа, не превышающих k — номера строки и столбца клетки, в которую перемещается стая гиен.

В случае, если после очередного хода Симбы он оказался в клетке строки с номером k , в которой нет гиен, завершите программу — это будет означать успешный побег Симбы. За все время работы программы Симба должен сделать не более 1000 ходов.

Примеры

<code>stdin</code>	<code>stdout</code>
4	2 1
1	3 2
4 4	4 2
2 2	
3 1	

Комментарий

Для корректной работы программы после каждой операции вывода данных вам необходимо делать следующие операции:

- В языке Pascal: `flush(output);`
- В C/C++: `fflush(stdout);`
- В Java: `System.out.flush();`
- В Python: `sys.stdout.flush();`

Кроме этого, не забывайте после каждой выведенной строки ставить перевод строки.

Задача I. Шрам готовит атаку

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

Шрам готовит армию гиен для нападения. На отбор пришло n гиен. Шрам желает выбрать самых способных из них: тех, кому природа подарила прирожденный талант.

У каждой гиены есть свой генотип. Генотип представляет из себя последовательность генов. Гены обозначены строчными латинскими буквами.

Некоторая непустая подпоследовательность генов генотипа каждой гиены являлась ее генотипом при рождении. После этого, за время жизни гиены в различные места генотипа могло добавиться произвольное количество произвольных генов, после чего он пришел к нынешнему своему виду.

Скажем, что генотип a мощнее генотипа b , если существует такой индекс i , что для всех $j < i$ $a_j = b_j$, а a_i идет в алфавите раньше, чем b_i . Если для всех i $a_i = b_i$, то мощнее тот генотип, в котором меньше генов.

Шраму стало интересно, какие генотипы могли быть при рождении у каждой гиены, если известно, что при рождении генотип i -ой гиены был мощнее, чем генотип $(i+1)$ -ой. Определите число вариантов наборов начальных генотипов гиен, удовлетворяющих всем описанным условиям, взятое по модулю 1 000 000 007.

Формат входного файла

В самой первой строке написано число n — число гиен ($1 \leq n \leq 10$). Далее следует n строк, в каждой из которой записана строка s_i , обозначающая генотип i -ой гиены в данный момент. Длина каждой строки не превосходит 13.

Формат выходного файла

Выведите ответ на задачу по модулю 1 000 000 007.

Пример

<code>scar.in</code>	<code>scar.out</code>
2 aaa aa	1
2 ab abc	13
3 qwerty qwerty ytrewq	87506

Задача J. Стражи антилопы и гиены

Имя входного файла:	<code>stdin</code>
Имя выходного файла:	<code>stdout</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Это интерактивная задача.

На ежегодное совещание по делам Саванны, проходящее в этом году во владениях Шрама, приглашён также и Муфаса. Муфаса считает, что путь из его владений во владения Шрама небезопасен и не намерен идти напрямик. Вместо этого, он решил поставить на своём пути антилоп, которые будут следить за безопасностью и, в случае непредвиденных проблем, заблаговременно предупредят Муфасу. Шрам прознал про это и решил помешать Муфасе организовать безопасный путь, приказывая своим гиенам занимать стратегически важные позиции, не отдавая их антилопам Муфасы.

Будем считать, что Саванна является клетчатым прямоугольником $n \times m$, владения Шрама находятся в клетке $(1, 1)$, а владения Муфасы — в клетке (n, m) . Владения Шрама и Муфасы уже считаются Муфасой безопасными. Шрам и Муфаса могут по очереди приказывать своим приспешникам (антилопам и гиенам) занимать некоторые клетки этого поля. Если некоторая клетка занята антилопами, она считается безопасной. Ни одна клетка, занятая гиенами, никогда не будет безопасной.

Муфаса хочет добиться того, чтобы из его владений во владения Шрама существовал путь, проходящий только по безопасным клеткам, каждые две последовательные клетки в котором являются соседними по стороне. Шрам же хочет не дать Муфасе возможности это сделать. Помогите Шраму добиться своего.

Протокол взаимодействия с программой жюри:

В первой строке входного файла заданы числа n и m ($3 \leq n, m \leq 20$) — размеры Саванны. Далее, до конца работы программы, выполняются следующие действия:

- Если вы считаете, что с данной расстановкой гиен для Муфасы не существует и уже никогда не будет существовать безопасного пути, выведите `x_X` и завершите работу программы.
- Иначе, выведите пару целых чисел (x, y) — координаты очередной гиены, такую, что:
 - $1 \leq x \leq n, 1 \leq y \leq m$.
 - $(x, y) \neq (1, 1)$.
 - $(x, y) \neq (n, m)$.
 - (x, y) не выводилось ранее ни программой участника, ни программой жюри.

После этого считайте со стандартного ввода пару чисел, соблюдающую такие же ограничения — координаты клетки, занятой очередной антилопой.

Ваша программа должна сделать не более $\lfloor \frac{nm}{2} \rfloor + 1$ действий, при этом, последним должен быть вывод `x_X`, и такой вывод должен быть всего один.

Пример

stdin	stdout
3 3	2 2
2 1	3 1
3 2	x_X
4 4	1 4
2 1	2 3
1 2	3 2
2 2	4 1
3 2	x_X

Комментарий

Для корректной работы программы после каждой операции вывода данных вам необходимо делать следующие операции:

- В языке Pascal: `flush(output);`
- В C/C++: `fflush(stdout);`
- В Java: `System.out.flush();`
- В Python: `sys.stdout.flush();`

Кроме этого, не забывайте после каждой выведенной строки ставить перевод строки.