

Задача А. Z-функция

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

Вам дана строка из строчных латинских букв. От вас требуется вычислить Z-функцию данной строки.

Формат входных данных

Вводится строка, состоящая из строчных латинских букв. Длина строки не превышает 10^6 .

Формат выходных данных

Требуется вывести Z-функцию данной строки.

Пример

стандартный ввод	стандартный вывод
abacaba	7 0 1 0 3 0 1

Замечание

Предполагается, что значение Z-функции для первого символа равно длине строки.

Задача В. Префикс-функция

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

Вам дана строка из строчных латинских букв. От вас требуется вычислить префикс-функцию данной строки.

Формат входных данных

Вводится строка, состоящая из строчных латинских букв. Длина строки не превышает 10^6 .

Формат выходных данных

Требуется вывести префикс-функцию данной строки.

Пример

стандартный ввод	стандартный вывод
abacaba	0 0 1 0 1 2 3

Замечание

Предполагается, что значение префикс-функции для первого символа равно нулю.

Задача С. Неточное совпадение

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

Даны строки p и t . Требуется найти все вхождения строки p в строку t в качестве подстроки с точностью до возможного несовпадения одного символа.

Формат входных данных

Первая строка входного файла содержит p , вторая — t ($1 \leq |p|, |t| \leq 10^6$). Строки состоят из букв латинского алфавита.

Формат выходных данных

В первой строке выведите количество вхождений строки p в строку t . Во второй строке выведите в возрастающем порядке номера символов строки t , с которых начинаются вхождения p . Символы нумеруются с единицы.

Пример

стандартный ввод	стандартный вывод
aaaa	4
Caabdbaaaa	1 2 6 7

Задача D. Подпалиндромы

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

Строка называется палиндромом, если она читается одинаково как слева направо, так и справа налево. Например, строки "abba", "kek" являются палиндромами.

Дана строчка. Ее подстрокой называется некоторая непустая последовательность подряд идущих символов. Напишите программу, которая определит, сколько подстрок данной строки является палиндромами.

Формат входных данных

Вводится одна строка, состоящая из маленьких латинских букв. Длина строки не превышает 100 000 символов.

Формат выходных данных

Выведите одно число – количество подстрок данной строки, являющихся палиндромами.

Примеры

стандартный ввод	стандартный вывод
aaa	6
aba	4

Задача Е. Мультимножество Василия

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

У автора уже закончились истории про Василия, поэтому он просто написал формальную постановку задачи.

У вас есть q запросов и мультимножество A , изначально содержащее только число 0. Запросы бывают трёх видов:

- «+ x » — добавить в мультимножество A число x .
- «- x » — удалить одно вхождение числа x из мультимножества A . Гарантируется, что хотя бы одно число x в этот момент присутствует в мультимножестве.
- «? x » — вам даётся число x , требуется вычислить $\max_{y \in A} x \oplus y$, то есть максимальное значение побитового исключающего ИЛИ (также известно как XOR) числа x и какого-нибудь числа y из мультимножества A .

Мультимножество — это множество, в котором разрешается несколько одинаковых элементов.

Формат входных данных

В первой строке входных данных содержится число q ($1 \leq q \leq 200\,000$) — количество запросов, которые требуется обработать Василию.

Каждая из последующих q строк входных данных содержит один трёх символов «+», «-» или «?» и число x_i ($1 \leq x_i \leq 10^9$). Гарантируется, что во входных данных встречается хотя бы один запрос «?».

Обратите внимание, что число 0 всегда будет присутствовать в мультимножестве.

Формат выходных данных

На каждый запрос типа «?» выведите единственное целое число — максимальное значение побитового исключающего ИЛИ для числа x_i и какого-либо числа из мультимножества A .

Пример

стандартный ввод	стандартный вывод
10	11
+ 8	10
+ 9	14
+ 11	13
+ 6	
+ 1	
? 3	
- 8	
? 3	
? 8	
? 11	

Замечание

После первых пяти операций в мультимножестве A содержатся числа 0, 8, 9, 11, 6 и 1.

Ответом на шестой запрос будет число $11 = 3 \oplus 8$ максимальное из чисел $3 \oplus 0 = 3$, $3 \oplus 9 = 10$, $3 \oplus 11 = 8$, $3 \oplus 6 = 5$ и $3 \oplus 1 = 2$.

Задача F. Библиотека

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

Прошел почти год с момента, как Рик оказался на Флорине, однако его сознание никак не прояснялось. Воспоминания о прошлом были спрятаны в глубинах его разума, а может и вовсе утеряны. Однако сегодня что-то случилось. Рик вспомнил: у него была работа. Он анализировал Ничто. Наверное, Ничто — это космос, а значит Рик в прошлом был космоаналитиком. А еще Рик вспомнил, что все жители Флорины должны были погибнуть, но он не знал, почему.

Резидента Мирлина Теренса заинтересовала эта информация, поэтому он взял Рика с собой в библиотеку Верхнего города. Может быть, какая-нибудь литература по космоанализу могла бы вернуть ему память? Теренс не знал, что пропавшего космоаналитика активно ищут, а потому в библиотеке был получен приказ сообщать о любых посетителях, которые спросят о такой литературе. Библиотекарь отследил запросы наших героев в поисковой системе и поспешил вызвать патрульных.

Тем временем Теренс предложил Рiku ознакомиться с книгой известного автора Врийта "Трактат об инструментальном космоанализе". Рiku книга определенно показалась знакомой, особенно его привлекла строка s . Смысла самой строки, он, к сожалению, не понимал, однако в ее частях он видел что-то знакомое. Чтобы разобраться подробнее, Рик решил изучить все подстроки s . Однако изучать равные подстроки не было смысла, а остальные стоило как-либо систематизировать. Например, расставить их по длине и в алфавитном порядке. Поэтому Рик попросил вас узнать, сколько у данной строки существует пар подстрок s_1 и s_2 равной длины, таких, что $s_1 < s_2$ лексикографически.

Формат входных данных

Задана строка s , состоящая из строчных латинских букв ($|s| \leq 2500$).

Формат выходных данных

Выведите одно число — количество искомых пар подстрок.

Пример

стандартный ввод	стандартный вывод
abac	9

Замечание

Рассмотрим подстроки длины 1. Имеется две подстроки "a", каждая из которых меньше подстрок "b" и "c". Также подстрока "b" меньше подстроки "c". Отсюда получаем 5 пар искомых подстрок.

Теперь рассмотрим подстроки длины 2. Подстрока "ab" меньше подстрок "ba" и "ac", а строка "ac" меньше, чем строка "ba". Отсюда получаем еще 3 пары.

Наконец, рассмотрим подстроки длины 3. Подстрока "aba" меньше подстроки "bac".

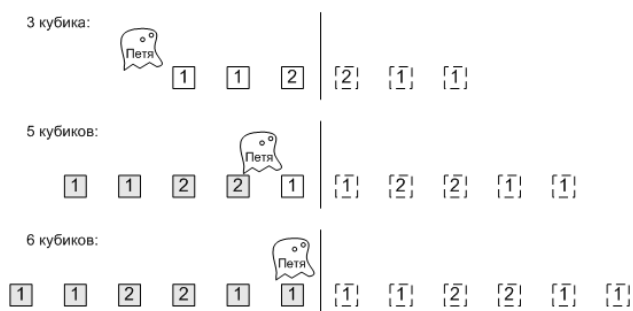
Таким образом, суммарно получаем 9 искомых пар подстрок.

Задача G. Кубики

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

Привидение Петя любит играть со своими кубиками. Он любит выкладывать их в ряд и разглядывать своё творение. Однако недавно друзья решили подшутить над Петей и поставили в его игровой комнате зеркало. Ведь всем известно, что привидения не отражаются в зеркале! А кубики отражаются.

Теперь Петя видит перед собой N цветных кубиков, но не знает, какие из этих кубиков настоящие, а какие — всего лишь отражение в зеркале.



Помогите Пете! Выясните, сколько у него может быть кубиков. Петя видит отражение всех кубиков в зеркале и часть кубиков, которая находится перед ним. Часть кубиков может быть позади Пети, их он не видит.

Формат входных данных

Первая строка входного файла содержит два целых числа: N ($1 \leq N \leq 100\,000$) и количество различных цветов, в которые могут быть раскрашены кубики, — M ($1 \leq M \leq 100\,000$). Следующая строка содержит N целых чисел от 1 до M — цвета кубиков.

Формат выходных данных

В выходной файл выведите в порядке возрастания все такие K , что у Пети может быть K кубиков.

Пример

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

Задача Н. Название команды

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

Во Всеберляндской олимпиаде по программированию могут участвовать команды в составе n человек. Для участия каждой команде нужно выбрать название.

Участники одной из команд составили название, включающее каждое из их имен, и записали его в строку t длины m . Однако, такое название получилось слишком длинным! Поэтому они хотят сократить его следующим образом:

- выбрать префикс строки t минимальной длины, который включал бы все их имена без пересечений.

Префиксом строки называется строка, полученная удалением нескольких (возможно, нуля) последних символов из исходной строки.

Некоторое множество подстрок входит в строку *без пересечений*, если никакой символ не принадлежит двум подстрокам одновременно. Например, подстроки «a» и «bc» входят в строку «ababc» без пересечений, а подстроки «aba» и «abc» — нет.

Если название команды возможно сократить — выведите длину минимального подходящего префикса.

Формат входных данных

Первая строка входных данных содержит два целых числа: n ($2 \leq n \leq 8$) — количество участников в команде, и m ($2 \leq m \leq 2 \cdot 10^5$) — длину названия, составленного командой.

Вторая строка входных данных содержит строку t , состоящую из строчных латинских букв — название, составленное командой.

Далее следуют n строк, каждая из которых содержит строку s_i , состоящую из строчных латинских букв ($1 \leq |s_i| < m$) — имя i -го участника команды ($1 \leq i \leq n$). Длина строки s_i обозначается как $|s_i|$.

Гарантируется, что сумма значений $|s_i|$ не превосходит значения m , и в строку t можно поместить имена всех участников без пересечений.

Формат выходных данных

Выведите:

- Число -1 , если название команды невозможно сократить;
- Иначе длину минимального префикса строки t , в который могут войти имена всех участников без пересечений.

Примеры

стандартный ввод	стандартный вывод
4 22 omihaakrisssashaannaaa kris miha sasha anna	20
2 16 coolnickgoodalex nick alex	-1
3 7 bababab a bab b	5

Задача I. Профессиональный декоратор заборов

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

Вася работает подмастерьем в известной студии. Недавно ему поручили помогать молодому, но подающему большие надежды художественному декоратору заборов и изгородей Витезславу Смолокурову. Миссия эта очень ответственная, и от ее выполнения зависит Васино будущее.

Стиль Смолокурова очень необычен, а его работы пользуются большим спросом. Процесс работы разделен на два этапа. На первом этапе Вася делает заготовку — длинный забор, который состоит из набора цветных вертикальных планок. На втором этапе Витезслав приступает к работе.

Для того, чтобы придать забору более спокойный и гармоничный вид, он несколько раз производит следующую операцию: выбирает некоторый цвет и отрезок, после чего перекрашивает этот отрезок забора в выбранный цвет. По своей творческой натуре, Смолокуров может в корне менять концепцию узора по несколько раз за час, поэтому иногда он перекрашивает одну и ту же планку несколько раз. Кроме того, Витезслав не хочет, чтобы какой-то узор повторялся слишком часто. Для того, чтобы избежать этого, он иногда проверяет, не совпадает ли один отрезок забора с другим.

Несложно догадаться, что и перекрашивание, и проверки осуществляет Вася. Работа эта не самая простая, поэтому Вася просит ему помочь хотя бы с проверками на совпадение.

Формат входных данных

Первая строка входного файла содержит одно целое число n — количество планок в заборе ($1 \leq n \leq 100\,000$). Вторая строка содержит n целых чисел, разделенных пробелами — цвета соответствующих планок.

Третья строка входного файла содержит одно целое число m — количество сравнений и перекрашиваний ($1 \leq m \leq 100\,000$). Следующие m строк содержат описания заданий, который Вася получает от Витезслава: четыре целых числа q, l, r и k .

В случае перекрашивания $q = 0$. Эта запись означает перекрашивание всех планок с l по r включительно в цвет k ($1 \leq l \leq r \leq n$). В запросе на сравнение $q = 1$. Эта запись означает сравнение кусков забора длины k начиная с позиций l и r соответственно ($1 \leq l, r \leq n - k + 1, k > 0$).

Все числа во входном файле положительные и не превышают 100 000.

Формат выходных данных

Выведите одну строку: для каждого запроса на сравнение выведите '+' в случае совпадения соответствующих кусков забора и '-' в противном случае.

Примеры

стандартный ввод	стандартный вывод
7 1 2 1 3 1 2 1 3 0 4 5 2 1 3 1 2 1 3 1 3	+-
2 1 2 5 1 1 2 1 0 2 2 1 1 1 2 1 0 1 2 3 1 1 1 2	-++

Задача J. Автодополнение

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

Тор все-таки решился на покупку мобильного телефона, чтобы связываться со Мстителями было проще. В процессе написания смс-сообщений Тор не мог не заметить интересную и довольно полезную функцию — автодополнение. Эта функция по написанному непустому началу слова предлагает на выбор три самых популярных слова из своего словаря с таким же началом, чтобы для ускорения написания сообщения пользователь мог согласиться на одно из них, а не дописывать слово полностью (если таких слов меньше трех, предлагаются все возможные слова). Таким образом, пользователь при наборе сообщения может делать три действия — написать новую букву, удалить букву из конца написанного текста и согласиться на один из не более трех вариантов автодополнения.

Тони Старк любезно согласился взломать телефон Тора, чтобы узнать все слова из словаря, который использует функция автодополнения. Таких слов оказалось ровно n штук, а также оказалось, что автодополнение предлагает 3 самых первых слова из списка, то есть чем раньше слово находится в списке, тем популярнее оно считается. Тони также подметил, что система автодополнения устроена так, что если пользователь набрал слово s , и оно есть в словаре, то система не будет его предлагать.

Тор так заинтересовался автодополнением, что захотел узнать все способы набора своего сообщения s с помощью него. Понятно, что таких способов бесконечно много, поэтому Тор хочет найти все способы набора сообщения, используя не более k действий. Тор действует довольно логично и не собирается набирать новую букву так, что получившийся текст не будет являться префиксом желаемого сообщения s (однако он может согласиться на автодополнение, которое не будет являться префиксом s). Также после того, как сообщение s набрано, Тор может продолжить набор сообщения, если на текущий момент он сделал меньше k действий (а может и не продолжать и остановиться).

Для начала Тор решил ограничиться сообщениями, состоящими только из одного слова и находить не варианты набора сообщения, а только их количество по модулю $10^9 + 7$. Помогите ему с этой задачей — по данному слову s , состоящему из строчных латинских букв, и числу k найдите количество способов написать слово s не более чем за k действий.

Формат входных данных

В первой строке содержится число n — количество слов из словаря ($1 \leq n \leq 100$).

В следующих n строках содержатся слова w_i из словаря ($1 \leq |w_i| \leq 100$). Гарантируется, что каждое слово из словаря состоит только из строчных латинских букв, а также что суммарная длина слов из словаря не превышает 10^3 .

В $n + 2$ строке содержится строка s — слово, состоящее только из строчных латинских букв, которое хочет набрать Тор ($1 \leq |s| \leq 100$).

В последней строке содержится число k — максимальное количество действий (набор одного символа, удаление одного символа из конца текущего текста или соглашение на один из вариантов автодополнения), которое можно сделать ($1 \leq k \leq 10^3$).

Формат выходных данных

В единственной строке выведите количество способов набрать слово s по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
4 abacaba ababb abcabac babacb abacb 6	3
1 exactword exactword 4	7

Замечание

В первом примере возможны следующие три варианта написания слова «abacb»:

- 6 действий: набрать букву «a» → согласиться на автодополнение «abacaba» → удалить 3 раза последнюю букву → набрать букву «b»;
- 6 действий: набрать букву «a» → согласиться на автодополнение «ababb» → удалить 2 раза последнюю букву → набрать букву «c» → набрать букву «b»;
- 5 действий: набрать слово по одной букве.

Задача К. Анаграммы-2

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

Недавно Человек-Невидимка от нечего делать прогуливался по крышам домов и случайно подслушал интересный разговор, доносящийся из открытого окна последнего этажа. Разговаривали два человека, одного из которых звали «Нолик», а второго — «Симка». «Странные имена», — подумал Человек-Невидимка. Но для него это было неважно, намного интереснее была тема разговора — это было что-то, связанное с программированием, а он никогда не мог пройти мимо такого соблазна.

Внимательно все послушав, Человек-Невидимка понял, что суть задачи, которую обсуждали эти два странных человека, состоит в следующем: по данному массиву-шаблону и массиву-тексту надо было понять, существует ли такой подотрезок текста, совпадающий с массивом-шаблоном как анаграмма. Под анаграммами в данном случае понимались два слова, в которых можно как-то переставить буквы, чтобы они стали одинаковыми. Оценив задачу, Человек-Невидимка понял, что она для него слишком простая, поэтому он решил усложнить ее. После некоторых раздумий, ему в голову пришла следующая ее модификация: по данным двум массивам требовалось найти такое максимальное число k , что в первом и втором массивах существуют подотрезки длиной k , совпадающие как анаграммы. Но эта задача уже оказалась Человеку-Невидимке не по силам, поэтому он попросил у вас помощи в решении этой задачи.

Формат входных данных

В первой строке дано число n ($1 \leq n \leq 1\,000$) — длина первого массива.

Во второй строке через пробел заданы n чисел a_i ($1 \leq a_i \leq 100\,000$) — первый массив.

В третьей строке дано число m ($1 \leq m \leq 1\,000$) — длина второго массива.

В четвертой строке через пробел заданы m чисел b_i ($1 \leq b_i \leq 100\,000$) — второй массив.

Формат выходных данных

В единственной строке выведите максимальная длина подотрезков, совпадающих как анаграммы.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 3 3 2 1	3
3 1 2 3 3 4 5 6	0

Задача L. Яся и таинственное дерево

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	512 мегабайт

Яся гуляла по лесу и совершенно случайно нашла дерево на n вершинах. Дерево — это связный неориентированный граф, в котором отсутствуют циклы.

Рядом с деревом девочка нашла древний манускрипт, на котором записаны m запросов. Запросы бывают двух видов.

Первый вид запросов описывается числом y . Вес **каждого** ребра в дереве заменяется на побитовое исключающее «ИЛИ» веса этого ребра и числа y .

Второй вид описывается вершиной v и числом x . Яся выбирает вершину u ($1 \leq u \leq n$, $u \neq v$) и мысленно проводит в дереве двунаправленное ребро веса x из v в u .

Затем Яся находит простой цикл в получившемся графе и считает побитовое исключающее «ИЛИ» от всех рёбер на нём. Она хочет выбрать такую вершину u , чтобы посчитанное значение было **максимально**. Это посчитанное значение и будет ответом на запрос. Можно показать существование и единственность такого цикла в указанных ограничениях (независимо от выбора u). Если ребро между v и u уже существовало, простым циклом будет путь $v \rightarrow u \rightarrow v$.

Обратите внимание, что запрос второго типа выполняется *мысленно*, то есть дерево после него **никак** не меняется.

Помогите Ясе ответить на все запросы.

Формат входных данных

В первой строке дано целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных.

Далее следуют описания наборов.

В первой строке каждого набора даны целые числа n , m ($2 \leq n \leq 2 \cdot 10^5$, $1 \leq m \leq 2 \cdot 10^5$) — количество вершин в дереве и количество запросов.

В следующих $n-1$ строках каждого набора даны целые числа v , u , w ($1 \leq v, u \leq n$, $1 \leq w \leq 10^9$) — концы некоторого ребра в дереве и его вес.

Гарантируется, что заданный набор рёбер образует дерево.

В следующих m строках каждого набора описаны запросы:

- $\wedge y$ ($1 \leq y \leq 10^9$) — параметр запроса первого типа;
- $? v x$ ($1 \leq v \leq n$, $1 \leq x \leq 10^9$) — параметры запроса второго типа.

Гарантируется, что сумма n по всем наборам входных данных не превосходит $2 \cdot 10^5$. То же самое гарантируется для m .

Формат выходных данных

Для каждого набора входных данных выведите ответы на запросы второго типа.

Примеры

стандартный ввод	стандартный вывод
2 3 7 1 2 1 3 1 8 ~ 5 ? 2 9 ~ 1 ? 1 10 ~ 6 ? 3 1 ? 2 9 5 6 1 2 777 3 2 2812 4 1 16 5 3 1000000000 ~ 4 ? 3 123 ? 5 1000000000 ~ 1000000000 ? 1 908070 ? 2 1	13 15 11 10 1000000127 2812 999756331 999999756
3 8 4 8 6 3 6 3 4 2 5 4 7 6 2 7 1 10 4 1 4 5 1 2 ~ 4 ~ 7 ? 7 8 ? 4 10 5 6 3 1 4 2 3 9 4 3 6 5 2 10 ? 5 7 ~ 1 ~ 8 ? 4 10 ? 1 9 ? 3 6 4 2 2 1 4 4 3 5 2 3 4 ~ 13 ? 1 10	14 13 13 8 11 11 10