

Задача А. Задачка на строчечки

Имя входного файла: inputik.txt
Имя выходного файла: outputik.txt
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайтов

В стандартном поточике вводика или файлик
е
кой. На следующей строчечке программочк
ь а і
н к; эти словечки мы назовём словарик п
е о о н р
н в utputik.txt N строчечек. В і-ой ма и
д о о . й т
о к - количюсик (сколько штучеч с д і
х у к е т В ё k
с б и о ем через пробельчик для к р а т .
и л к т) о ш т
х й л а в всех вхожденъечек к ч а ч х
м и а е з о а в е и t
ё к ф с к ртированном про э ж х ч п с
в с и , и о я т д о к р е в
о н в ч й л с рочечек нач д о о же о л а
з и о а т т и о й г д г к ш
а т и е к ч о с ничек. н ч о е п р о а
на л о ъ а и а к с н р а
л и в н н в х де с ястю е т в ъ о м N п
ы р е е . р х е г м , р
м х а е н и ю св и к и с к е д н и о о ч р о о
и к п д к у ч ж е а ч а г
ю к и о и к ъ н е н д о x с и в и к ч е д к м к р
у ъ д : х с е м а в а
р н о к с к е д н и е т и д е в ы в а к ч и ъ н с о м
о е в о и т ч д с м
т л ы л в а к и р а в о л с з и і и к ч е ч о р к о л o
о а в е а л e ч
к м с и ч о к ъ л o x с e n и т с e в ы в а н ж л o d ж д к
к н у а
, е е ч и ч o т o п й н т р a d н a т c a н и т с e в ы в а ю
к ж щ н
о х e т з и у к ч e в o л c o п - х a k ч e ч o р т c N х и а
в й
о к у б х и к с н и т a л х и к ъ н e л a m з и у k ч e ч o р т c т ё d

Формат входных данных

В стандартном поточике ввода или файлике `inputik.txt` ваша программочка найдёт строчечку из маленьких латинских буковок, которую мы назовём исходненькой. На следующей строчечке программочка найдёт числище N ($1 \leq N \leq 1\,000\,000$), а в следующих N строчечках — по словечку из тех же маленьких латинских буковок; эти словечки мы назовём словариком. Суммарненькая суммочка длинниц словечек из словарика не превосходит $1\,000\,000$.

Формат выходных данных

Ваша программочка должна вывести на стандартный поточичек вывода или в файл `outputik.txt` N строчечек. В i -ой строчечке программочка должна вывести несколько чиселок: первое чиселко — количюсик (сколько штукечек) вхожденьчек строчечки i из словарика в исходненькой, затем через пробельчик для каждого вхожденьчика выведите индексики началиков всех вхожденьчик этой строчечки в исходненькую в отсортированном порядочке. Индексики всех строчечек начинаются с единичек. Няшечки-преподавашечки гарантируют, что колчюсик вхожденьчик не превосходит $1\,000\,000$.

Пример

inputik.txt	outputik.txt
abrachkacadabrachka	2 1 12
4	1 9
abrachka	2 1 12
cadabrachka	0
ab	
marazmik	

Задача В. Палиндромы

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

Вам дана строка s из маленьких английских букв. Определим для подстроки *красоту* как количество вхождений подстроки в строку, умноженное на длину подстроки. Для данной строки найдите максимальную красоту среди всех её палиндромных подстрок.

Формат входных данных

Первая строка содержит строку s . Длина строки не превосходит $3 \cdot 10^5$.

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
abacaba	7
www	4

Задача С. Бинарная игра

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

Искандер и Оля любят придумывать ребусы. Но больше, чем придумывать ребусы, они любят придумывать какие-нибудь игры на строках. Вот и сейчас им в голову пришла забавная игра со следующими правилами:

- Выбирается какой-то набор *запрещённых* двоичных (состоящих из нулей и единиц) строк f_1, f_2, \dots, f_n .
- Выбирается некоторая стартовая бинарная строка s , такая что ни одна из запрещённых строк не входит в неё как подстрока.
- Игроки по очереди дописывают в конец строки s по одному символу «0» или «1». Оля ходит первой.
- Проигрывает тот, после чьего хода хотя бы одна из запрещённых строк f_1, f_2, \dots, f_n входит в s как подстрока.
- В случае если при оптимальной игре обоих игроков игра может продолжаться сколь угодно долго, то объявляется ничья.

Вы обожаете портить другим людям их любимые развлечения, поэтому решили написать программу, которая будет определять исход игры по заданному набору запрещённых строк и стартовой строке s .

Формат входных данных

В первой строке входных данных записаны два целых числа n и m ($0 \leq n \leq 100\,000$, $0 \leq m \leq 1\,000\,000$) — количество запрещённых строк и изначальная длина строки s .

В каждой из последующих n строк содержится одна запрещённая строка. Гарантируется, что все эти строки непусты, состоят из символов «0» и «1» и никакая из них не является подстрокой строки s . Дополнительно гарантируется, что **суммарная длина** всех запрещённых строк не превосходит 1 000 000.

В последней строке входных данных записана стартовая строка s длины m , состоящая только из символов «0» и «1». Обратите внимание, строка s может быть пустой, в этом случае соответствующая строка входных данных отсутствует (в том числе символ перевода строки). Длина s не превосходит 1 000 000.

Формат выходных данных

В зависимости от результата игры при оптимальной игре обоих игроков выведите:

- «Olya» (без кавычек), если Оля может победить вне зависимости от того как будет играть Искандер. Напомним, что Оля ходит первой.
- «Iskander» (без кавычек), если Искандер может победить не зависимо от ходов Оли.
- «Friendship» (без кавычек), если при оптимальной игре обоих игроков игра будет продолжаться бесконечно долго.

Примеры

стандартный ввод	стандартный вывод
1 0 1	Friendship
3 1 000 001 011 0	Olya
2 3 1001 000 100	Iskander

Замечание

В первом примере строка s изначально пустая. Любой из игроков может не проиграть на любом ходу просто приписав к s символ «0».

Задача D. Двоичные строки

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

Дано n непустых двоичных строк s_1, s_2, \dots, s_n и еще m непустых двоичных строк t_1, t_2, \dots, t_m . Требуется определить, существует ли двоичная строка S , такая что:

- Существуют такие i и j ($1 \leq i < j \leq n$), что строки s_i и s_j являются подстроками строки S .
- Для всех i ($1 \leq i \leq m$) строка t_i не является подстрокой строки S .

Формат входных данных

Первая строка содержит одно целое число t ($1 \leq t \leq 100\,000$) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка описания содержит два целых числа n и m ($2 \leq n \leq 100\,000$, $1 \leq m \leq 100\,000$).

Каждая из следующих n строк содержит непустую двоичную строку s_i .

Каждая из следующих m строк содержит непустую двоичную строку t_i .

Гарантируется, что сумма $n + m$ по всем наборам входных данных не превосходит 100 000. Также гарантируется, что сумма длин всех строк по всем наборам входных данных не превосходит 10^6 .

Формат выходных данных

Для каждого набора входных данных выведите «Yes», если требуемая строка S существует, и «No» в противном случае.

Пример

стандартный ввод	стандартный вывод
2	Yes
3 2	No
100	
001	
010	
1001	
000	
2 4	
100	
001	
010	
1001	
000	
11	

Замечание

В первом наборе входных данных можно взять в качестве строки S строку «0100». Строки s_1 и s_3 встречаются в ней в качестве подстрок. Строки t_1 и t_2 не встречаются в ней в качестве подстрок.

Задача E. Цензура

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

Посчитайте, сколько строк над алфавитом из n символов длины m не содержат ни одной подстроки из заданного множества “запрещенных” строк.

Формат входных данных

В первой строке написаны целые числа n ($1 \leq n \leq 100$) — количество символов в алфавите, m ($1 \leq m \leq 100$) — длина искомых строк и p ($0 \leq p \leq 10$) — количество “запрещенных” подстрок. Следующая строка содержит n символов с кодами больше 32 — буквы алфавита. Далее идет p “запрещенных” строк, длины которых не превосходят $\min(m, 10)$ символов. Строки целиком состоят из символов алфавита.

Формат выходных данных

В первой строке выведите ответ на задачу по модулю 1000000007.

Пример

стандартный ввод	стандартный вывод
2 3 1 ab bb	5

Задача F. Подстрока

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

Недавно разведка перехватила зашифрованное сообщение — строку s . Все ресурсы аналитического центра, в котором вы работаете, были брошены на его декодирование. Ваш отдел занимается шифрами нового поколения. На данный момент известно всего n таких шифров. Для каждого из них есть три характерных параметра — целые числа l, r и строка t . Пусть строка g была получена в результате применения этого метода. Тогда строка $g_l g_{l+1} \dots g_{r-1} g_r$ (здесь g_i — это i -й символ строки g) содержит t как подстроку.

Вам поручено определить для каждого типа шифрования, могло ли сообщение s быть получено в результате его применения.

Формат входных данных

Первая строка входного файла содержит строку s ($1 \leq |s| \leq 100\,000$, где $|s|$ — длина строки s).

Вторая строка входного файла содержит целое число n — количество типов шифрования ($1 \leq n \leq 100\,000$). Последующие n строк содержат по два целых числа l_i, r_i и строку t_i , разделенные пробелами — характерные параметры i -го метода шифрования ($1 \leq l_i \leq r_i \leq |s|$).

Все строки состоят из строчных букв латинского алфавита. Суммарная длина всех t_i не превосходит 100 000.

Формат выходных данных

Выведите одну строку — для каждого типа шифрования «+», если сообщение s могло быть получено в результате его применения, или «-» в противном случае.

Пример

стандартный ввод	стандартный вывод
frommarsiam 3 6 10 i 2 11 am 1 9 human	++-

Задача G. Палиндромчики

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

Необходимо для каждого префикса данной строки найти количество различных ее подстрок-палиндромов.

Формат входных данных

Дана строка s ($1 \leq |s| \leq 10^5$).

Формат выходных данных

Выведите $|s|$ чисел, i -е из которых равно количеству различных подстрок-палиндромов префикса строки s длины i .

Пример

стандартный ввод	стандартный вывод
aba	1 2 3

Задача Н. Загадочное устройство

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

У Димы есть устройство, содержащее $n + m$ кнопок и экран. На экране отображается строка s , изначально она пустая. Если нажать на кнопку с номером i ($1 \leq i \leq n$), то в конец строки s дописется строка w_i . Если нажать на кнопку с номером $n + i$ ($1 \leq i \leq m$), то из конца строки s удалится a_i символов. Если в строке s было меньше a_i символов, она станет пустой.

У Димы есть q любимых строк t_i . Для каждой из них он хочет узнать, можно ли в результате нескольких нажатий на кнопки из пустой строки s сделать строку равную t_i .

Формат входных данных

В первой строке дано два целых числа n и m — количество кнопок первого и второго типа ($1 \leq n, m \leq 100\,000$).

В следующих n строках даны непустые строки w_i , состоящие из строчных английских букв. Сумма длин всех строк w_i не превышает 10^6 .

В следующих m строках даны числа a_i ($1 \leq a_i \leq 10^6$).

В следующей строке дано целое число q — количество любимых строк Димы ($1 \leq q \leq 100\,000$).

В следующих q строках даны непустые строки t_i , состоящие из строчных английских букв. Сумма длин всех строк t_i не превышает 10^6 .

Формат выходных данных

Для каждой строки t_i в новой строке выведите «Yes», если можно s сделать равным t_i , и «No» иначе.

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены.

Подзадача	Баллы	Ограничения	Необходимые подзадачи	Информация о проверке
1	12	$\sum w_i \leq 1\,000, \sum t_i \leq 1\,000,$ $m = 1, w_i = a_1$		первая ошибка
2	13	$\sum w_i \leq 1\,000, \sum t_i \leq 1\,000,$ $m = 1, a_1 = 1$		первая ошибка
3	19	$\sum w_i \leq 1\,000, \sum t_i \leq 1\,000$	1	первая ошибка
4	29	$\sum w_i \leq 100\,000, \sum t_i \leq 100\,000$	1, 2	первая ошибка
5	27	Без дополнительных ограничений	1–3	первая ошибка

Пример

стандартный ввод	стандартный вывод
2 1	Yes
bba	Yes
abacaba	Yes
2	No
4	
b	
ababbab	
abacabba	
caba	