

Задача А. Подстроки

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

Дано K строк из маленьких латинских букв. Требуется найти их наибольшую общую подстроку.

Формат входных данных

В первой строке число K ($1 \leq K \leq 10$).

В следующих K строках — собственно K строк (длины строк от 1 до 10 000).

Формат выходных данных

Выведите наибольшую общую подстроку.

Пример

стандартный ввод	стандартный вывод
3 abacaba mucabarchive acabistrue	cab

Задача В. 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 программч
ь а і
н к; эти словечки мы назовём словарик п
е о о н р
н в utputik.txt N строчечек. В i-ой ма u
д о о . й t
о к - количюсик (сколько штукеч 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 t
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 ж е 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
н а 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 a
, e e ч и ч o т o п ѡ н т р a d н a t c a n и т с e в ы в а ю
к ж щ н
о х e t з и у к ч e в o l s o p - х a k ч e ч o р т c N х и a
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 к ч e ч o р т c т ё d

Формат входных данных

В стандартном поточике вводика 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 выводика или в файл `outputik.txt` N строчечек. В i -ой строчечке программочка должна вывести несколько чиселок: первое чиселко — количюсик (сколько штучечек) 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

<code>inputik.txt</code>	<code>outputik.txt</code>
<code>abrachkacadabrachka</code>	<code>2 1 12</code>
<code>4</code>	<code>1 9</code>
<code>abrachka</code>	<code>2 1 12</code>
<code>cadabrachka</code>	<code>0</code>
<code>ab</code>	
<code>marazmik</code>	

Задача С. Логотип

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

В целях своей рекламной кампании большая компания в городе Гдыня хочет разместить в городе свой логотип. Компания хочет потратить весь свой годовой рекламный бюджет на логотип, поэтому он должен быть действительно большим. Один из управляющих решил использовать целые здания как части логотипа.

Логотип состоит из n вертикальных полос разной длины. Полосы пронумерованы числами от 1 до n слева направо. Логотип описан перестановкой (s_1, s_2, \dots, s_n) чисел $1, 2, \dots, n$. Полоса под номером s_1 самая короткая, полоса под номером s_2 самая короткая среди оставшихся, и так далее, полоса s_n самая длинная. Нам не будет важно, какая именно длина каждой полосы.

На главной улице Гдыни m зданий. Удивительно, но высоты всех m зданий различны. Встала проблема, найти все позиции где логотип соответствует зданиям.

Помогите компании и найдите все непрерывные отрезки последовательности домов, которые соответствуют логотипу. Непрерывный отрезок домов соответствует логотипу, если здание под номером s_1 в этом отрезке самое низкое, здание под номером s_2 самое низкое среди оставшихся на отрезке, и так далее. Например, последовательность зданий с высотами 5, 10, 4 соответствует логотипу, описанному перестановкой $(3, 1, 2)$, так как здание номер 3 (высоты 4) самое низкое, здание 1 (высоты 5), самое низкое среди оставшихся двух и здание номер 2 самое высокое.

Формат входных данных

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

Вторая строка содержит n целых чисел s_i , которые описывают перестановку из чисел $1, 2, \dots, n$, это значит, что $1 \leq s_i \leq n$ и $s_i \neq s_j$ при $i \neq j$.

Третья строка содержит m целых чисел h_i — высоты зданий ($1 \leq h_i \leq 10^9$). Все h_i различны.

Формат выходных данных

В первую строку выведите целое число k — число соответствий. Во вторую строку выведите k целых чисел — номера домов, которые соответствуют полоске номер 1 логотипа в каждом из k соответствий. Числа во второй строке требуется выводить в возрастающем порядке. Если $k = 0$, выведите второй строкой пустую.

Пример

стандартный ввод	стандартный вывод
5 10	2
2 1 5 3 4	2 6
5 6 3 8 12 7 1 10 11 9	

Замечание

Последовательности 6, 3, 8, 12, 7 и 7, 1, 10, 11, 9 соответствуют логотипу, описанному перестановкой $(2, 1, 5, 3, 4)$. В частности, в первой последовательности здание номер 2 (высоты 3) самое низкое, здание номер 1 (высоты 6) самое низкое среди оставшихся, здание номер 5 (высоты 7) следующее по высоте, и так далее.

Задача D. Вирусы

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

Комитет По Исследованию Бинарных Вирусов обнаружил, что некоторые последовательности единиц и нулей являются кодами вирусов. Комитет изолировал набор кодов вирусов. Последовательность из единиц и нулей называется безопасной, если никакой ее подотрезок (т.е. последовательность из соседних элементов) не является кодом вируса. Сейчас цель комитета состоит в том, чтобы установить, существует ли бесконечная безопасная последовательность из единиц и нулей.

Формат входных данных

Первая строка входного файла содержит одно целое число N , равное количеству всех вирусных кодов. Каждая из следующих n строк содержит непустое слово, составленное из символов 0 и 1 — код вируса. Суммарная длина всех слов не превосходит 30000.

Формат выходных данных

Первая и единственная строка выходного файла должна содержать слово:

- ТАК — если бесконечная, безопасная последовательность из нулей и единиц существует;
- NIE — в противном случае.

Примеры

стандартный ввод	стандартный вывод
3 01 11 00000	NIE
3 011 11 0000	ТАК

Задача Е. Ахо

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

Это интерактивная задача.

Каждый раз, когда Маргарет и Альфред посещают кафе «У Дональда», они играют в странную игру про угадывание строк.

Её правила таковы:

- Первый игрок пишет строку S фиксированной длины N . Также у первого игрока есть строка T , изначально пустая. Обе строки состоят только из маленьких букв английского алфавита.
- Второй игрок не знает эти строки в течение всей игры. Однако ему разрешено спрашивать про любые две позиции (в обеих строках), правда ли, что символы в них равны. Например, вопрос может выглядеть так “Равны ли второй символ строки S и пятый символ строки T ?” Обратите внимание, что можно спрашивать про два символа одной строки.
- Игра состоит из M раундов. В начале каждого раунда первый игрок добавляет один символ в конец строки T .
- После добавления символа второй игрок может задать не более пяти вопросов. После этого он должен сказать, какое число подстрок строки T равно строки S .

Маргарет быстро заметила, что Альфред всегда преуспевает в роли второго игрока. Она подозревает наличие стратегии, позволяющей второму игроку выигрывать независимо от S и T . А вы так сможете?

Формат входных данных

При запуске ваша программа должна считывать два целых числа N и M ($1 \leq N, M \leq 20\,000$) из стандартного потока ввода.

Далее следуют M раундов игры. В i -м раунде вы можете задать **не более пяти** вопросов в формате “<позиция1> <позиция2>”. Описание любой позиции выглядит как “s x ” где $1 \leq x \leq N$ (если это x -й символ строки S) или как “t y ” где $1 \leq y \leq i$ (если это y -й символ строки T). Ответ программы жюри будет “Yes”, если символы на этих позициях равны и “No” иначе.

Формат выходных данных

В конце любого раунда вы должны вывести ответ в формате “\$ k ”, где k равно числу вхождений строки S в строку T . После этого в строку T будет автоматически добавлен новый символ (если это не последний раунд).

Не забудьте делать `flush` после каждого вопроса. После того, как вы вывели все m чисел, ваша программа должна автоматически завершиться, иначе ваш вердикт может быть каким угодно.

Пример

стандартный ввод	стандартный вывод
3 7	s 1 s 2
No	s 1 s 3
Yes	s 2 t 1
No	s 1 t 1
Yes	\$ 0
Yes	s 2 t 2
Yes	\$ 0
No	s 3 t 3
No	\$ 1
Yes	s 2 t 4
Yes	s 1 t 4
Yes	\$ 1
	s 1 t 5
	\$ 1
	s 2 t 6
	\$ 1
	s 3 t 7
	\$ 2

Замечание

В примере, строка S изначально равна “aba”, а строка T получается добавлением символов “a”, “b”, “a”, “c”, “a”, “b”, “a”.

Задача F. Divljak

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	768 мегабайт

Это история про N варваров и Тарзана. Варвары занумерованы числами от 1 до N , и у каждого на табличке написана строка из строчных букв латинского алфавита. Игра состоит из Q раундов, каждый из которых может быть одного из двух типов:

1. Тарзан показывает слово P варварам.
2. Тарзан спрашивает варвара номер i , сколько слов из показанных им до текущего момента таковы, что слово на табличке варвара является их подстрокой.

Варвары не очень любят задачи на строчки, поэтому помогите им ответить на все запросы Тарзана.

Формат входных данных

Первая строка содержит число варваров N ($1 \leq N \leq 10^5$).

Каждая из следующих N строк содержит слово на табличке очередного варвара.

Далее идет число запросов Q ($1 \leq Q \leq 10^5$).

Каждая из следующих Q строк описывает очередной запрос. Сперва записано число t — тип запроса. Если t равно 1, то далее записано слово P , которое показывает Тарзан. Иначе t равно 2 и далее записан номер варвара i ($1 \leq i \leq N$), которого спрашивает Тарзан.

Суммарная длина слов на табличках варваров $\leq 2 \cdot 10^6$.

Суммарная длина показанных Тарзаном слов $\leq 2 \cdot 10^6$.

Примеры

стандартный ввод	стандартный вывод
3 a bc abc 3 1 abca 2 1 2 3	1 1
7 abba bbaa b bbaa abba a ba 7 1 aaabbabbaab 2 7 1 baabaaa 1 aabbbab 2 3 1 aabba 2 3	1 3 4

Задача G. Словарь

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 1024 мегабайта

Словарь — это множество слов. Вы должны уметь обрабатывать запросы трех типов:

- «+ word» — добавить слово **word** в словарь, если оно в нем не присутствует.
- «- word» — удалить слово **word** из словаря, если оно там присутствует.
- «? text» — вычислить суммарное количество вхождений всех слов из словаря в текст **text**, при этом, если слово входит в текст несколько раз, то необходимо учесть каждое вхождение.

Гарантируется, что любое слово или текст являются непустыми строками, состоящими из букв **a**, **b** и **c**, суммарная длина которых не превосходит L . Однако, для упрощения задачи перед выполнением каждого запроса необходимо поступить следующим образом: пусть x обозначает ответ на последний запрос **?**, или 0, если таких запросов еще не было. Тогда необходимо очередную строку (**word** или **text**) циклически сдвинуть x раз. Напомним, что циклическим сдвигом строки $s = s_0s_1 \dots s_{|s|}$ называется строка $s' = s_1 \dots s_{|s|}s_0$.

Формат входных данных

В первой строке дано одно число Q — число запросов. В следующих Q строках находятся запросы. Суммарная длина строк во всех запросах не превосходит L ($L \leq 5\,000\,000$)

Формат выходных данных

Для каждого запроса «?» выведите одно число — ответ на него.

Пример

стандартный ввод	стандартный вывод
11	0
+ a	6
+ a	5
- a	7
- ab	
? abca	
+ ab	
+ a	
? abaaabb	
? aaabbab	
+ baa	
? babaca	

Задача Н. 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 I. Три строки

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

В новом учебном году школа, в которой обучается 7-классница Андреева Анна Андреевна, решилась на беспрецедентные меры по повышению знаний учащихся: каждый преподаватель школы должен помимо основных занятий прочитать хотя бы один годовой спецкурс. Темы спецкурсов преподаватели подобрали самые разные и интересные, а чтобы все учащиеся окончательно побросали свои внешкольные дела и с увлечением слушали новые курсы, их решено было сделать обязательными, а в конце еще и ставить дополнительную оценку в дневник.

Аня тщательно следит, чтобы в ее дневнике появлялись лишь оценки, строго большие четырех, поэтому, выкроив время между посещением бассейна и музыкальной школой, она села делать домашнее задание по спецкурсу «История магии», куда они с подружками ходят по средам, сразу после математики.

На последнем занятии рассказывалось, что все заклинания представляют собой конечные последовательности строчных латинских букв (возможно, пустые). Более того, история магии такая длинная и насыщенная, а маги в стародавние времена трудились столь плодотворно, что в наше время можно считать, что каждая конечная последовательность строчных латинских букв (в том числе и пустая) обязательно является заклинанием. После этого преподаватель рассказал, что в древности магические ордена с одной стороны беспокоились о передаче и сохранении своих знаний, а с другой ни в коем случае не желали делиться своими результатами с кем бы то ни было, вследствие чего в ордене Данных Строк применялся следующий метод шифрования. Чтобы сохранить три заклинания A , B и C , из них формировали три строки: $A' = *A*B*$, $B' = *C*A*$, $C' = *B*C*$, где вместо каждого символа $*$ подставлялась любая последовательность строчных латинских букв (возможно, пустая). Дополнительно требовалось, чтобы длины всех трех полученных строк были равны некоторому заранее выбранному числу N . Например, строки $A = \text{«a»}$, $B = \text{«tb»}$ и $C = \text{«c»}$ могли быть зашифрованы при помощи строк $A' = \text{«agtb»}$, $B' = \text{«icea»}$ и $C' = \text{«tbhc»}$.

В качестве домашнего задания преподаватель предложил каждому школьнику попытаться расшифровать по одному результату такой записи. Ане даны три строки A' , B' , C' , вероятно полученные из каких-то заклинаний A , B и C описанными выше преобразованиями.

Поскольку некоторые наборы из трех строк могли быть получены и больше чем из одного варианта исходных заклинаний A , B и C , Ане необходимо всего лишь найти максимально возможную суммарную длину исходных заклинаний.

Формат входных данных

Первая строка входного файла содержит единственное целое число N ($1 \leq N \leq 2000$) — длину каждой из входных строк. В следующих трех строках содержатся строки A' , B' , C' , каждая из которых состоит из N строчных латинских букв.

Формат выходных данных

Выведите единственное целое число — ответ на поставленную задачу.

Примеры

стандартный ввод	стандартный вывод
2 ac ba cb	3
4 agtb icea tbhc	4
3 abc cde dea	2

Задача J. Тренажёр «10₂-пальцевый набор»

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4.5 секунд
Ограничение по памяти:	512 мегабайт

Современный робот-программист должен владеть слепым 10₂-пальцевым методом набора бинарных строк, состоящих из символов «0» и «1». В инновационном тренажёре, созданном специально для уже освоивших 10₂-пальцевый набор роботов, предлагается следующее упражнение. В верхней части экрана выводится строка, состоящая из нулей и единиц. Ниже выводятся пары (c_i, w_i) , каждая из которых состоит из бинарного слова w_i и его *стоимости* c_i — количества штрафных баллов, начисляемых за каждое использование слова w_i при наборе строки.

Робот должен набрать заданную строку в виде последовательности записанных подряд префиксов или суффиксов предложенных ему бинарных слов. Одно и то же слово можно использовать произвольное количество раз, но за каждое использование префикса или суффикса начисляются штрафные баллы, равные стоимости этого слова.

Префиксом слова называется последовательность подряд идущих символов этого слова, начинающаяся с первого символа слова, а *суффиксом* — последовательность подряд идущих символов этого слова, заканчивающаяся последним символом слова. Слово целиком является как своим префиксом, так и своим суффиксом.

Требуется написать программу, которая вычисляет минимально возможное суммарное количество штрафных баллов, начисляемых роботу за набор заданной строки с использованием префиксов и суффиксов предложенных бинарных слов, или определяет, что строку набрать невозможно.

Формат входных данных

Первая строка входных данных содержит три целых числа m , n и L — длину заданной строки, количество слов, префиксы и суффиксы которых можно использовать при её наборе, и суммарную длину этих слов ($1 \leq m \leq 300\,000$; $1 \leq n \leq 300\,000$; $1 \leq L \leq 300\,000$).

Во второй строке находится заданная строка, состоящая из m символов «0» или «1». Следующие n строк описывают предлагаемые для использования бинарные слова. Сначала указывается целая стоимость слова в штрафных баллах c_i ($1 \leq c_i \leq 10^9$). Затем в той же строке через пробел следует непустое слово, состоящее из символов «0» или «1». Длина каждого слова не превосходит значения l_{max} , дополнительные ограничения на которое накладываются в некоторых подзадачах.

Формат выходных данных

Выходные данные должны содержать одно целое число — минимальное количество штрафных баллов, которое потребуется, чтобы набрать заданную строку, или число -1 , если требуемым образом набрать её невозможно.

Примеры

стандартный ввод	стандартный вывод
3 1 3 100 1 101	-1
9 2 8 000110100 1 100 1 11001	4