

Задачка А. Задачка на строчечки

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

В стандартном потоке ввода или файле
е
кой. На следующей строчке программч
ь а i
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 i
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 д е с я стю 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 њ 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 i 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 њ л o x e n и т с e в ы в а н ж л o d ж д к
к н у а
, e e ч и ч o т o п њ н т р a d н a t c a н и т s e v ы в а ю
к ж щ н
о х e т з и у к ч e v o l s o p - х a k ч e ч o р t c N х и а
в њ
о к у б х и k c н и т a л х и к њ н e л a m з и у k ч e ч o р t c т њ d

Форматик 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 $1\,000\,000$.

Форматик 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 $1\,000\,000$.

Примерчик

<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>	

Задачка В. Префикс-функция

Имя входного файла: `inputik.txt`
Имя выходного файла: `outputik.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Дана непустая строка S , длина которой N не превышает 10^6 . Будем считать, что элементы строки нумеруются от 1 до N .

Требуется для всех i от 1 до N вычислить её префикс-функцию $\pi[i]$.

Формат входных данных

Одна строка длины N , $0 < N \leq 10^6$, состоящая из маленьких латинских букв.

Формат выходных данных

Выведите N чисел — значения префикс-функции для каждой позиции, разделённые пробелом.

Примерчик

<code>inputik.txt</code>	<code>outputik.txt</code>
<code>abrachkacadabrachka</code>	<code>0 0 0 1 0 0 0 1 0 1 0 1 2 3 4 5 6 7 8</code>

Задачка С. Ахо

Имечко входного файла:	стандартный вводик
Имечко выходного файла:	стандартный выводик
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайтов

Это интерактивная задачка.

Каждый разок, когда мальчик Ибрагимчик зовёт мальчика Степанчика заниматься взрослыми неприличными делишками, они перед этим посещают кафешечку «У Дональдика» и играют в странную игрушку про угадывание строчечек.

Её правила таковы:

- Степанчик пишет строчечку S фиксированной длины N . Также у Степанчика есть строчечка T , изначально пустенькая. Обе строчечки состоят только из маленьких букв английского алфавита.
- Ибрагимчик не знает эти строчечки в течение всей игрушки. Однако Ибрагимчику разрешено спрашивать про любые две позиции (в обеих строчечках), правда ли, что символы в них одинаковые. Например, вопросик может выглядеть так “Равны ли второй символ строчечки S и пятый символ строчечки T ?” Обратите внимание, что можно спрашивать про два символа одной строчечки.
- Игра состоит из M раундов. В начале каждого раунда Степанчик добавляет один символ в конец строчечки T .
- После добавления символа Ибрагимчик может задать не более пяти вопросиков. После этого он должен сказать, какое число подстрочечек строчечки T равно строчечке S .

Степанчик быстро заметил, что Ибрагимчик всегда преуспевает в роли второго игрока. Он подозревает наличие стратегии, позволяющей Ибрагимчику выигрывать независимо от S и T . А вам так слабо?

Формат входных данных

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

Далее следуют M раундов игрушки. В i -м раунде вы можете задать не более пяти вопросиков в формате “<позиция1> <позиция2>”. Описание любой позиции выглядит как “s x ” где $1 \leq x \leq N$ (если это x -й символ строчечки S) или как “t y ” где $1 \leq y \leq i$ (если это y -й символ строчечки T). Ответ программки напечет преподавачек 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 строчечки S в строчечку T . После этого в строчечку T будет автоматически добавлен новый символ (если это не последний раунд).

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

Примерчик

стандартный вводик	стандартный выводик
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
Yes	s 3 t 3
No	\$ 1
No	s 2 t 4
Yes	s 1 t 4
Yes	\$ 1
Yes	s 1 t 5
Yes	\$ 1
Yes	s 2 t 6
Yes	\$ 1
Yes	s 3 t 7
Yes	\$ 2

Замечанье

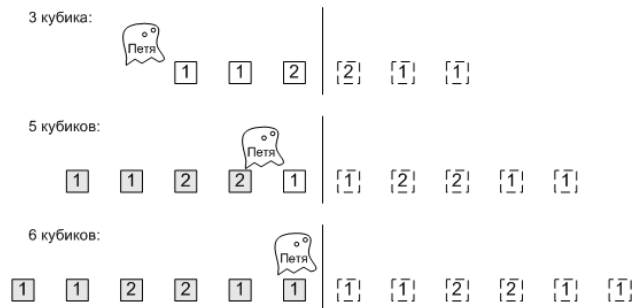
В примере, строчка S изначально равна "aba", а строчка T получается добавлением символов "a", "b", "a", "c", "a", "b", "a".

Задачка D. Кубики

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

Привидение Петечка любит играть со своими кубиками. Он любит выкладывать их в рядочек и разглядывать своё творение. Однако недавно друзьяшки решили подшутить над Петечкой и поставили в его игровой комнатке зеркальце. Ведь всем известно, что привидения не отражаются в зеркалах! А кубики отражаются.

Теперь Петечка видит перед собой N цветных кубиков, но не знает, какие из этих кубиков настоящие, а какие — всего лишь отражения в зеркальце.



Помогите Петечке! Выясните, сколько у него может быть кубиков. Петечка видит отражение всех кубиков в зеркальце и частичку кубиков, которая находится перед ним. Частичка кубиков может быть позади Петечки, их он не видит.

Формат входных данных

Первая строчечка входного файла содержит два целых числа: N ($1 \leq N \leq 100\,000$) и колличество (сколько повторений) различных цветиков, в которые могут быть раскрашены кубики, — M ($1 \leq M \leq 100\,000$).

Следующая строчечка содержит N целых чисел от 1 до M — цветики кубиков.

Формат выходных данных

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

Примерчик

inputik.txt	outputik.txt
6 2	3 5 6
1 1 2 2 1 1	

Задачка Е. Сравнение подстрочечек

Имя входного файла: `inputik.txt`
Имя выходного файла: `outputik.txt`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Дана строчечка $S = s_1 s_2 \dots s_n$ и множество запросов (l_1, r_1, l_2, r_2) . Для каждого запроса требуется ответить, какая из подстрочечек больше — $s_{l_1} \dots s_{r_1}$ или $s_{l_2} \dots s_{r_2}$.

Формат входных данных

В первой строчечке записана строчечка S , состоящая из строчных латинских букв. Эта строчечка непустая и имеет длину не более 100 000 символов.

Во второй строчечке записано число q ($1 \leq q \leq 100\,000$) — количество запросов.

В каждой из следующих q строчечек записаны числа l_1, r_1, l_2, r_2 ($1 \leq l_1 \leq r_1 \leq |S|$; $1 \leq l_2 \leq r_2 \leq |S|$).

Формат выходных данных

Для каждого запроса выведите «=», если соответствующие подстрочечки одинаковые, «>», если первая подстрочечка больше и «<», если первая подстрочечка меньше.

Примеры

inputik.txt	outputik.txt
abacaba	=
3	<
1 3 5 7	>
1 3 3 5	
4 7 2 5	
ab	<
2	<
1 1 2 2	
1 1 1 2	

Задачка F. Принтерчик

Имя входного файла:	inputik.txt
Имя выходного файла:	outputik.txt
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Вам нужно напечатать N словечек на *Movable Typik Printerchik*. *Movable Typik Printerchiki* — это старенькие принтерчики, для работы которых требуется ставить малюсенькие металлические кусочечки (каждый из кусочечков содержит одну букву) в определенном порядке, образуя таким образом словечки. Потом все они вдавливаются в листочек бумажечки. Таким образом печатается одно словечко. Ваш принтерчик позволяет делать следующие операции:

- Добавить букву в конец словечка, находящегося сейчас на принтерчике.
- Удалить последнюю букву из словечка, находящегося сейчас на принтерчике. Эту операцию можно делать, только если словечко содержит хотя бы одну букву.
- Напечатать словечко, находящееся на принтерчике (при этом словечко никуда не исчезает, можно печатать его ещё разок и ещё разок).

Изначально на принтерчике содержится пустое словечко. В конце печати на принтерчике можно оставить непустое словечко. Словечки, которые вам даны, вы можете печатать в произвольном порядке.

Каждая из трёх операций занимает одну единицу времени. Вам нужно найти последовательность операций, которая печатает данные N словечек за минимальное время. Если минимальных последовательностей несколько, выведите любую.

Формат входных данных

В первой строчке содержится число N ($1 \leq N \leq 25\,000$).

В следующих N строчках содержатся словечки (по одному на строчке), состоящие из маленьких букв латинского алфавита. Длина каждого словечка не превышает 20. Все словечки различны.

Формат выходных данных

Ваша программка должна вывести следующие данные:

- На первой строчке число M — число операций.
- На следующих M строчках по одному символу — описание операций. Каждая операция описывается одним символом:
 - Добавление символа обозначается собственно символом, который добавляется.
 - Удаление символа обозначается символом «-» (минусик, ASCII-код 45).
 - Операция «напечатать текущее словечко» обозначается символом «P» (заглавная латинская буква P).

Примерчик

inputik.txt	outputik.txt
3 print the poem	20 t h e P - - - p o e m P - - - r i n t P

Задачка G. Склеечка стречек

Имечко входного файла: `inputik.txt`
Имечко выходного файла: `outputik.txt`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайтов

Вам дан наборчик словечек. Найдите наиболее коротенькую строчечку, которая содержит все строчечки наборчика как подстрочечки. Если существует несколько возможных ответиков, выведите первенький в лексикографическом порядочке.

Форматик входных данных

В первой строчечке содержится чиселко n ($1 \leq n \leq 12$) — количюсик (сколько штукечек) строчечек в наборчике.

Следующие n строчечек являются строчечками наборчика. Няшечки-преподавашечки гарантируют, что они состоят только из прописных латинских буквочек, они не пустенькие, и что их длинина не превосходит 50.

Форматик выходных данных

Выведите одну строчечку, являющуюся ответиком на задачку.

Примерчики

inputik.txt	outputik.txt
2 BAB ABA	ABAB
4 ABABA AKAKA AKABAS ABAKA	ABABAKAKABAS
6 AAA BBB CCC ABC BCA CAB	AAABBB CABCCC
8 OFG SDOFGJTILM KBWNF YAAPO AWX VSEAWX DOFGJTIL YAA	KBWNFSDOFGJTILMVSEAWXYAAPO
2 STRING RING	STRING

Задачка Н. Бинарненькая игрушка

Имечко входного файла: `inputik.txt`
Имечко выходного файла: `outputik.txt`
Ограниченье по времени: 2 секундошки
Ограниченье по памяти: 512 мегабайтиков

Искандерчик и Олечка любят придумывать ребусики. Но больше, чем придумывать ребусики, они любят придумывать какие-нибудь игрушечки на строчечках. Вот и сейчас им в головки пришла забавненькая игрушечка со следующими правилами:

- Выбирается какой-то наборчик *запрещённых* двоичных (состоящих из нолеков и единичек) строчечек f_1, f_2, \dots, f_n .
- Выбирается некоторая стартовая бинарненькая строчечка s , такая что ни одна из запрещённых строчечек не входит в неё как подстрочечка.
- Игрок по очереди дописывают в конец строчечки s по одному символу «0» или «1». Олечка ходит первенькой.
- Проигрывает тот, после чьего хода хотя бы одна из запрещённых строчечек f_1, f_2, \dots, f_n входит в s как подстрочечка.
- В случае если при оптимальной игрушечке обоих игроков игрушечка может продолжаться сколь угодно долго, то объявляется ничья.

Вы боитесь портить другим людям их любимые развлечения, поэтому решили написать программку, которая будет определять исход игры по заданному набору запрещённых строчечек и стартовой строчечке s .

Формат входных данных

В первой строчке входных данных записаны два целых числа n и m ($0 \leq n \leq 100\,000$, $0 \leq m \leq 1\,000\,000$) — количество (сколько штук) запрещённых строчечек и начальная длина строчечки s .

В каждой из последующих n строчек содержится одна запрещённая строчечка. Няшки-преподаватели гарантируют, что все эти строчечки непусты, состоят из символов «0» и «1» и никакая из них не является подстрочечкой строчечки s . Дополнительно няшки-преподаватели гарантируют, что **суммарная длина** всех запрещённых строчечек не превосходит 1 000 000.

В последней строчке входных данных записана стартовая строчечка s длины m , состоящая только из символов «0» и «1». Обратите внимание, строчечка s может быть пустой, в этом случае соответствующая строчечка 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 игрушечке обоих игроков выведите:

- «*Olyechka*» (без кавычек), если Олечка может победить вне зависимости от того как будет играть Искандерчик. Напомним, что Олечка ходит первенькой.
- «*Iskanderchik*» (без кавычек), если Искандерчик может победить независимо от ходов Олечки.
- «*Nichyashchka*» (без кавычек), если при оптимальной игрушечке обоих игроков игрушечка будет продолжаться бесконечно долго.

Примерчики

inputik.txt	outputik.txt
1 0 1	Nichyashechka
3 1 000 001 011 0	Olyechka
2 3 1001 000 100	Iskanderchik

Задачка I. Сложная задача

Имечко входного файла: стандартный ввод
Имечко выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Задача настолько сложная, что уменьшительно-ласкательных суффиксов тут не предусмотрено. *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 $s' = s_1 \dots s_{|s|}s_0$.

Форматик 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 ($L \leq 5\,000\,000$)

Форматик выходных данных

Эта необычная задача. В ней всего 2 теста, причём ответ не обязательно должен совпадать с ответом на поставленную выше задачу. Первый тест совпадает с тестом из условия и ответ на него должен быть таким же, как в примере. Про второй тест известно только то, что ответ в нём — это одна строчная буква латинского алфавита. Какая именно — не сообщается. Рекомендуется написать скрипт, посылающий ваше решение в `ejudge`.

Примерчик

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