

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

D. Филин Серафим

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Ребята выстроились в очередь из n человек, начинающуюся с человека под номером $i = 1$, чтобы узнать у филина Серафима, в чем смысл жизни. К сожалению, Кирилл был очень занят написанием легенды к этой задаче, поэтому пришел чуть позже и встал в конец очереди после n -го человека. Кирилла такое положение совершенно не устраивает, поэтому он решил подкупить некоторых людей впереди себя.

Для i -го человека в очереди Кирилл знает два значения: a_i и b_i . Если в данный момент Кирилл стоит на позиции i , то он может выбрать любую позицию j такую, что $j < i$ и поменяться с человеком на позиции j местами. При этом Кириллу придется заплатить ему a_j монет. А для каждого k такого, что $j < k < i$ Кириллу придется заплатить b_k монет человеку на позиции k . Кирилл может выполнить это действие любое количество раз.

Кирилл человек бережливый, поэтому он хочет потратить как можно меньше монет, но он не хочет ждать слишком долго, поэтому Кирилл считает должен оказаться среди первых m человек в очереди.

Помогите Кириллу определить минимальное количество монет, которое ему придется потратить, чтобы не ждать слишком долго.

Входные данные

Каждый тест состоит из нескольких наборов входных данных. В первой строке находится одно целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора содержит два целых числа n и m ($1 \leq m \leq n \leq 200\,000$) — количество людей в очереди помимо Кирилла и максимальная допустимая финальная позиция Кирилла соответственно.

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

Третья строка содержит n целых чисел b_1, b_2, \dots, b_n ($1 \leq b_i \leq 10^9$).

Гарантируется, что сумма значений n по всем наборам входных данных не превосходит $2 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите единственное целое число — минимальное количество монет, которое нужно потратить Кириллу.

Пример

входные данные	Скопировать
4 4 2 7 3 6 9 4 3 8 5 6 2 6 9 7 1 8 3 5 8 8 1 4 1 7 7 7 2 9 2 6 5 9 9 1 10 7 1 4 9 2 1 2 3 1 1	
выходные данные	Скопировать
14 22 9 3	

С. Чередуи!

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Дана двоичная строка s . Двоичная строка — это строка, состоящая из символов 0 и/или 1.

Вы можете производить следующую операцию со строкой s любое количество раз (даже ноль):

- выбрать целое число i такое, что $1 \leq i \leq |s|$, затем удалить символ s_i .

Вам нужно сделать строку s чередующейся, т. е. после выполнения операций каждые два соседних символа в s должны быть разными.

Ваша задача — вычислить два значения:

- минимальное количество операций, необходимых для того, чтобы сделать строку s чередующейся;
- количество различных кратчайших последовательностей операций, которые делают строку s чередующейся. Две последовательности операций считаются различными, если хотя бы в одной операции выбранное целое число i отличается в этих двух последовательностях.

Входные данные

Первая строка содержит одно целое число t ($1 \leq t \leq 10^4$) — количество наборов входных данных.

Каждый набор входных данных состоит из одной строки, содержащей строку s ($1 \leq |s| \leq 2 \cdot 10^5$). Строка s состоит только из символов 0 и/или 1.

Дополнительное ограничение на входные данные:

- суммарная длина строк s по всем наборам входных данных не превышает $2 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите два целых числа: минимальное количество операций, которые вам нужно выполнить, и количество различных кратчайших последовательностей операций. Поскольку второе число может быть большим, выведите его остаток по модулю 998244353.

Пример

входные данные	Скопировать
3 10010 111 0101	
выходные данные	Скопировать
1 2 2 6 0 1	

Примечание

В первом наборе входных данных примера кратчайшими последовательностями операций являются:

- [2] (удалить 2-й символ);
- [3] (удалить 3-й символ).

Во втором наборе входных данных примера кратчайшими последовательностями операций являются:

- [2, 1] (удалить 2-й символ, затем удалить 1-й символ);
- [2, 2];
- [1, 1];
- [1, 2];
- [3, 1];
- [3, 2].

В третьем наборе входных данных примера единственной кратчайшей последовательностью операций является [] (пустая последовательность).

С. Лед и пламень

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Little09 и его друзья играют в игру. Есть n игроков, и значение температуры игрока i равно i .

Типы окружающей среды обозначаются как 0 или 1. Когда два игрока сражаются в определенной среде, если её тип равен 0, то всегда побеждает игрок с более низким значением температуры; если он равен 1, то всегда побеждает игрок с более высоким значением температуры. Типы этих сред $n - 1$ образуют бинарную строку s длиной $n - 1$.

Если в игре участвует x игроков, то всего будет $x - 1$ сражений, а типами $x - 1$ сред будут первые $x - 1$ символов s . Если в турнире осталось более одного игрока, выберите для сражения любых двух оставшихся игроков. Игрок, который проигрывает, выбывает из турнира. Типом среды для i -го сражения является s_i .

Для каждого x от 2 до n ответьте на следующий вопрос: если в игре участвуют все игроки, значение температуры которых не превышает x , то сколько игроков имеют шансы на победу?

Входные данные

Каждый тест содержит несколько наборов входных данных. Первая строка содержит одно целое число t ($1 \leq t \leq 10^3$) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит одно целое число n ($2 \leq n \leq 2 \cdot 10^5$) — количество игроков.

Вторая строка каждого набора входных данных содержит бинарную строку s длиной $n - 1$.

Гарантируется, что сумма n по всем наборам входных данных не превышает $3 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите $n - 1$ целых чисел — для каждого x от 2 до n выведите количество игроков, имеющих шанс на победу.

Пример

входные данные	Скопировать
2 4 001 4 101	
выходные данные	Скопировать
1 1 3 1 2 3	

Примечание

В первом наборе входных данных для $x = 2$ и $x = 3$ победителем может стать только игрок, чье значение температуры равно 1. Для $x = 4$ победителем могут стать игроки, чьи значения температуры равны 2, 3, 4.

ЗАДАЧИ ОТОСЛАТЬ СТАТУС ПОЛОЖЕНИЕ ЗАПУСК

С. Гамильтонова стена

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Сэр Монокарп Гамильтон планирует покрасить свою стену. Стену можно представить как сетку, состоящую из 2 строк и m столбцов. Изначально стена полностью белая.

Монокарп хочет нарисовать на стене черное изображение. В частности он хочет, чтобы клетка (i, j) (j -я клетка в i -й строке) была покрашена в черный, если $c_{i,j} = 'B'$, и была оставлена белой, если $c_{i,j} = 'W'$. Кроме того он хочет, чтобы в каждом столбце была хотя бы одна черная клетка, поэтому для каждого j выполняется следующее условие: $c_{1,j}, c_{2,j}$ или они обе равны 'B'.

Чтобы картинка получилась ровной, Монокарп хочет поставить кисть в какую-нибудь клетку (x_1, y_1) и провести ей по пути $(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)$ так, чтобы:

- для каждого i (x_i, y_i) и (x_{i+1}, y_{i+1}) имели общую сторону;
- все чёрные клетки встречались в пути **ровно по одному разу**;
- в пути не встречались белые клетки.

Определите, может ли Монокарп покрасить стену.

Входные данные

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

В первой строке каждого набора входных данных записано одно целое число m ($1 \leq m \leq 2 \cdot 10^5$) — количество столбцов в стене.

В i -й из следующих двух строк записана строка c_i , состоящая из m символов, где каждый символ — это 'B' или 'W'. $c_{i,j}$ равно 'B', если клетка (i, j) должна быть покрашена черным, и 'W', если клетка (i, j) должна быть оставлена белой.

Кроме того, для каждого j выполняется следующее условие: $c_{1,j}, c_{2,j}$ или они обе равны 'B'.

Сумма m по всем наборам входных данных не превосходит $2 \cdot 10^5$.

Выходные данные

На каждый набор входных данных выведите «YES», если Монокарп может покрасить стену. В противном случае выведите «NO».

Пример

входные данные	Скопировать
6 3 WBB BBW 1 B B 5 BWBWB BBBBB 2 BW WB 5 BBBBW BWBWB 6 BWBWB BBBBB	
выходные данные	Скопировать
YES YES NO NO NO YES	

Примечание

В первом наборе входных данных Монокарп может провести кистью по пути $(2, 1)$, $(2, 2)$, $(1, 2)$, $(1, 3)$. Все черные клетки встречаются в пути ровно один раз, никакая белая клетка не встречается в пути.

Во втором наборе входных данных Монокарп может воспользоваться путем $(1, 1)$, $(2, 1)$.

В третьем наборе входных данных:

- путь $(1, 1)$, $(2, 1)$, $(2, 2)$, $(2, 3)$, $(1, 3)$, $(2, 4)$, $(2, 5)$, $(1, 5)$ не подходит, потому что пара клеток $(1, 3)$ и $(2, 4)$ не имеют общую сторону;
- путь $(1, 1)$, $(2, 1)$, $(2, 2)$, $(2, 3)$, $(1, 3)$, $(2, 3)$, $(2, 4)$, $(2, 5)$, $(1, 5)$ не подходит, потому что клетка $(2, 3)$ посещается дважды;
- путь $(1, 1)$, $(2, 1)$, $(2, 2)$, $(2, 3)$, $(2, 4)$, $(2, 5)$, $(1, 5)$ не подходит, потому что черная клетка $(1, 3)$ не встречается в пути;
- путь $(1, 1)$, $(2, 1)$, $(2, 2)$, $(2, 3)$, $(2, 4)$, $(2, 5)$, $(1, 5)$, $(1, 4)$, $(1, 3)$ не подходит, потому что белая клетка $(1, 4)$ встречается в пути.

[Codeforces](#) (c) Copyright 2010-2024 Михаил Мирзаянов
Соревнования по программированию 2.0
Время на сервере: 10.11.2024 17:56:56^{UTC+5} (11).
Мобильная версия, переключиться на [десктопную](#).
[Privacy Policy](#)

При поддержке



ИТМО

С. Получи четную строку

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Строка $a = a_1 a_2 \dots a_n$ называется *чётной*, если она состоит из конкатенации (соединения) строк длины 2, состоящих из одинаковых символов. Иными словами, строка a четная, если **одновременно** выполняются два условия:

- чётна её длина n ;
- для всех нечётных i ($1 \leq i \leq n - 1$) выполнено $a_i = a_{i+1}$.

Например, следующие строки являются чётными: «» (пустая строка), «tt», «aabb», «oooo» и «ttrrrrooooooooooukk». Следующие строки чётными не являются: «aaa», «abab» и «abba».

Задана строка s , состоящая из строчных латинских букв. Необходимо найти, какое минимальное количество символов нужно удалить из строки s , чтобы она стала четной. Удалённые символы не обязаны идти подряд.

Входные данные

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

Далее следуют описания наборов входных данных.

Каждый набор входных данных состоит из одной строки s ($1 \leq |s| \leq 2 \cdot 10^5$), где $|s|$ — длина строки s . Строка состоит из строчных букв латинского алфавита.

Гарантируется, что сумма $|s|$ по всем наборам входных данных не превышает $2 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите единственное число — минимальное количество символов, которые необходимо удалить, чтобы строка s стала четной.

Пример

входные данные	Скопировать
6 aabbda b ddcc zu x aaababbb aabbcc oaoaaaao bme f bmuw	
выходные данные	Скопировать
3 3 2 0 2 7	

Примечание

В первом наборе входных данных можно удалить символы под номерами 6, 7 и 9, получив четную строку «aabbddcc».

Во втором наборе входных данных каждый символ встречается ровно один раз, поэтому для того, чтобы получить четную строку, необходимо удалить все символы из строки.

В третьем наборе входных данных можно получить четную строку «aaaabb», удалив, например, 4-й и 6-й символы, или строку «aabbbb», удалив 5-й символ и любой из первых трех.

С. Псиная поэзия

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

После трагической смерти его жены, Эвридики, Орфей решил спуститься в царство мертвых, чтобы увидеть её. Достичь врат царства было непросто, однако пройти через них будет еще сложнее. В основном, из-за Цербера — трёхглавого пса Аида.

Орфей, известный поэт и музыкант, планирует усыпить Цербера своей поэзией и спокойно пройти мимо него. Он создал очень своеобразную поэму специально для Цербера. Она состоит только из строчных английских букв.

Подстрока поэмы является палиндромом, если она читается одинаково слева направо и справа налево. Строка a является подстрокой строки b , если a может быть получена из b удалением нескольких (возможно, нуля или всех) символов из начала и нескольких (возможно, нуля или всех) символов из конца.

К сожалению, Цербер не любит палиндромы длины больше 1. Например, в поэме `abaa` псу Аида не понравились бы подстроки `aba` и `aa`.

Орфей сможет успокоить Цербера только если псу понравится его поэзия. Поэтому, он хочет изменить свою поэму так, чтобы она не содержала ни одной подстроки палиндрома с длиной **строго больше 1**.

Орфей может изменять поэму, заменяя букву на любой позиции любой строчной английской буквой. Он может применять эту операцию произвольное количество раз (возможно, ноль). Так как в его поэме могут быть палиндромы, ему придется сделать несколько изменений. Но сколько именно? Вам дана поэма, определите минимальное количество букв, которые нужно изменить, чтобы поэма перестала содержать подстроки палиндромы длины больше 1.

Входные данные

Во входных данных находятся несколько (не меньше одного) наборов входных данных. В первой строке дано одно целое число t ($1 \leq t \leq 10^5$), обозначающее количество наборов входных данных. Затем даны t наборов входных данных.

Единственная строка каждого набора входных данных содержит непустую строку, состоящую из строчных английских букв — поэму Орфея.

Сумма длин поэм во всех наборах входных данных не превышает 10^5 .

Выходные данные

Вы должны вывести t строк, i -я из них должна содержать одно целое число — ответ на i -й набор входных данных.

Пример

<div>входные данные</div> <div>7 babba abaac codeforces zeroorez abdcdba bbbbbbb a</div>	<div>Скопировать</div>
<div>выходные данные</div> <div>1 1 0 1 1 4 0</div>	<div>Скопировать</div>

Примечание

В первом наборе входных данных можно заменить третий символ на `c` и получить поэму `bacba`, не содержащую палиндромных подстрок.

Во втором наборе входных данных можно заменить третий символ на `d` и получить поэму `abdac`, не содержащую палиндромных подстрок.

В третьем наборе входных данных исходная поэма уже не содержит палиндромных подстрок, поэтому Орфею не нужно больше ничего делать.

https://codeforces.com/problemset/problem/1466/C

1/2

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

В. Ходы и повороты

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 512 мегабайт

В начале координат бесконечной двумерной плоскости находится робот. Каждую секунду робот перещается ровно на 1 метр в одном из четырёх направлений: на север, юг, запад или восток. Робот **может выбрать произвольное направление** для первого шага, но после этого в конце каждой секунды он **обязан повернуть** на 90 градусов влево или вправо относительно направления, в котором только что двигался. Например, если робот только что переместился на север или юг, его следующий шаг должен быть на запад или восток, и наоборот.

Робот совершает **ровно n** шагов из начального положения в соответствии с правилами выше. В скольких различных точках робот может оказаться в итоге? Финальное направление робота можно игнорировать.

Входные данные

В единственной строке записано одно целое число n ($1 \leq n \leq 1000$) — количество шагов робота.

Выходные данные

Выведите одно целое число — количество различных возможных положений робота после **ровно n** шагов.

Примеры

входные данные	Скопировать
1	
выходные данные	Скопировать
4	

входные данные	Скопировать
2	
выходные данные	Скопировать
4	

входные данные	Скопировать
3	
выходные данные	Скопировать
12	

Примечание

В первом примере робот переместится на 1 метр на север, юг, запад или восток в зависимости от исходного направления.

Во втором примере робот всегда окажется в $\sqrt{2}$ метрах на северо-западе, северо-востоке, юго-западе или юго-востоке.



[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

C1. Армия покемонов (простая версия)

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

Это простая версия задачи. Различия между версиями заключаются в том, что в простой версии нет запросов обмена. Вы можете делать взломы, только если обе версии задачи сданы.

Пикачу — милый и дружелюбный покемон, живущий в стае диких пикачу.

Однако недавно стало известно, что команда R хочет украсть всех этих покемонов! Тренер покемонов Андрей решил помочь Пикачу собрать армию для борьбы с командой R.

В первую очередь Андрей посчитал всех покемонов: их оказалось ровно n штук. Затем он установил силу каждого покемона, и так получилось, что i -й покемон имеет силу, равную a_i , и силы всех покемонов различны.

В качестве армии Андрей может выбрать любую непустую подпоследовательность покемонов. Иными словами, Андрей выбирает какой-то массив b из k индексов таких, что $1 \leq b_1 < b_2 < \dots < b_k \leq n$, и его армия будет состоять из покемонов с силами $a_{b_1}, a_{b_2}, \dots, a_{b_k}$.

Сила армии вычисляется как знакопеременная сумма элементов подпоследовательности, то есть $a_{b_1} - a_{b_2} + a_{b_3} - a_{b_4} + \dots$.

Андрей экспериментирует с построением покемонов. Он q раз меняет двух покемонов местами, а именно, в i -й раз он менял местами покемонов с номерами l_i и r_i .

Обратите внимание, в этой версии задачи $q = 0$.

Андрею надо знать: какую максимальную силу армии он мог получить при начальной расстановке покемонов, а также после каждого изменения строя?

Помогите Андрею и покемонам, иначе команде R удастся воплотить в жизнь свой коварный план!

Входные данные

Каждый тест содержит несколько наборов входных данных.

В первой строке находится одно целое положительное число t ($1 \leq t \leq 10^3$) — количество наборов входных данных. Описание наборов входных данных приведено ниже.

В первой строке каждого набора входных данных находятся два целых числа n и q ($1 \leq n \leq 3 \cdot 10^5, q = 0$) — количество покемонов и количество обменов соответственно.

Во второй строке находятся n различных целых положительных чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) — силы покемонов.

i -я из следующих q строк содержит два целых положительных числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$) — номера обмениваемых покемонов в i -й операции.

Гарантируется, что сумма n по всем наборам входных данных не превосходит $3 \cdot 10^5$, а также сумма q по всем тестовым случаям не превосходит $3 \cdot 10^5$.

Выходные данные

Для каждого набора входных данных выведите $q + 1$ число — максимально возможную силу армии до изменений и после каждого изменения.

Пример

входные данные	Скопировать
<pre>3 3 0 1 3 2 2 0 1 2 7 0 1 2 5 4 3 6 7</pre>	
выходные данные	Скопировать
<pre>3 2 9</pre>	

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

B. WOW фактор

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Строка a является подпоследовательностью строки b , если a может быть получена из b удалением нескольких (возможно, ни одного или всех) символов. Например, для строки $a = \text{«wowwo»}$, следующие строки являются подпоследовательностями: «wowwo» , «wowo» , «oo» , «wow» , «» и другие, а следующие не являются подпоследовательностями: «owoo» , «owwwwo» , «ooo» .

Wow фактор строки равен количеству подпоследовательностей, которые равны строке «wow» . Боб хотел выписать строку, которая имеет большой *wow фактор*. Однако кнопка «w» на его клавиатуре сломана, поэтому он пишет вместо нее две буквы «V» .

Боб неожиданно осознал, что, возможно, в строке можно найти больше букв «w» , чем он написал. Например, рассмотрим строку «ww» . Боб написал бы ее как «vvvv» , но в этой строке можно найти три буквы «w» :

- «vvvv»
- «vvvv»
- «vvvv»

Например, *wow фактор* в слове «vvovvv» равен четырем, потому что есть четыре способа выбрать «wow» :

- «vvovvv»
- «vvovvv»
- «vvovvv»
- «vvovvv»

Обратите внимание, что подпоследовательность «vvovvv» не считается, так как буквы «V» должны быть последовательными.

Посчитайте *wow фактор* для строки s и выведите его. Обратите внимание, что **не** гарантируется, что возможно получить строку s из другой строки, заменяя «w» на «vv» . Например, s может быть «vov» .

Входные данные

Первая строка содержит одну непустую строку s , которая состоит только из символов «v» и «o» . Длина строки s не более 10^6 символов.

Выходные данные

Выведите одно целое число — *wow фактор* строки s .

Примеры

входные данные	Скопировать
vvvovvv	
выходные данные	Скопировать
4	

входные данные	Скопировать
vvovooovovvovvovvovvovvovvov	
выходные данные	Скопировать
100	

Примечание

Первый пример объяснен в легенде.

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

С. Добиться равенства

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Вам даны две бинарные строки a и b равной длины. Вы можете производить над строкой a следующие операции:

- Поменять два бита на позициях i и j соответственно ($1 \leq i, j \leq n$), стоимость этой операции составляет $|i - j|$, то есть модуль разницы i и j .
- Выбрать произвольную позицию i ($1 \leq i \leq n$) и инвертировать (заменить 0 на 1 или 1 на 0) значение бита на этой позиции. Стоимость этой операции равна 1

Найдите минимальную стоимость, которую нужно потратить, чтобы сделать строку a равной строке b . Менять строку b не разрешается.

Входные данные

Первая строка входных данных содержит одно целое число n ($1 \leq n \leq 10^6$) — длина строк a и b .

Вторая и третья строки содержат строки a и b соответственно.

Обе строки a и b имеют длину n и состоят только из «0» и «1».

Выходные данные

Выведите минимальную стоимость, которая требуется, чтобы превратить строку a в b .

Примеры

входные данные	Скопировать
3 100 001	
выходные данные	Скопировать
2	

входные данные	Скопировать
4 0101 0011	
выходные данные	Скопировать
1	

Примечание

В первом примере одним из оптимальных способов является следующий: нужно инвертировать позицию 1, а затем позицию 3. Тогда строка a меняется следующим образом: «100» → «000» → «001». Стоимость составляет $1 + 1 = 2$.

Другим оптимальным решением является следующее: можно поменять местами позиции 1 и 3, тогда строка a меняется как «100» → «001», стоимость также равна $|1 - 3| = 2$.

Во втором примере оптимальным решением является следующее: нужно поменять местами позиции 2 и 3, тогда строка a меняется как «0101» → «0011». Стоимость равна $|2 - 3| = 1$.

С. Простые строки

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

zscoder любит простые строки! Строка t называется простой, если любая пара соседних букв различается. Например, строки `ab`, `aba`, `zscoder` — простые, в то время как строки `aa`, `add` не являются таковыми.

У zscoder есть строка s . Он хочет изменить в ней наименьшее количество букв, чтобы строка s стала простой. Помогите ему с этой задачей!

Входные данные

В единственной строке находится строка s ($1 \leq |s| \leq 2 \cdot 10^5$) — строка, которая есть у zscoder. Строка s состоит только из строчных английских букв.

Выходные данные

Выведите простую строку s' — строку s после минимального количества изменений. Если существует несколько решений, можете вывести любое из них.

Обратите внимание, что строка s' также должна состоять только из строчных английских букв.

Примеры

входные данные	Скопировать
aab	
выходные данные	Скопировать
bab	
входные данные	Скопировать
caaab	
выходные данные	Скопировать
cabab	
входные данные	Скопировать
zscoder	
выходные данные	Скопировать
zscoder	



В. Новый скейтборд

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

Макс хочет купить новый скейтборд. Он уже посчитал сколько ему нужно денег, чтобы купить новый скейтборд. Оставив калькулятор на полу, он пошёл просить у родителей денег. В это время его маленький брат Юсуф пришёл в комнату и начал нажимать на случайные кнопки калькулятора. К сожалению Макс уже забыл сумму, посчитанную на калькуляторе. Единственное он помнит, что сумма была кратна 4-м.

Вам задана строка s состоящая из цифр (число, которое отображалось на экране калькулятора после того как Юсуф понажимал на случайные кнопки). Ваша задача определить количество подстрок кратных 4-м. Подстрока может начинаться с нуля.

Подстрокой строки называется непустая последовательность подряд идущих символов.

Например, строка 124 содержит четыре подстроки кратные 4-м: 12, 4, 24 и 124. Для строки 04 ответ равен трём: 0, 4, 04.

Рекомендуется для ввода и вывода данных использовать функции `gets`, `scanf`, `printf` в языке C++, поскольку они работают значительно быстрее чем `getline`, `cin`, `cout`. Аналогично, рекомендуется использовать классы `BufferedReader`, `PrintWriter` вместо `Scanner`, `System.out` в языке Java.

Входные данные
В единственной строке находится строка s ($1 \leq |s| \leq 3 \cdot 10^5$). Строка s состоит только из цифр от 0 до 9.

Выходные данные
Выведите целое число a — количество подстрок строки s кратных 4-м.

Обратите внимание, что ответ может быть достаточно большим и не поместиться в 32-битном типе данных. Для сохранения числа вы можете использовать, например, тип `long long` в языке C++ или тип `long` в языке Java.

Примеры

входные данные	Скопировать
124	
выходные данные	Скопировать
4	

входные данные	Скопировать
04	
выходные данные	Скопировать
3	

входные данные	Скопировать
5810438174	
выходные данные	Скопировать
9	

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

A. Разрежь ленточку

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт

У Поликарпа есть ленточка длины n . Он хочет разрезать ее так, чтобы выполнялись два условия:

- После разрезания, каждый кусочек ленточки должен быть длины a , b или c .
- Количество кусочков ленточки после разрезания должно быть как можно больше.

Помогите Поликарпу, найдите количество кусочков ленточки после требуемого разрезания.

Входные данные

В первой строке записано через пробел четыре целых числа n , a , b и c ($1 \leq n, a, b, c \leq 4000$) — длина исходной ленточки и разрешенные длины кусочков ленточки после разрезания, соответственно. Числа a , b и c могут совпадать.

Выходные данные

Выведите одно число — максимально возможное количество кусочков ленточки. Гарантируется, что существует хотя бы одно корректное разрезание ленточки.

Примеры

входные данные	Скопировать
5 5 3 2	
выходные данные	Скопировать
2	

входные данные	Скопировать
7 5 5 2	
выходные данные	Скопировать
2	

Примечание

В первом тестовом примере нужно разрезать ленточку на два кусочка: один из них длины 2, второй длины 3.

Во втором примере нужно разрезать ленточку на два кусочка: один из них длины 5, второй длины 2.

При поддержке



[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

В. Майк и строки

ограничение по времени на тест: 2 секунды
ограничение по памяти на тест: 256 мегабайт

У Майка есть n строк s_1, s_2, \dots, s_n . Каждая строка состоит из маленьких букв латинского алфавита. За один ход он может выбрать строку s_i , удалить первый символ и вставить его в конец этой строки. Например, если у него имеется строка «coolmike», то за один ход он может преобразовать эту строку в строку равную «oolmikes».

Теперь Майк задается вопросом: какое минимальное количество ходов необходимо сделать, чтобы все строки стали равными.

Входные данные

Первая строка содержит целое число n ($1 \leq n \leq 50$) — количество строк.

После этого следуют n строк, каждая из которых содержит строку. i -я строка соответствует строке s_i . Длины строк одинаковы. Длина каждой строки положительна и не превосходит 50.

Выходные данные

Выведите минимальное количество ходов, которое необходимо сделать, чтобы все строки стали равными, или выведите - 1, если решения не существует.

Примеры

входные данные	Скопировать
4 xzzwo zwoxz zzwox xzzwo	
выходные данные	Скопировать
5	
входные данные	Скопировать
2 molzv lzmvo	
выходные данные	Скопировать
2	
входные данные	Скопировать
3 kc kc kc	
выходные данные	Скопировать
0	
входные данные	Скопировать
3 aa aa ab	
выходные данные	Скопировать
-1	

Примечание

В первом тестовом примере оптимальным образом будет преобразовать все строки в строку «zwoxz».