

Задача А. Завоеватель [2 секунды, 256 мегабайт]

И я видел, что Агнец снял первую из семи печатей, и я услышал одно из четырёх животных, говорящее как бы громовым голосом: иди и смотри.

Я взглянул, и вот, конь белый, и на нем всадник, имеющий лук, и дан был ему венец;

и вышел он как победоносный, и чтобы победить.

Откровение Иоанна Богослова

Первый всадник Апокалипсиса Завоеватель пришел на Землю. Увидели это жители одного города и решили предупредить свою столицу, дабы ее жители успели покаяться. С этой целью жителями города в столицу был отправлен гонец на лошади.

Дорога между этим городом и столицей представляет собой прямую, на которой, включая этот город и столицу, расположены n городов, причем расстояние между любыми соседними городами на этой прямой одинаково. В каждом из $n - 2$ городов, мимо которых гонцу необходимо проехать, он может сменить лошадь в конюшне. Хозяин каждой конюшни знает, сколько минут требуется лошади в этой конюшне на дорогу между двумя соседними городами. За какое минимальное время гонец сможет добраться до столицы (n -го города на этой прямой)?

Формат входных данных

В первой строке входного файла содержится одно целое число n ($1 \leq n \leq 100\,000$) — количество городов. В следующей строке содержатся n натуральных чисел t_i ($1 \leq t_i \leq 10^6$) — количество минут, необходимое лошади из конюшни в i -ом городе на преодоление расстояния между двумя городами. Заметим, что первое число означает скорость лошади, которая была у гонца при выезде из первого города, а последнее — скорость лошади в конюшне столицы.

Формат выходных данных

В выходной файл выведите одно число — количество минут, за которое гонец доедет до столицы.

Примеры

conquest.in	conquest.out
6 3 4 3 2 1 5	12

Задача В. Смерть [5 секунд, 256 мегабайт]

И когда Он снял четвертую печать, я слышал голос четвертого животного, говорящий: иди и смотри.
И я взглянул, и вот, конь бледный, и на нем всадник, которому имя «смерть»;
и ад следовал за ним;
и дана ему власть над четвертою частью земли — умерщвлять мечом и голодом, и мором и зверями земными.

Откровение Иоанна Богослова

Смерть — Четвертый всадник Апокалипсиса, и за этим всадником следует ад. Однако, даже этот всадник готов пощадить некоторые города и оставить их жителей в живых.

Карта страны, которую изучает Смерть, представляет собой клетчатый прямоугольник размера $n \times m$. Каждая клетка — город, и в каждом городе живут люди, подчиняющиеся одному определенному лорду. Смерть хочет пощадить несколько городов так, чтобы выполнялись два правила:

- из любого выжившего города можно добраться в любой другой, перемещаясь только по выжившим городам, соседним по стороне
- в выживших городах проживают люди, подчиняющиеся не более, чем двум различным лордам

Теперь Смерть заинтересовало максимальное количество городов, которые он может пощадить.

Формат входных данных

В первой строке входного файла задано два целых числа n и m ($1 \leq n, m \leq 10^3$) — размеры страны. Следующие n строк содержат по m чисел каждая — номера лордов, которым подчиняются люди в соответствующих городах. Номера лордов — натуральные числа, не превышающие 10^6 .

Формат выходных данных

Выведите в первой строке одно натуральное число — максимальное количество городов, которые смогут выжить после набега смерти. Во второй — два числа: номера лордов, которым подчиняются люди в выживших городах. В случае, если выживут только люди, подчиняющиеся одному лорду, номер этого лорда должен быть выведен два раза.

Примеры

death.in	death.out
5 5 1 1 2 2 1 1 2 2 1 1 1 2 3 3 1 1 3 3 2 4 1 2 2 4 1	17 1 2

Задача С. Голод [2 секунды, 256 мегабайт]

И когда Он снял третью печать, я слышал третье животное, говорящее: иди и смотри.
Я взглянул, и вот, конь вороной, и на нем всадник, имеющий меру в руке своей.
И слышал я голос посреди четырех животных, говорящий: хиникс пшеницы за динарий, и три хиникса ячменя за динарий; еля же и вина не повреждай.

Откровение Иоанна Богослова

До конца света осталось не так много дней, но Третий всадник Апокалипсиса Голод так и не смог понять, какие города стоит наказывать за прегрешения, а какие нет. Тогда он решил потренироваться на одной небольшой стране.

Эта страна представляет собой клетчатый прямоугольник размера $n \times m$, в котором каждая клетка — отдельный город. Голод может либо наслать голод на город, либо пощадить его. При этом есть города, в которых еды хватит и до следующего конца света, и Голод не в силах заставить их голодать, а есть те, в которых еды никогда и не было, и им в любом случае придется голодать.

Страшный Суд Голода должен быть еще и справедлив, а это значит, что в любом квадрате размера 2×2 должно быть поровну голодающих и сытых городов. Теперь Голод хочет узнать количество различных вариантов распределения городов этой страны на голодающие и сытые.

Формат входных данных

В первой строке входного файла задано два целых числа n и m ($2 \leq n, m \leq 10^3$) — размер страны. Далее следует n строк по m символов в каждой, где символ '+' означает, что соответствующий город не может голодать, символ '-' — соответствующий город будет голодать в любом случае и символ '.' — Голод может решить, что ему делать с этим городом.

Формат выходных данных

Выведите количество различных вариантов наслать голод на эту страну по модулю $10^9 + 7$

Примеры

famine.in	famine.out
3 2 +. .. .-	2

Задача D. Проклятие Черной жемчужины [1 секунда, 256 мегабайт]

Всем известно, что корабль «Черная жемчужина» на самом деле существует. Долгое время им командовал всем известный капитан Джек Воробей. И этот корабль, вместе со своим капитаном прошел огонь, воду и медные трубы. Так же учеными установлено, что «Черная жемчужина» является самым быстроходным кораблем в мире. Она даже быстрее, чем «Разящий» и «Летучий голландец», про который говорят, что он ходит быстрее ветра.

После ограбления сокровищницы на Исла де Муэрте на команду «Чёрной Жемчужины» легло проклятие, которое повлияло и на сам корабль: паруса корабля порвались, а само судно стал окружать жутковатый туман. Снять проклятие было достаточно сложно, и поэтому никто не стал этого делать.

В трюме корабля, на самой гнилой доске самой дальней стены, написана загадка, которая является ключом к снятию проклятия. Доска очень старая, и из-за этого некоторые буквы на ней стерлись. Согласно легендам, в загадке не было пробелов, то есть она выглядела как одно слово. Поскольку загадку восстановить уже нельзя, снять проклятие не представляется возможным. Однако, его можно попробовать смягчить.

Если верить Мудрецу, для смягчения проклятия нужно выбросить за борт большой мешок с золотом. За каждую подстроку слова-загадки, в которой, при каком-нибудь заполнении пропусков буквами, все буквы могли оказаться одинаковыми, в мешок необходимо положить одну монету.

Вам дана загадка, определите, сколько монет нужно выбросить за борт для смягчения проклятия.

Формат входных данных

Во входном файле дана строка, длина которой не превышает 10^6 . Строка состоит из строчных латинских букв и знаков вопроса, обозначающих стертую букву (пропуск).

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

theblackpearl.in	theblackpearl.out
ab?c	6
aa???b?c	19

Задача Е. Два капитана [3 секунды, 256 мегабайта]

Как известно, у Черной Жемчужины два капитана: капитан Джек Воробей и Барбосса. Корабль содержит ровно n пушек, расположенных в ряд. Во время боя оба капитана раз в минуту одновременно дают команды своим матросам. Команды бывают следующих видов:

- **send** $l\ r$ — послать своих матросов стрелять из пушек с номерами от l до r включительно
- **back** $l\ r$ — отозвать всех своих матросов с пушек с номерами от l до r включительно. Если на каких-то пушках из этого отрезка нет матросов, подчиняющихся этому капитану, то с такими пушками ничего не происходит
- **rum** — принести еще одну бутылку рома

Каждая команда выполняется мгновенно, после чего сражение идет еще минуту до следующей команды. Если в какой-то момент у одной и той же пушки окажутся матросы, подчиняющиеся разным капитанам, они подерутся и убьют друг друга. Эта ситуация не устраивает никого из капитанов, и поэтому они обратились к вам с просьбой помочь им в решении этой проблемы.

Перед началом очередного сражения капитан Джек Воробей и Барбосса составили планы своих действий. Известно, что план капитана Джека Воробья состоит из m_1 команд, а план Барбоссы — из m_2 команд. В начале i -ой минуты боя каждый капитан дает своим матросам i -ую команду из своего плана, если в нем есть хотя бы i команд. Вам поручили исправить планы так, чтобы все матросы остались живы. Единственная доступная вам модификация плана сражения — вставка нескольких команд **rum** в любые места. Понятно, что капитаны не очень любят менять свои планы, поэтому суммарное количество команд, добавленных Вами в оба плана, должно быть минимально.

Формат входных данных

В первой строке дано число n — количество пушек на корабле ($1 \leq n \leq 10^9$).

Во второй строке задано число m_1 — количество команд в плане Джека Воробья ($1 \leq m_1 \leq 3000$). В следующих m_1 строках перечислены команды из плана Джека Воробья. Команды заданы так, как они описаны выше. Для всех команд, использующих l и r , верно, что $1 \leq l \leq r \leq n$. Гарантируется, что последняя команда в плане — **back** 1 n .

Во следующей строке задано число m_2 — количество команд в плане Барбоссы ($1 \leq m_2 \leq 3000$). В следующих m_2 строках перечислены команды из плана Барбоссы. Команды заданы так, как они описаны выше. Для всех команд, использующих l и r , верно, что $1 \leq l \leq r \leq n$. Гарантируется, что последняя команда в плане — **back** 1 n .

Формат выходных данных

В единственной строке выведете минимальное количество дополнительных команд.

Примеры

twocaptains.in	twocaptains.out
3 4 send 1 1 send 2 2 back 1 1 back 1 3 5 send 2 3 send 1 1 back 2 2 rum back 1 3	3

Задача F. Война [2 секунды, 256 мегабайт]

И когда он снял вторую печать, я слышал второе животное,
говорящее: иди и смотри.
И вышел другой конь, рыжий;
и сидящему на нем дано взять мир с земли, и чтобы убивали друг
друга;
и дан ему большой меч.

Откровение Иоанна Богослова

Мир перенаселён. Ресурсы планеты на исходе. Жители всех стран на грани отчаяния. Политическая обстановка накаляется с каждым днём. У государств нет ни сил, ни ресурсов на войну друг с другом. В любой момент в любой стране может вспыхнуть кровопролитная гражданская война. Миллионы жертв неизбежны. В результате прошлых войн многие государства оказались разделены географически, образовались анклавов.

Представим карту мира как прямую, на которой расположены города, принадлежащие разным странам. Если на этой прямой подряд находятся несколько городов, принадлежащих одной стране, то все эти города считаются отдельным анклавом этой страны. Если в каком-то анклав какой-то страны начинается гражданская война, то этот анклав мгновенно слабеет, после чего его целиком захватывает один из соседних с ним анклавов. Если оба соседних анклава принадлежат одной и той же стране, то они объединяются в один анклав.

Второй всадник Апокалипсиса Война, видя приближение конца света, хочет закончить всё это побыстрее. Он может инкогнито внедрить провокатора в любой анклав любой страны и начать там гражданскую войну, что приведет к исчезновению этого анклава с карты мира. Выясните, за сколько внедрений Война сможет оставить в мире ровно один анклав и начать в нем гражданскую войну.

Формат входных данных

В первой строке задано единственное число n ($1 \leq n \leq 500$) — количество стран на карте мира. Во второй строке задана сама карта мира: номера стран a_i , которым принадлежат соответствующие города на прямой. Все номера стран являются натуральными числами, не превосходящими 10^9 .

Формат выходных данных

Выведите искомое минимальное число внедрений.

Примеры

war.in	war.out
4 1 2 3 1	3
3 1 1 1	1
4 1 2 2 1	2