

Задача А. Бутфол

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	512 мегабайт

Тима и его N друзей очень любят играть в *Бутфол*. *Бутфол* — спортивная игра, в которой участвуют $N + 1$ игроков. Каждый игрок имеет силу, которая характеризуется целым положительным числом. Игра состоит из $N + 1$ раундов, в каждом раунде кто-то из игроков записывает раунд на видео, а остальные N игроков делятся на две команды так, что каждый игрок будет в одной из двух команд и обе команды непустые. Сила команды — это сумма сил всех игроков в команде. Также, каждый игрок должен снимать на видео **ровно в одном** раунде.

Раунд называется *ничейным* если существует разбиение на две команды с **равной** силой, и игра называется *дружной* если **все раунды ничейные**. Каждый из N друзей сообщил Тиме свою силу, а сам Тима может выбрать себе силу любой допустимой величины.

По заданным значениям сил N друзей, помогите Тиме определить все варианты сил, которые он может выбрать себе так, чтобы игра могла стать *дружной*.

Формат входных данных

В первой строке входных данных находится целое число N ($1 \leq N \leq 500$) — количество друзей у Тимы. Во второй строке находятся N целых чисел $a_1, a_2, \dots, a_N$ ($1 \leq a_i \leq 500$; $1 \leq i \leq N$) разделенных через единичный пробел, a_i — сила i -го игрока.

Формат выходных данных

В первой строке выведите одно целое число K — количество способов выбрать Тиме силу. Если же не существует вариантов сил для Тимы, выведите «0» (без кавычек), иначе во второй строке выведите K целых положительных чисел разделенных через одиночный пробел — сами значения сил для Тимы, значения следует выводить в **возрастающем** порядке.

Примеры

стандартный ввод	стандартный вывод
4 1 3 1 5	1 3
6 3 5 7 11 9 13	4 1 3 17 19
3 2 2 2	0
4 200 200 200 200	2 200 600

Замечание

Пояснение к первому примеру.

Покажем, что если Тима выберет себе силу 3, то игра может быть *дружной*.

— Когда Тима будет отвечать за съемку, чтобы раунд был *ничейным*, остальные могут поделиться таким образом : (1, 3, 1) в первой команде, и (5) во второй.

— Когда друг с номером 1 будет отвечать за съемку, остальные могут поделиться таким образом: (1, 5) в первой команде, (3, 3) во второй.

— Когда друг с номером 2 будет отвечать за съемку, остальные могут поделиться таким образом: (1, 1, 3) в первой команде, (5) во второй.

— Когда друг с номером 3 будет отвечать за съемку, остальные могут поделиться таким образом: (3, 3) в первой команде, (1, 5) во второй.

— Когда друг с номером 4 будет отвечать за съемку, остальные могут поделиться таким образом: (1, 3) в первой команде, (1, 3) во второй.

Если Тима выберет себе силу не равной 3, то игра не может быть *дружной*.

Задача В. Халява

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

Все домики в ЛКШ пронумерованы от 1 до n . В домике с номером 1 живет большая часть преподавателей. Также все домики связаны между собой двусторонними тропинками с определенными длинами, так чтобы от каждого домика до любого другого можно было добраться единственным образом.

Из-за того, что все домики на базе деревянные, то иногда они горят. В случае ночного пожара школьникам необходимо оповестить об этом преподавателей, но ходить самим по ночной базе им страшно. Как раз на такой случай во всехдомиках есть дежурный преподаватель. Все дежурные преподаватели действуют по одному алгоритму во время экстренных ситуаций. Во-первых, i -му дежурному нужно a_i секунд, чтобы проснуться и прийти в себя. Также для каждого известно, за сколько секунд он пробегает один метр — v_i . После пробуждения в i -м домике преподаватель ждёт a_i секунд и выдвигается в сторону домика 1. При этом, когда он заходит по пути в другой домик, у него есть опция разбудить дежурного этого домика и отправить его вместо себя, при этом промежуточному преподавателю нужны свои a_j секунд, чтобы проснуться.

Помогите школьникам из каждого домика узнать, какое минимальное время они будут гореть, пока все преподаватели не узнают о бедствии.

Формат входных данных

В первой строке входного файла содержится число $n \leq 10^5$ — количество домиков на базе. В следующих $n - 1$ строках содержится описание тропинок. Каждая строка задает одну тропинку тремя числами u, v, d , где $d \leq 10^4$ — длина в метрах, а тропинка соединяет дома u и v . Далее идет еще $n - 1$ строка, задающие характеристики дежурных преподавателей во всехдомиках, кроме 1-го, т.е. i -я из этих строк задает дежурного в $i + 1$ -м домике. Каждая строка содержит числа $a_i \leq 10^9$ — время на пробуждение в секундах и $v_i \leq 10^9$ — скорость.

Формат выходных данных

Выведите $n - 1$ число в одну строку, i -е число задаёт минимальное время, которое будут гореть школьники в $i + 1$ -м домике, пока про это не узнает штаб преподавателей — 1-й домик.

Пример

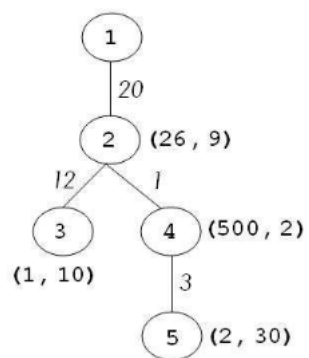
стандартный ввод	стандартный вывод
5 1 2 20 2 3 12 2 4 1 4 5 3 26 9 1 10 500 2 2 30	206 321 542 328

Замечание

Дороги и их протяженность показаны на рисунке. Время запуска и скорость движения преподавателей указаны в скобках.

Минимальное время для отправки сообщения из домика 5 в домик 1 достигается следующим образом. Преподаватель из города 5 получает сообщение и покидает домик через 2 секунды. Он проходит 4 метра за 120 секунд и приходит в домик 2. Там он передаёт эстафету преподавателю из этого домика. Второй преподаватель начинает бежать через 26 секунд и бежит 180 секунд, прежде чем прибегает в домик 1.

Таким образом, общее время составляет $2 + 120 + 26 + 180 = 328$.



Задача С. Как скоротать время

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

Васе и Пете нечем заняться, и они очень хотят придумать, как себя развлечь. К счастью, Вася вспомнил, что совсем недавно к нему в руки попала записка из N строчных латинских букв, и он предположил (не знаем, на каком основании), что в этой записке зашифрован город, в котором будет проводиться следующая всероссийская олимпиада школьников по информатике. А если точнее, то он уверен, что город — это некоторая подпоследовательность данной записки, то есть строка, которая может быть получена из неё удалением нескольких (возможно, нуля) символов.

Так как мальчикам всё ещё нечем заняться, они решили перебирать все варианты, пока не найдут подходящий. Однако им интересно, насколько много времени они смогут таким образом убить. Помогите им: скажите, какое количество различных подпоследовательностей существует у этой записки. Так как ответ может быть очень большим, они решили, что их устроит остаток от деления этого количества на $10^9 + 7$.

Формат входных данных

В единственной строке находится строка длины N ($1 \leq N \leq 5 \cdot 10^5$) из строчных латинских букв — записка Васи.

Формат выходных данных

Выведите одно число: количество различных подпоследовательностей по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
aybabbu	103

Задача D. Олег

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

Как известно, не только люди любят принимать участие в различных соревнованиях. Среди рыб и моллюсков также проводится множество состязаний на дне океана. Акула Олег участвует в самом престижном соревновании Мирового океана на звание самой опасной акулы. Во время этого состязания акулы соревнуются в различных дисциплинах: плавание на скорость, маскировка, навигация по картам и многих других. Сейчас Олег проходит испытание под названием «разрушение».

Во время этого испытания перед акулой ставят m доминошек. Все доминошки стоят на одной прямой, однако высоты доминошек могут различаться. Расстояние между соседними доминошками равно 1. Кроме того, у каждой доминошки есть своя стоимость, выраженная целым числом. Цель акулы — уронить все доминошки. Для этого акула может толкнуть любую доминошку влево или вправо, после чего она начнёт падать в этом направлении. Если во время падения доминошка задевает другие, они также начинают падать в ту же сторону, в которую падала исходная, таким образом начинается цепная реакция, в результате которой может упасть множество доминошек. Падающая доминошка задевает другую, если и только если расстояние между ними строго меньше высоты падающей доминошки, причём доминошки не обязательно должны быть соседними.

Разумеется, любая акула справится с тем, чтобы уронить все доминошки таким образом, поэтому оценивается не факт разрушения, а его стоимость. Стоимостью разрушения называется сумма стоимостей доминошек, которые акуле придётся толкнуть, чтобы все доминошки упали.

Олег уже победил в прошлых испытаниях, однако недостаточно умён, чтобы победить в этом. Помогите Олегу и определите, какая минимальная суммарная стоимость доминошек, которые ему придётся толкнуть, чтобы все доминошки упали.

Формат входных данных

Для уменьшения размера ввода высоты и стоимости доминошек задаются при помощи блоков.

В первой строке заданы два целых числа n и m ($1 \leq n \leq 250\,000$, $1 \leq m \leq 10^7$) — количество блоков и суммарное количество доминошек, которые надо уронить Семёну, соответственно.

Затем следует описание n блоков. Описание каждого блока состоит из трёх строк.

В первой строке описания блока с номером i содержится целое число k_i ($1 \leq k_i \leq 250\,000$, $\sum_{i=1}^n k_i \leq 250\,000$) — количество доминошек в блоке.

Во второй строке описания блока с номером i содержатся k_i целых чисел $a_{i,j}$ ($1 \leq a_{i,j} \leq m$) — высоты доминошек в блоке.

В третьей строке описания блока с номером i содержатся k_i целых чисел $c_{i,j}$ ($1 \leq c_{i,j} \leq 100\,000$) — стоимости доминошек в блоке.

Далее следует описание последовательности доминошек, которые надо уронить Семёну, в порядке слева направо.

В первой строке описания последовательности задано целое число q ($n \leq q \leq 250\,000$) — количество блоков в последовательности доминошек, которые надо уронить.

В следующих q строках содержатся пары целых чисел $id_i \ mul_i$ ($1 \leq id_i \leq n$, $1 \leq mul_i \leq 100\,000$), обозначающие, что очередные k_{id_i} в порядке слева направо доминошек — это доминошки блока id_i , чьи стоимости были умножены на число mul_i .

Гарантируется, что $\sum_{i=1}^q k_{id_i} = m$, а также, что каждый блок встречается хотя бы один раз в последовательности доминошек, которые требуется уронить, то есть для всех i от 1 до n найдётся j такое, что $id_j = i$.

Формат выходных данных

Выведите одно целое число — минимальную суммарную стоимость доминошек, которые надо толкнуть, чтобы все доминошки упали.

Примеры

стандартный ввод	стандартный вывод
2 7 3 1 2 2 1 2 1 1 3 2 3 2 2 1 3 1 1	5
1 1 1 1 100000 1 1 100000	10000000000

Замечание

В первом примере перед Семёном стоят 7 доминошек. Их высоты равны [3, 1, 2, 2, 1, 2, 2], а стоимости равны [4, 3, 6, 3, 1, 2, 1]. Сначала Семёну следует уронить доминошку с номером 7 влево, она упадёт и заденет доминошку с номером 6. Доминошка 6, падая, заденет доминошку с номером 5, которая упадёт, но не заденет другие доминошки. Затем Семён должен уронить доминошку с номером 1 вправо, она, падая, заденет доминошки с номерами 2 и 3, а доминошка 3, падая, заденет доминошку 4, таким образом все доминошки упадут.

Во втором примере перед Семёном стоит одна доминошка стоимостью 10000000000.

Задача Е. Долгое путешествие

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

Есть полоса, разделенная на ячейки, пронумерованные от 0 до m слева направо. Вы управляете фишкой, которая изначально находится в ячейке 0.

В каждой ячейке есть ловушка; они активируются согласно следующим правилам:

- в конце ходов $1, (1 + n), (1 + 2n), \dots$, ловушки активируются в ячейках x , где $x \bmod a_1 = b_1$;
- в конце ходов $2, (2 + n), (2 + 2n), \dots$, ловушки активируются в ячейках x , где $x \bmod a_2 = b_2$;
- ...
- в конце ходов $n, (n + n), (n + 2n), \dots$, ловушки активируются в ячейках x , где $x \bmod a_n = b_n$.

За один ход вы можете либо переместиться из текущей ячейки в следующую, либо остаться на месте. Затем все ловушки для этого хода активируются. Если фишка находится в ячейке с активированной ловушкой в начале хода, игра заканчивается.

Ваша задача — вычислить минимальное количество ходов, чтобы достичь ячейки m , или сообщить, что это невозможно. **Если фишка достигает клетки m и в конце того же самого хода в клетке m активируется ловушка, это не считается корректным путем до клетки m .**

Формат входных данных

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

Первая строка каждого набора входных данных содержит два целых числа n и m ($1 \leq n \leq 10$; $1 \leq m \leq 10^{12}$).

Вторая строка содержит n целых чисел $a_1, a_2, \dots, a_n$ ($2 \leq a_i \leq 10$).

Третья строка содержит n целых чисел $b_1, b_2, \dots, b_n$ ($0 \leq b_i < a_i$).

Формат выходных данных

Для каждого набора входных данных выведите одно целое число — минимальное количество ходов, чтобы достичь ячейки m . Если это невозможно, выведите -1.

Пример

стандартный ввод	стандартный вывод
5	5
2 5	6
2 2	-1
0 1	424796303424
2 5	5
2 2	
1 0	
1 7	
3	
2	
4 212398151713	
3 2 5 2	
0 1 3 0	
2 4	
3 4	
0 0	

Задача F. Kaavi и магическое заклинание

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

Kaavi, таинственная гадалка, глубоко верит в то, что судьба человека неизбежна и неотвратима. Конечно, она зарабатывает себе на жизнь предсказывая будущее других людей. Занимаясь гаданием, Kaavi верит, что магические заклинания могут дать ей огромную силу видеть будущее.

У Kaavi есть строка T длины m и все строки, имеющие префикс T это магические заклинания. У Kaavi также есть строка S длины n и пустая строка A .

Во время гадания, Kaavi нужно применить некоторую последовательность операций. Есть две различные операции:

- Удалить первый символ строки S и добавить его **в начало** строки A .
- Удалить первый символ строки S и добавить его **в конец** строки A .

Kaavi может сделать **не более, чем n** операций. Для того, чтобы завершить гадание, она хочет узнать количество различных последовательностей операций, для того, чтобы сделать A магическим заклинанием (то есть эта строка имеет префикс T). Как ее ассистент, можете ли вы помочь ей? Поскольку ответ может быть очень большим, Kaavi хочет знать его остаток при делении на 998 244 353.

Две последовательности операций считаются различными, если они различаются по длине или существует такое i , что их i -е операции различаются.

Подстрокой называется некоторая последовательность последовательных символов строки. Префиксом строки S называется подстрока S , которая начинается в начале строки S .

Формат входных данных

В первой строке находится строка S длины n ($1 \leq n \leq 3000$).

Во второй строке находится строка T длины m ($1 \leq m \leq n$).

Обе строки состоят только из строчных латинских символов.

Формат выходных данных

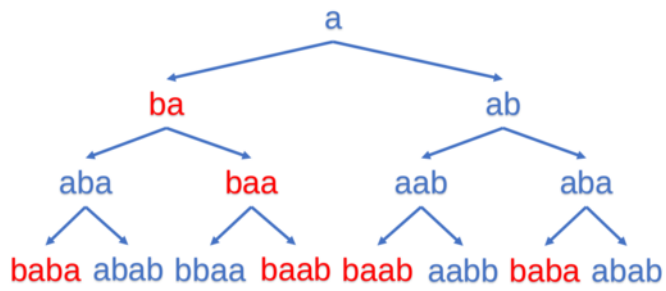
Выведите единственное целое число — ответ на задачу по модулю 998 244 353.

Примеры

стандартный ввод	стандартный вывод
abab ba	12
defineintlonglong signedmain	0
rotator rotator	4
cacdcdbbbb bdcacdbbbb	24

Замечание

В первом тесте:



Красные строки это магические заклинания. В первой операции, Кааві может добавить символ «a» в начало или в конец A. Хотя в обоих случаях результат будет одинаковым, это две разные операции. Поэтому ответ $6 \times 2 = 12$.

Задача G. Перемещение фишки

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

На координатной прямой лежит фишка. Изначально фишка находится в точке 0. Вы можете переместить фишку любое количество раз; каждый раз при перемещении ее координата увеличивается на некоторое целое положительное число (будем его называть *длиной перемещения*). При этом длина первого перемещения должна делиться на k , длина второго — на $k + 1$, длина третьего — на $k + 2$, и так далее.

Например, если $k = 2$, то последовательность перемещений может выглядеть следующим образом: $0 \rightarrow 4 \rightarrow 7 \rightarrow 19 \rightarrow 44$, т. к. $4 - 0 = 4$ делится на $2 = k$, $7 - 4 = 3$ делится на $3 = k + 1$, $19 - 7 = 12$ делится на $4 = k + 2$, $44 - 19 = 25$ делится на $5 = k + 3$.

Вам дано два целых положительных числа n и k . Ваша задача — посчитать количество способов попасть в точку x , начиная из точки 0, для всех $x \in [1, n]$. Количество способов может быть очень большим, поэтому выведите его по модулю 998244353. Два способа считаются различными, если они отличаются как наборы посещенных позиций.

Формат входных данных

Единственная строка содержит два целых числа n и k ($1 \leq k \leq n \leq 2 \cdot 10^5$).

Формат выходных данных

Выведите n целых чисел — количество способов попасть из 0 в точку x для всех $x \in [1, n]$, взятое по модулю 998244353.

Примеры

стандартный ввод	стандартный вывод
8 1	1 1 2 2 3 4 5 6
10 2	0 1 0 1 1 1 1 2 2 2

Замечание

Рассмотрим способы в первом примере:

Способы попасть в точку 1: $[0, 1]$;

Способы попасть в точку 2: $[0, 2]$;

Способы попасть в точку 3: $[0, 1, 3]$, $[0, 3]$;

Способы попасть в точку 4: $[0, 2, 4]$, $[0, 4]$;

Способы попасть в точку 5: $[0, 1, 5]$, $[0, 3, 5]$, $[0, 5]$;

Способы попасть в точку 6: $[0, 1, 3, 6]$, $[0, 2, 6]$, $[0, 4, 6]$, $[0, 6]$;

Способы попасть в точку 7: $[0, 2, 4, 7]$, $[0, 1, 7]$, $[0, 3, 7]$, $[0, 5, 7]$, $[0, 7]$;

Способы попасть в точку 8: $[0, 3, 5, 8]$, $[0, 1, 5, 8]$, $[0, 2, 8]$, $[0, 4, 8]$, $[0, 6, 8]$, $[0, 8]$.

Задача Н. Пары вершин

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

Давайте назовем упорядоченную пару вершин (u, v) в ориентированном графе *однонаправленной* если $u \neq v$, существует путь от u до v , и не существует пути от v до u .

Ориентированный граф называется *p -достижимым*, если он содержит ровно p упорядоченных пар вершин (u, v) таких, что $u < v$, u и v достижимы друг из друга. Найдите минимальное количество вершин, необходимое для создания p -достижимого ориентированного графа.

Еще, среди всех p -достижимых ориентированных графов с минимальным числом вершин, пусть G обозначает граф, который максимизирует количество однонаправленных пар вершин. Найдите это число.

Формат входных данных

Первая и единственная строка содержит одно целое число p ($0 \leq p \leq 2 \cdot 10^5$) — количество упорядоченных пар вершин.

Формат выходных данных

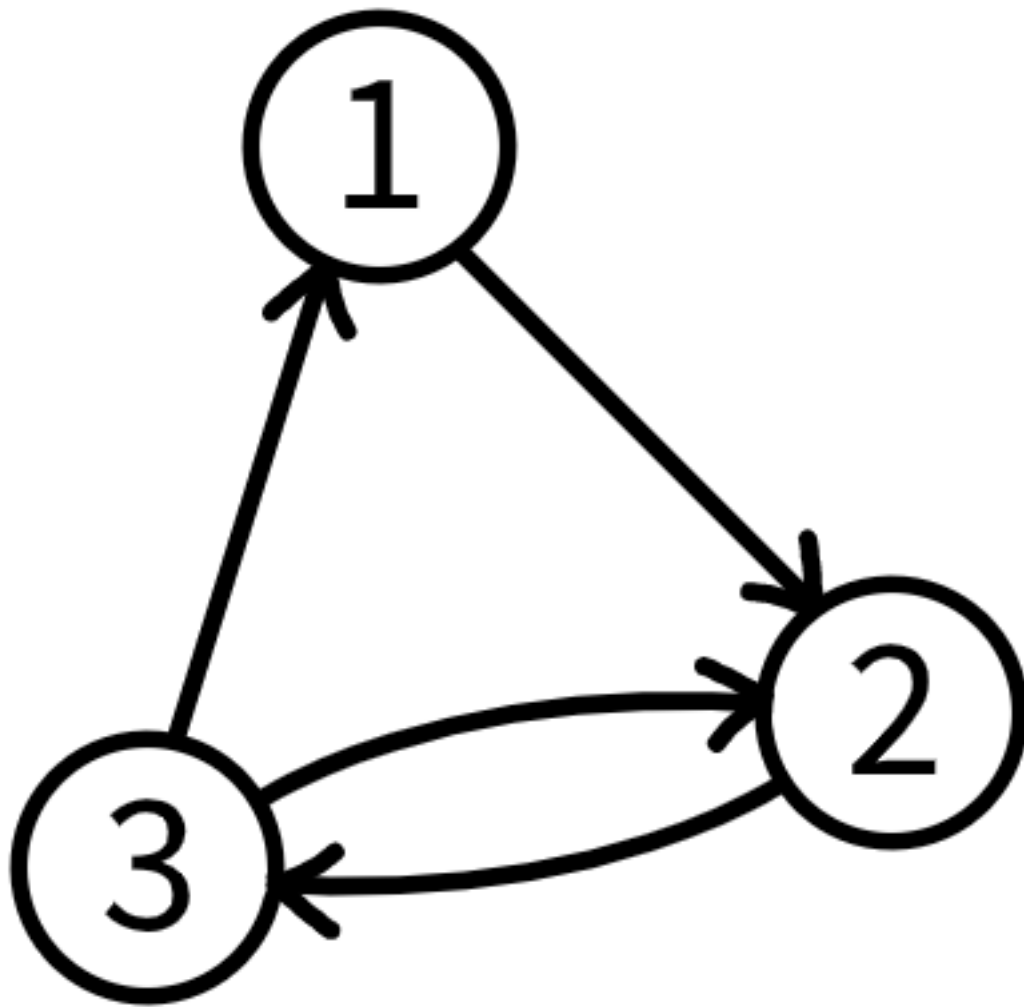
Выведите одну строку, содержащую два целых числа — минимальное количество вершин, необходимое для создания p -достижимого ориентированного графа, и максимальное количество однонаправленных пар вершин среди всех таких p -достижимых ориентированных графов с минимальным количеством вершин.

Примеры

стандартный ввод	стандартный вывод
3	3 0
4	5 6
0	0 0

Замечание

В первом примере минимальное количество вершин, необходимое для создания 3-достижимого ориентированного графа это 3. Среди всех 3-достижимых ориентированных графов с 3 вершинами, следующий граф G является одним из графиков с максимальным количеством однонаправленных пар вершин, который является 0.



Задача I. Плитка в Полянах

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 512 мегабайт

В этом году в Берендеевых Полянах меняют плитку. Они хотят сделать покрытие более прочным и, что не менее важно, более красивым. Вот, например, на площади перед столовой строители хотят выложить совсем прекрасный узор. Только вот у них странные представления о прекрасном...

Площадь представляет собой таблицу $N \times M$, в каждой ячейке которой находится плитка одного из двух цветов: черного или белого. В понимании строителей, узор на площади будет *прекрасным*, если:

1. Должна быть хотя бы одна чёрная плитка.
2. От любой чёрной плитки можно дойти до любой другой, передвигаясь только между соседними по стороне чёрными плитками.
3. Не существует простого цикла из соседних по стороне чёрных плиток.

Строители уже заменили все испортившиеся плитки, и теперь хотят сделать узор прекрасным, а для этого просят вас сообщить им, у какого минимального числа плиток им нужно поменять цвет, и какой узор у них получится в результате.

Формат входных данных

В первой строке даны числа N, M ($1 \leq N \leq 100, 1 \leq M \leq 10$) — размеры площади.

В следующих N строках записаны по M символов `.` и `#`, где `.` обозначает белую плитку, а `#` — чёрную.

Формат выходных данных

Выведите N строк по M символов, как во входных данных — итоговый узор.

Примеры

стандартный ввод	стандартный вывод
3 3 ### #. # ###	### #. # ##.
4 3 ##. .## ### ##.	##. .## ##. #..
2 3	... #..