**Croatian Open Competition in Informatics, Round 2, December 12th 2020**

**Task 1. Knjige 1 sec / 512 MB / 50 points**

Тин не любит Барселону. Сегодня Тин пришёл в гости к другу Анти, чтобы выяснить, кто из них лучший игрок в FIFA. И его ждал неприятный сюрприз. У Анти есть две книжные полки, подвешенные рядом. Левая полка содержит N книг о многочисленных достижениях Барселоны, сложенных друг на друга, а правая пустая.

Книги не отсортированы от самой тонкой до самой толстой. Анти решил сделать это для Тина. На каждом ходу он может:

* Взять книгу сверху с некоторой полки в левую или правую руку, если он уже не держит какую-то книгу в этой руке
* Положить книгу которую он держит в некоторой руке на верх некоторой полке

Вы должны помочь Анти найти некоторую последовательность ходов, такую, при выполнении которой все книги окажутся на левой полке отсортированные от самой тонкой до самой толстой сверху вниз.

Ввод

Первая строка ввода содержит целое число N (1<=N<=100), количество книг на левой полке.

Вторая строка содержит N целых чисел Di (1<=Di<=1000), которые представляют толщину книг, сверху вниз.

Вывод

В первой строке выведите целое число K (0<=K<=100 000), количество ходов в Вашем решении.

В следующих K строках выведите ходы в форме ИНСТРУКЦИЯ РУКА ПОЛКА, где  
 ИНСТРУКЦИЯ – это слово UZMI (хорватское «возьми») ,   
 если Анти должен взять книгу с некоторой полки

или STAVI (хорватское «поставь»),   
 если Анти должен положить книгу на некоторую полку

РУКА - это буква L, если Анти хочет использовать левую руку  
 D, если Анти хочет использовать правую руку  
 (по-хорватски вправо - desno)

ПОЛКА - это буква L, если Анти хочет использовать левую руку  
 D, если Анти хочет использовать правую руку

Ваше решение не должно быть минимально возможной длины, но количество ходов должно не превышать 100 000. Можно доказать, что при заданных ограничениях, решение существует всегда.

Примеры (рисунок к первому примеру)

|  |  |  |
| --- | --- | --- |
| Ввод  3  2 3 1  Вывод  8  UZMI L L  STAVI L D  UZMI L L  UZMI D L  STAVI L L  UZMI L D  STAVI L L  STAVI D L | Ввод  4  1 1 2 5  Вывод  0 |  |

**Croatian Open Competition in Informatics, Round 2, December 12th 2020**

**Task 2. Vlak 1 sec / 512 MB / 70 points**

Нина и Эмилия играют в игру на листке бумаги. Изначально бумага пуста. На своём ходу игрок добавляет букву в конец слова, которое написано на бумаге. Игроки ходят по очереди, начинает Нина.

Игроки должны выбирать букву таким образом, чтобы выполнялось следующее ограничение: слово, которое написано после хода игрока, должно быть префиксом некоторого слова в любимой песне этого игрока. Если игрок не может сделать ход, он поиграл.

Определите, кто победит, если оба игрока играют оптимально.

Ввод

Первая строка содержит положительное целое число N – количество слов в любимой песне Нины.

Каждая из последующих N строк содержит слово из любимой песни Нины.  
Следующая строка содержит положительной целое число M – количество слов в любимой песне Эмилии. Каждая из последующих M строк содержит слово из любимой песни Эмилии.

Слова на вводе содержат только маленькие латинские буквы, суммарная длина всех слов не превышает 200 000.

Вывод

Выведите Nina или Emilija – имя победителя.

Оценивание

В тестах на 40 баллов сумма длин не превысит 2000.

Примеры

|  |  |  |  |
| --- | --- | --- | --- |
| Ввод  2  aaa  bbb  3  aab  aba  bbb  Вывод  Nina | Ввод  2  acg  beh  2  adi  bfj  Вывод  Emilija | Ввод  3  ja  sam  vlak  5  sto  zgazit  ce  te  mali  Вывод  Nina | Пояснение к первому примеру.  Если Нина напишет b, Эмилия должна написать и тогда Нина может написать b. Текущее слово bbb и Эмилия не может сделать ход, поэтому выиграла Нина.  Если Нина напишет сначала a, Эмилия может написать b. Текущее слово ab и Нина не может сделать ход, она проиграла. |

**Croatian Open Competition in Informatics, Round 2, December 12th 2020**

**Task 3. Sateliti 3 sec / 512 MB / 110 points**

Для целей исследования кратера, телескоп Arecibo записывает образы спутников Сатурна. Команда ученых должна различать образы спутника и группы образов по спутникам, но это не просто, поскольку спутники могут быть сняты под разными углами.

Образ отображается матрицей N\*M символов, вида ‘\*’(кратер) и ‘.’ (ровная поверхность). Мы говорим, что два образа соответствуют одному спутнику если можно получить один образ из другого **циклическими сдвигами строк или столбцов.**

Чтобы сделать процесс верификации проще, учёные хотят найти лексикографически минимальный образ, соответствующий спутнику с данного образа. При сравнении образов мы сравниваем строки, полученные конкатенацией всех строк образа, где символы сравниваются по их ASCII-кодам.

Ввод

Первая строка содержит целые числа N и M (1<=N,M<=1000), размеры образа.

Каждая из последующих N строк содержит M символов ‘\*’ или ‘.’. Они представляют образ.

Вывод

Выведите N строк по M символов в каждой - лексикографически минимальный образ.

Оценивание

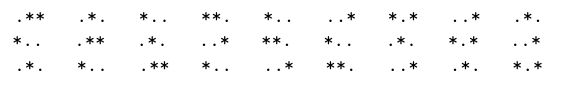
|  |  |  |
| --- | --- | --- |
| Подзадача | Баллы | Ограничения |
| 1 | 10 | 1<=N,M<=50 |
| 2 | 40 | 1<=N,M<=300 |
| 3 | 60 | Нет дополнительных ограничений |

Примеры

|  |  |  |
| --- | --- | --- |
| Ввод  3 3  .\*\*  \*..  .\*.  Вывод  \*\*.  ..\*  \*.. | Ввод  3 4  ....  ..\*.  ....  Вывод  \*...  ....  .... | Ввод  3 5  .\*\*..  .\*\*\*.  ..\*\*.  Вывод  \*\*\*..  .\*\*..  \*\*... |

Пояснение к первому примеру.

Вот все образы, которые могут быть получены циклическими сдвигами:



**Croatian Open Competition in Informatics, Round 2, December 12th 2020**

**Task 4. Selotejp 1 sec / 512 MB / 110 points**

Для Мирко нет большего счастья, чем найти новый рулон липкой ленты, а сегодня он особенно счастлив, поскольку он нашёл календарь Славко.

Этот календарь может быть представлен таблицей с N строками и M столбцами. Каждый квадрат содержит маленькое окно, а позади каждого окна кусок шоколада. Славко уже открыл некоторые окна, а другие пока ещё закрыты.

Мирко решил использовать свою липкую ленту, чтобы заклеить все закрытые окна. Лента бесконечно длинная и имеет ширину на одну ячейку календаря. Мирко может отрезать кусок ленты и использовать его, чтобы заклеить последовательность горизонтальных или вертикальных соседних ячеек. Он не использует более одного куска ленты для любого окна.

Мирко хочет определить минимальное количество кусков ленты, которые потребуются, ему чтобы заклеить все закрытые окна.

Ввод

Первая строка содержит целые числа N и M (1<=N<=1000, 1<=M<=10) – размеры календаря.

Каждая из последующих N строк содержит M символов ‘.’ и ‘#’, которые представляют календарь. Символ ‘.’ обозначает открытое окно, символ ‘#’ обозначает закрытое окно.

Вывод

Выведите минимальное количество кусков ленты, которые потребуются, чтобы заклеить все закрытые окна.

Оценивание

|  |  |  |
| --- | --- | --- |
| Позадача | Баллы | Ограничения |
| 1 | 35 | Каждое закрытое окно соседствует не более чем с двумя закрытыми окнами |
| 2 | 35 | 1<=N<=10 |
| 3 | 40 | Нет дополнительных ограничений |

Примеры

|  |  |  |
| --- | --- | --- |
| Ввод  2 3  #.#  ###  Вывод  3 | Ввод  4 3  .#.  ###  .##  .#.  Вывод  3 | Ввод  4 4  ####  #.#.  #.##  ####  Вывод  5 |

Пояснение к первому примеру:

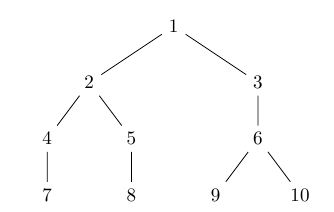
Одно возможное решение – использовать первый кусок ленты для первой колонки, второй для третьей колонки, и третий кусок для окна во второй строке и второй колонке.

**Croatian Open Competition in Informatics, Round 2, December 12th 2020**

**Task 5. Specijacija 4 secs / 1 GB / 110 points**

Вам дано целое число N и последовательность положительных целых чисел таких, что   
i\*(i-1)/2 <ai<=i\*(i+1)/2

Эта последовательность параметризует дерево с (N+1)\*(N+2) вершинами, содержащее N+1 уровень с 1, 2, …, N+1 вершинами следующим образом:



Дерево, параметризованное последовательностью a = (1, 2, 6)

I-ый уровень содержит вершины i\*(i-1)/2 +1, …, i\*(i+1)/2. Вершина ai имеет двух детей, остальные вершины имеют ровно по одному ребенку каждая.

Мы хотим ответить на Q запросов вида «Каков наибольший общий предшественник вершин x и y»,

то есть вершина с наибольшей меткой, которая является предшественником обоих вершин x и y.

Ввод

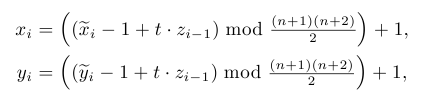
Первая строка содержит целые числа N, Q, T (1<=N,Q<=200 000), t=0 или 1, соответственно

количество параметров, количество запросов, и величина, которая будет использована для определения меток вершин в запросах.

Вторая строка содержит последовательность из N чисел ai (i\*(i-1)/2 <ai<=i\*(i+1)/2), которые параметризуют дерево.

i-ая из последующих Q строк содержит два целых числа XTi и YTi (1<=XTi,YTi<=(N+1)\*(N+2)/2), которые будут использованы чтобы определить метки запросов.

Пусть Zi это ответ на i-ый запрос, и пусть Z0=0; Метки i-го запросы Xi, Yi вычисляются так:



Где mod это остаток от целочисленного деления.

Замечание: Если T=0 то Xi=XTi, Yi=YTi поэтому все запросы известны из ввода.  
Если T=1, запросы не известны заранее, но определяются используя ответы на предыдущие запросы.

Вывод

Выведите Q строк. В i-ой строке выведите наибольшего общего предшественника Xi и Yi.

Оценивание

|  |  |  |
| --- | --- | --- |
| Подзадача | Баллы | Ограничения |
| 1 | 10 | T=0, Q=1 |
| 2 | 10 | T=0, N<=1000 |
| 3 | 30 | T=0 |
| 4 | 60 | T=1 |

Примеры

|  |  |
| --- | --- |
| Ввод  3 5 0  1 2 6  7 10  8 5  6 2  9 10  2 3  Вывод  1  5  1  6  1 | Ввод  3 5 1  1 2 6  7 10  8 5  6 2  9 10  2 3  Вывод  1  6  2  1  1 |

Пояснения к примерам

Дерево для обоих примеров показано на рисунке.

Метки вершин в запросах второго примера таковы:

