



Новруз

Залишилось декілька днів до Новруза (Персидського Нового Року) і дідусь запросив свою родину в свій сад. Серед гостей є k дітей. Щоб зробити свято більш веселим для дітей, дідусь збирається провести гру в схованки.

Сад може бути представлений як сітка з $m \times n$ клітинок. Деякі (можливо ноль) клітинок заблоковані камінням, а ті що залишились - називаються *вільними*. Дві клітинки називаються *сусідніми* якщо вони мають спільні сторони. Таким чином, кожна клітинка має до чотирьох сусідніх: дві в горизонтальному напрямку і дві у вертикальному напрямку. Дідусь хоче перетворити свій сад у лабіринт. З цією метою він може заблокувати деякі вільні клітинки посадивши в них кущі. Клітинки, в яких він посадив кущі, перестають бути вільними.

Лабіринт повинен мати наступні властивості. Для кожної пари a і b вільних клітинок в лабіринті має бути тільки один *простий шлях* між ними. Простий шлях між клітинками a і b - це послідовність вільних клітинок, серед яких перша клітинка - a , остання клітинка - b , всі клітинки є різними, і кожні 2 послідовні клітинки є сусідніми.

Дитина може сховатись в клітинці тоді і тільки тоді, якщо ця клітинка є вільною і має *рівно* одну вільну сусідньою клітинку. Двоє дітей не можуть сховатись у тій самій клітинці.

Вам дається карта саду на вході. Ваше завдання - допомогти дідусеві створити лабіринт, в якому може сховатись багато дітей.

Деталі реалізації

Це output-only задача з частковим оцінюванням. Вам дається 10 вхідних файлів, кожен з яких описує сад дідуся. Для кожного вхідного файлу вам потрібно відправити вихідний файл з картою лабіринту. Для кожного вихідного файлу ви отримаєте бали, кількість яких залежить від кількості дітей, для яких є можливість сховатись у лабіринті.

Вам не потрібно відправляти вихідний код для цієї задачі.

Вхідні дані

Кожен вхідний файл описує одну сітку, яка представляє сад, та задає кількість дітей k , запрошених дідусем у такому форматі:

- рядок 1: m n k
- рядок $1 + i$ (для $1 \leq i \leq m$): i -ий рядок сітки, задається текстовим рядком довжини n , що складається з таких символів (без пропусків):

- '.': вільна клітинка,
- '#': камінь.

Вихідні дані

- рядок i (для $1 \leq i \leq m$): i \$-ий рядок лабіринту (сад після того, як у ньому посадили куці). Це текстовий рядок довжини n , що складається з таких символів (без пропусків):
 - '.': вільна клітинка,
 - '#': камінь.
 - 'X': куц. (Врахуйте, що літера X має бути великою)

Обмеження

- $1 \leq m, n \leq 1024$

Оцінювання

Вихідний файл вважається *коректним* якщо він відповідає таким вимогам:

- Вихідна карта повинна співпадати з вхідною картою з єдиним виключенням, що довільна кількість символів '.' у вхідній карті може бути замінена на символи 'X' (клітинки, заблоковані куцями).
- Вихідна карта повинна мати властивості лабіринту, які вказано в умові задачі.

Якщо ваш вихідний файл для тесту не є коректним, то за цей тест ви отримаєте 0. У протилежному випадку ви отримаєте $\min(10, 10 \cdot l/k)$ балів, округлені вниз до двох цифр після коми. Тут l - це кількість дітей, які можуть сховатись у вашому лабіринті, а k - це число, задане у вхідному файлі. Ви отримаєте 10 балів за тест тоді і тільки тоді, коли на виході ви маєте лабіринт, в якому можуть сховатись k або більше дітей. Для кожного тесту існує рішення, за яке можна отримати 10 балів.

Зазначте, що якщо ваш вихідний файл є коректним, але при цьому набирає 0 балів відповідно до зазначеної вище формули, кінцевою оцінкою в CMS буде 'Wrong Answer'.

Приклад

Розглянемо наступні вхідні дані:

```
4 5 5
....#
.#..#
...#.
....#
```

Можливою коректною відповіддю може бути:

```
.X.X#  
.#..#  
...#X  
XX..#
```

Так як у цьому лабіринті можуть сховатись $l = 4$ дітей, цей розв'язок отримає $10 \cdot 4/5 = 8$ балів. Клітинки, в яких дітям можна сховатись, позначені нижче символом \circ :

```
OXOX#  
.#.O#  
...#X  
XX.O#
```

Наступні три приклади виводу не є коректними:

```
.XXX#      ...X#      XXXX#  
.#XX#      .#.X#      X#XX#  
...#.      ...#X      ..X#X  
XX..#      XXXX#      ..XX#
```

В лівому прикладі вихідних даних немає простого шляху між вільною клітинкою у верхньому лівому куті та клітинкою в крайній правій колонці. У двох інших варіантах для кожної пари різних вільних клітинок є рівно два прості шляхи між ними.