



Nowruz

Het is enkele dagen voor Nowruz (Persisch nieuwjaar) en opa heeft de hele familie uitgenodigd in de tuin. Onder de gasten zijn er k kinderen. Om de bijeenkomst leuker te maken voor de kinderen wil opa een spel verstoppertje organiseren.

De tuin kan voorgesteld worden als een rooster van $m \times n$ eenheidscellen. Sommige (mogelijk geen) van de cellen zijn geblokkeerd door rotsen; alle overige cellen noemen we *vrij*. Twee cellen zijn *buren* als ze een zijde delen. Elke cel heeft dus maximaal 4 buren: twee in horizontale richting, en twee in verticale richting. Opa wil zijn tuin in een doolhof omtoveren. Hiertoe kan hij vrije cellen blokkeren door er struiken te planten. De cellen waar hij een struik plant zijn niet langer vrij.

Een doolhof moet de volgende eigenschap hebben: Tussen elke twee vrije cellen a en b is precies één *simpel pad*. Een simpel pad tussen twee cellen is een opeenvolging van vrije cellen waarvan de eerste cel a is, de laatste cel b , alle cellen verschillend zijn en elke twee opvolgende cellen buren zijn.

Een kind kan zich verstoppen in een cel als en slechts als die vrij is en *exact één* vrije buur heeft. Twee kinderen kunnen zich niet in dezelfde cel verstoppen.

Je krijgt een kaart van de tuin als input. Jouw taak is opa te helpen om een doolhof te maken waar veel kinderen zich in kunnen verstoppen.

Implementatiedetails

Dit is een "output-only" taak met partiële scores. Je krijgt 10 inputbestanden, elk met een beschrijving van opa's tuin. Voor elk inputbestand moet je een outputbestand indienen met een kaart van een doolhof. Voor elk outputbestand krijg je punten gebaseerd op het aantal kinderen dat zich in jouw doolhof kan verstoppen.

Je moet geen broncode indienen voor deze taak.

Input

Elk inputbestand beschrijft een tuin en het aantal kinderen k dat opa uitgenodigde. Het formaat is als volgt:

- Lijn 1: $m n k$
- Lijn $1 + i$ (met $1 \leq i \leq m$): rij i van het rooster, gegeven als een string van lengte n , bestaande uit de volgende letters (zonder witruimte):

- '.': een vrije cel,
- '#': een rots.

Output

- Lijn i (met $1 \leq i \leq m$): rij i van het doolhof (de tuin, na het planten van de struiken). Het is een string van lengte n , bestaande uit de volgende letters (zonder witruimte):
 - '.': een vrije cel,
 - '#': een rots,
 - 'X': een struik. (De letter X moet een hoofdletter zijn)

Beperkingen

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

Score

Een outputbestand wordt als *geldig* gezien als aan de volgende voorwaarden wordt voldaan:

- De kaart in het outputbestand moet gelijk zijn aan de kaart in het inputbestand met als enige uitzondering dat een willekeurig aantal . tekens veranderd kan worden in X tekens (cellen geblokkeerd door struiken).
- De outputkaart moet de eigenschappen hebben van een doolhof zoals gedefinieerd in de taakomschrijving.

Als je output voor een test niet geldig is, is je score voor die test 0. Anders is je score gelijk aan $\min(10, 10 \cdot l/k)$ punten, naar beneden afgerond op twee cijfers na de komma. Hierbij is l het aantal kinderen dat zich kan verstoppen in jouw doolhof en k het aantal kinderen gegeven in de input. Je krijgt 10 punten voor een testbestand als en slechts als er k of meer kinderen zich in je doolhof kunnen verstoppen. Voor elke test bestaat er een oplossing die 10 punten waard is.

Bemerk dat als je oplossing geldig is, maar 0 punten verdient volgens bovenstaande berekening, het resultaat van de grader in CMS wordt weergegeven als `Wrong Answer`.

Voorbeeld

Beschouw de volgende input:

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

Hieronder staat een mogelijke geldige output:

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

Sinds $l = 4$ kinderen zich in dit doolhof kunnen verstoppen, krijgt deze oplossing $10 \cdot 4/5 = 8$ punten. De cellen waarin kinderen zich kunnen verstoppen worden hieronder weergegeven met een O:

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

De volgende drie outputs zijn niet niet geldig:

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

In de linkse output is er geen simpel pad tussen de vrije cel linksboven en de vrije cel in de meest rechtse kolom. In de twee andere outputs zijn er voor elk paar van losse vrije cellen precies twee verschillende simpele paden mogelijk tussen elkaar.