



## IOI 2017 Translation

NLD

# Nowruz

Het is enkele dagen voor Nowruz (Persisch nieuwjaar) en opa heeft zijn hele familie uitgenodigd voor een feest in de tuin. Onder de gasten zijn  $k$  kinderen. Om het feest leuker te maken voor de kinderen gaat opa verstopperijtje organiseren.

De tuin kun je zien als een grid van  $m \times n$  eenheidscellen. Sommige (mogelijkerwijs nul) van de cellen zijn geblokkeerd door rotsen, alle overige cellen zijn *vrij*. Twee cellen zijn *buren* als ze een zijde delen: dus elke cel heeft maximaal 4 burens; twee in horizontale richting, en twee in verticale richting. Opa wil zijn tuin in een doolhof veranderen. Om dit te bereiken kan hij enkele 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 elk paar vrije cellen  $a$  en  $b$  is precies één *simpel pad*. Een simpel pad tussen twee cellen is een rij van vrije cellen waarvan de eerste cel  $a$  is, de laatste cel  $b$ , alle cellen verschillend zijn en elke twee opvolgende cellen in de rij burens zijn.

Een kind kan zich verstoppen in een cel dan en slecht dan als die vrij is en *exact één* buurcel heeft. Twee kinderen kunnen zich niet in dezelfde cel verstoppen.

Je krijgt een kaart van de tuin als invoer. Help opa een doolhof te maken waarin zoveel mogelijk kinderen zich kan verstoppen.

## Implementatiedetails

Dit is is output-only taak met partiële scores. Je krijgt 10 invoerbestanden, elk met een beschrijving van opa's tuin. Voor elk invoerbestand moet je een uitvoerbestand inzenden met een kaart van een doolhof. Voor elk uitvoerbestand krijg je punten gebaseerd op het aantal kinderen dat zich kan verstoppen in het doolhof.

Je moet geen broncode insturen voor deze taak.

## Invoer

Elk invoerbestand beschrijft een tuin. Het bevat ook het aantal door opa uitgenodigde kinderen  $k$ . Het formaat is als volgt:

- regel 1:  $m \ n \ k$
- regel  $1 + i$  (met  $1 \leq i \leq m$ ): rij  $i$  van het grid, gegeven door een string van lengte  $n$ , bestaande uit de volgende letters (zonder spaties):
  - '.': een vrije cel,
  - '#': een rots.

## Uitvoer

- regel  $i$  (met  $1 \leq i \leq m$ ): rij  $i$  van het doolhof (de tuin nadat de struiken zijn geplant). Het is een string van lengte  $n$ , bestaande uit de volgende letters (zonder spaties):
  - '.': een vrije cel,
  - '#': een rots,
  - 'X': een struik. (Let op dat de letter X een hoofdletter is.)

## Randvoorwaarden

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

## Scoreverdeling

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

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

Als je uitvoer 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 achter de komma (punt). In deze vergelijking is  $l$  het aantal kinderen dat zich kan verstoppen in jouw doolhof en  $k$  het aantal kinderen gegeven in de invoer. Je krijgt 10 punten voor een testbestand wanneer er  $k$  of meer kinderen zich in je doolhof kunnen verstoppen. Voor elke test is er een oplossing die 10 punten waard is.

Let op dat als je oplossing geldig is, maar 0 punten waard is volgens bovenstaande berekening, de uitvoer van de grader in CMS wordt weergegeven als `Wrong Answer`.

## Voorbeeld

Neem de volgende invoer:

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

```

....#

Hieronder staat een mogelijke geldige uitvoer:

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

Sinds  $l = 4$  kinderen zich in dit doolhof kunnen verstoppen, is deze oplossing  $10 \cdot 4/5 = 8$  punten waard. De cellen waarin kinderen zich kunnen verstoppen worden gegeven met een  $\circ$ :

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

De volgende drie uitvoeren zijn niet geldig:

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

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