



Capodanno persiano

Mancano pochi giorni al Nowruz (il capodanno nel calendario persiano) e il nonno ha invitato tutta la famiglia nel suo giardino. Tra gli ospiti ci sono k bambini. Per rendere più divertente la serata, il nonno ha proposto ai bambini di giocare a nascondino.

Il giardino si può vedere come una griglia $m \times n$ di celle unitarie, in cui alcune celle (zero o più) sono bloccate da delle rocce, mentre le celle rimanenti sono *libere*. Due celle sono *vicine* se hanno un lato in comune, per cui ogni cella ha al massimo 4 vicini: due in orizzontale, e due in verticale. Il nonno ha deciso che per la festa trasformerà il giardino in un labirinto. Per fare questa trasformazione può bloccare alcune celle libere piantandoci dei cespugli: chiaramente, le celle dove il nonno pianta dei cespugli smettono di essere libere.

Il labirinto che il nonno vuole preparare deve possedere la seguente proprietà. Per ogni coppia a e b di celle libere nel labirinto deve esistere esattamente un *cammino semplice* tra di esse. Un cammino semplice tra le celle a e b è una sequenza di celle libere tale che la prima cella è a , l'ultima è b , tutte le celle sono distinte, e ogni due celle consecutive sono vicine.

Un bambino si può nascondere in una cella se e solo se quella cella è libera ed ha *esattamente* un vicino libero. Due bambini non possono nascondersi nella stessa cella.

Data la mappa del giardino, aiuta il nonno a creare un labirinto in cui tanti bambini possano nascondersi.

Dettagli di implementazione

Questo task è di tipo output-only con punteggi parziali. Ti vengono dati 10 file di input, ognuno dei quali descrive una possibile configurazione del giardino del nonno. Per ogni file di input, dovrai sottoporre un file di output con una mappa di un labirinto. Per ognuno di questi file ti verranno assegnati dei punti in base al numero di bambini che possono nascondersi nel tuo labirinto.

Non devi sottoporre alcun codice sorgente per questo task.

Formato di input

Ogni file di input descrive un possibile giardino come griglia e definisce il numero k di bambini invitati alla festa del nonno. Il formato è come segue:

- riga 1: $m \ n \ k$
- riga $1 + i$ ($1 \leq i \leq m$): l' i -esima riga della griglia, cioè una stringa di lunghezza n composta solo dai caratteri che seguono (senza spazi):
 - '.': una cella libera,
 - '#': una roccia.

Formato di output

- riga i ($1 \leq i \leq m$): l' i -esima riga del labirinto (ovvero il giardino dopo aver piantato i cespugli), cioè una stringa di lunghezza n composta dai seguenti caratteri (senza spazi):
 - '.': una cella libera,
 - '#': una roccia,
 - 'X': un cespuglio (nota che la lettera X deve essere maiuscola).

Assunzioni

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

Assegnazione del punteggio

Un file di output viene considerato *valido* se tutte le seguenti condizioni sono soddisfatte:

- La mappa in output corrisponde alla mappa in input con l'unica eccezione che i caratteri '.' nell'input possono essere arbitrariamente cambiati in caratteri 'X' (celle bloccate da cespugli).
- La mappa in output deve soddisfare la proprietà del labirinto, come definito nel testo del problema.

Se il tuo output per un certo testcase non è valido, il tuo punteggio per quel testcase sarà 0. Altrimenti, il punteggio sarà pari a $\min(10, 10 \cdot h/k)$ punti (troncando tale valore a due cifre dopo la virgola), dove h è il numero di bambini che possono nascondersi nel tuo labirinto e k è il valore fornito in input. Quindi, otterrai 10 punti per un testcase se e solo se il tuo output è un labirinto nel quale k o più bambini possono nascondersi. Per ogni testcase esiste una soluzione che totalizza 10 punti.

Nota che se la tua soluzione è valida ma totalizza 0 punti secondo la suddetta formula, il verdetto mostrato dal server nei dettagli di correzione sarà ugualmente 'Wrong Answer'.

Esempio

Consideriamo l'input seguente:

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

Qui sotto vediamo un possibile output valido:

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

Dato che $h = 4$ bambini riescono a nascondersi in questo labirinto, questa soluzione riceverà $10 \cdot 4/5 = 8$ punti. Qui sotto vediamo le celle in cui i bambini possono nascondersi marcate con una lettera \circ :

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

I seguenti tre output invece non sono validi:

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

Nell'output di sinistra non esiste un cammino semplice tra la cella libera nell'angolo in alto a sinistra e la cella libera nella colonna più a destra. Negli altri due output, per ogni coppia di celle libere distinte, esistono esattamente due cammini semplici distinti che le connettono.