



Nowruz

Čez nekaj dni bo Nowruz (perzijsko novo leto) in dedek se je odločil, da povabi vso družino na svoj vrt. Med gosti je k otrok. Da bi otrokom popestril srečanje, se je odločil, da bo pripravil teren za skrivalnice.

Vrt predstavimo z mrežo velikosti $m \times n$ enotskih polj. Nekatera polja (lahko tudi nobeno) so blokirana s skalami, ostala polja pa imenujmo *prosta*. Dve polji imenujemo *sosednji*, če si delita rob. Torej ima vsako polje 4 sosede. Dva vodoravna in dva navpična. Dedek želi spremeniti vrt v blodnjak. Zato je blokiral nekaj polj tako, da je na njih zasadil grmovje. Polja na katerih je zasadil grmovje, niso več prosta.

Blodnjak, ki ga bo postavil mora ustrezati naslednjim pravilom. Za vsak par prostih polj a in b mora obstajati natanko ena *preprosta pot* med njima. Preprosta pot med polji a in b je zaporedje prostih polj med katerimi je prvo polje a in zadnje b , vsa vmesna polja pa morajo obstajati, dve zaporedni polji pa morata biti sosednji.

Otrok se lahko skriva na polju če in samo če je polje prosto in ima *natanko* enega soseda. Na enem polju se ne moreta hkrati skriti dva otroka.

Na vhodu je podan zemljevid vrta. Tvoja naloga je, da dedku pomagaš postaviti blodnjak, v katerem se lahko skriva mnogo otrok.

Podrobnosti implementacije

To je naloga tipa zgolj-izhod z delnim točkovanjem. Podanih je 10 vhodnih datotek, v vsaki pa je opisan dedkov vrt. Za vsako vhodno datoteko moraš oddati izhodno datoteko z zemljevidom blodnjaka. Za vsako vhodno datoteko prejmeš točke, ki zavisijo od števila otrok, ki se lahko skrivajo v blodnjaku.

Za to nalogo ti ni potrebno oddati nobene izvorne kode.

Oblika vhoda

Vsaka vhodna datoteka opisuje mrežo, ki predstavlja vrt in podaja število otrok k , ki jih je dedek povabil. Oblika je sledeča:

- 1. vrstica: m n k
- $1 + i$ -ta vrstica (za $1 \leq i \leq m$): i -ta vrstica mreže, ki je podana kot niz dolžine n , sestoji iz sledečih znakov (brez presledkov):
 - '.': prosto polje,
 - '#': skala.

Oblika izhoda

- i -ta vrstica (za $1 \leq i \leq m$): i -ta vrstica blodnjaka (vrta, ko so grmovja že zasajena). To je niz dolžine n , ki sestoji iz sledečih znakov (brez presledkov):
 - '.': prosto polje,
 - '#': skala,
 - 'x': grm. (Znak X mora biti velika tiskana črka.)

Omejitve

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

Ocenjevanje

Izhodna datoteka je *veljavna*, če je zadoščeno vsem sledečim pogojem:

- Izhodni zemljevid se mora ujemati z vhodnim, z izjemo končno mnogo '.' znakov, ki so v izhodni datoteki spremenjeni v 'x' (polja kjer so grmovja).
- Izhodni zemljevid mora ustrezati lastnostim blodnjaka, kot je opisano v besedilu naloge.

Če je izhodna datoteka neveljavna, bo testni primer ocenjen z 0 točkami. Sicer bo ocena $\min(10, 10 \cdot l/k)$ točk, zaokroženo navzdol na dve decimalni mesti. Tukaj l označuje število otrok, ki se lahko skrijejo v blodnjaku, k pa je število podano na vhodu. 10 točk prejmete, če in samo če se v izhodni zemljevid blodnjaka lahko skrije k ali več otrok. Za vsak testni primer obstaja rešitev, ki prinaša 10 točk.

Vedi, da se lahko veljavni rešitvi po zgornji formuli dodeli tudi 0 točk. V tem primeru vam ocenjevalni strežnik vrne 'Wrong Answer'.

Primer

Oglejmo si sledeči primer:

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

Spodaj je veljavna rešitev:

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

Ker je $l = 4$ otrok, ki se lahko skrijejo v blodnjaku, rešitev prejme $10 \cdot 4/5 = 8$ točk. Polja, na katera se lahko skrijejo otroci, so spodaj označena z 0:

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

Spodnje rešitve so neveljavne:

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

Na levem izhodu ni preproste poti do prostega polja v skrajnem desnem stolpcu. V preostalih dveh izhodih obstajata dve različni preprosti poti med dvema različnima poljema.