

# Vakond

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:           **1 second**  
Memory limit:        **256 megabytes**

Egy nehéz kutatás után Anton úgy döntött, hogy kikapcsolódik a vidéki házában. Gyönyörű kertje van sok különböző virággal. De, ó ne, megérkezésekor jelentős számú lyukat látott a földön. Ez egy vakond!

Most, egy ásóval felfegyverkezve, Anton várni fogja a vakondot. A vakond bármelyik lyukból előbukkanhat. Anton olyan pozíciót szeretne választani, hogy a legrosszabb esetben a lehető **minimális** idő alatt fussa le a vakondot.

A kertet egy  $n \times m$  mátrixként lehet reprezentálni, ahol  $n$  a sorok száma, és  $m$  az oszlopok száma. A sorokat felülről lefelé, 1-től  $n$ -ig számozzuk. Az oszlopokat balról jobbra, 1-től  $m$ -ig számozzuk. Így a  $(1; 1)$  indexű cella a bal felső sarokban található.

A kert minden cellája,  $a_{i,j}$ , leírja a cella állapotát:

- $a_{i,j} = \text{"."}$  — ez a cella nem tartalmaz virágokat vagy lyukakat;
- $a_{i,j} = \text{"F"}$  — ez a cella virágokat tartalmaz;
- $a_{i,j} = \text{"H"}$  — ez a cella lyukat tartalmaz.

Anton tudja, hogy a lyukak száma nem haladja meg a 100-at.

Mint olyan, aki sok időt fektetett ezekbe a virágokba, a szíve nem bírja elviselni, hogy letapossa azokat. Ezért olyan utat kell találnia, amely nem halad át rajtuk.

Bármikor Anton át tud menni a  $(x, y)$  pozícióból az alábbi pozíciók egyikére:  $(x-1, y)$ ,  $(x+1, y)$ ,  $(x, y-1)$ ,  $(x, y+1)$ , feltéve, hogy az új pozíció nem tartalmaz virágokat, és a kertben van.

Keressük meg az összes olyan pozíciót  $(x, y)$ , ahonnan Anton a legrosszabb esetben a lehető legkevesebb idő alatt fog futni a vakondhoz.

## Input

Az első sor két egész számot tartalmaz,  $n, m$  ( $1 \leq n \cdot m \leq 2 \cdot 10^5$ ) — a kert hosszát és szélességét.

A következő  $n$  sor mindegyike  $m$  karaktert tartalmaz — a kert leírása.

Biztosított, hogy minden olyan cellából, amely nem tartalmaz virágokat, elérhető másik olyan cella, amely nem tartalmaz virágokat, a cellákon keresztüli mozgással, amelyek nem tartalmaznak virágokat.

Biztosított, hogy legalább egy lyuk van, és a lyukak száma a kertben nem haladja meg a 100-at.

## Output

Az első sorban egyetlen egész számot kell kiírni,  $x$  ( $1 \leq x \leq n \cdot m$ ) — az optimális pozíciók száma.

Minden következő  $x$  sorban ki kell írni az optimális pozíciókat  $(x; y)$  a vakondra várásra ( $1 \leq x \leq n$ ,  $1 \leq y \leq m$ ).

A pozíciókat bármilyen sorrendben ki lehet írni.

## Scoring

Legyen  $k$  a lyukak száma a kertben.

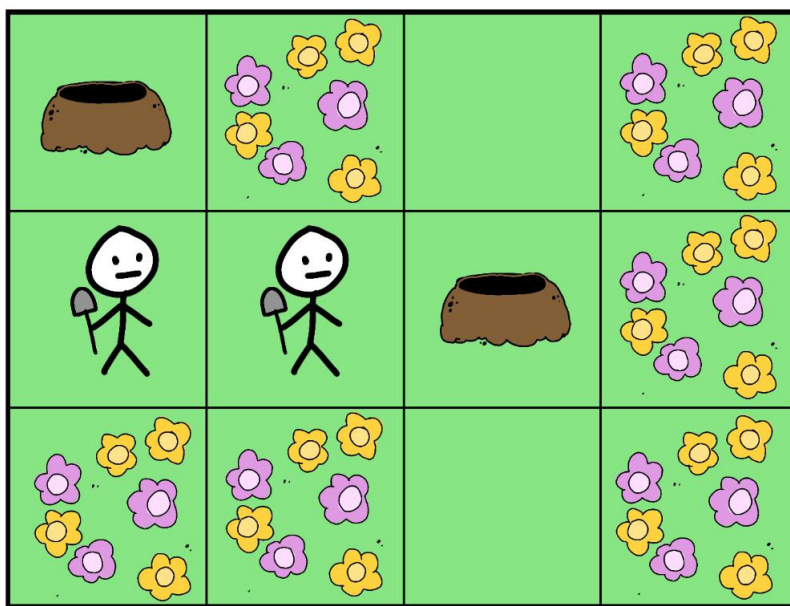
1. (6 pont):  $n = 1, m = 2$ ;
2. (9 pont):  $n = 1$ ;

3. (15 pont):  $k = 1, n \cdot m \leq 5 \cdot 10^3$ ;
4. (22 pont):  $n \cdot m \leq 5 \cdot 10^3$ ;
5. (17 pont):  $k = 1$ ;
6. (31 pont): további korlátozások nélkül.

## Examples

standard input	standard output
3 4 HF.F ..HF FF.F	2 2 1 2 2
4 9 .....FFH .F..FHFF. HF..... .FHF..FFF	2 1 6 3 4

## Note



Fent látható az első példa és az optimális várakozási pozíciók vannak megjelölve.