

## Problema Atentie

Fisier intrare: `standard input`  
Fisier iesire: `standard output`

Argintolac și Bronzolac concurează într-un joc pe o tablă cu  $N$  linii și  $M$  coloane care conține obstacole. Jocul se desfășoară în următorul fel:

- Dealerul Nitsoc alege un șir *nevid* de caractere  $S$  ce conține doar literele 'U', 'D', 'L', 'R'.
- Argintolac alege o celulă liberă (care nu conține vreun obstacol) a tablei și plasează un robot în aceasta.
- Bronzolac reordonează (sau nu) caracterele șirului  $S$ .
- Robotul parcurge traseul descris de șirul dat de Bronzolac, deplasându-se la pasul  $i$  în celula adiacentă din direcția reprezentată de caracterul  $S_i$  ('U' - în sus, 'D' - în jos, 'L' - în stânga, 'R' - în dreapta).
- Argintolac câștigă dacă robotul ajunge teafăr în celula finală (acesta nu iese de pe tablă și nu se lovește de niciun obstacol pe parcurs), altfel, Bronzolac câștigă.

Înainte de începerea jocului, dealerul Nitsoc primește o atenție de la Argintolac și se întreabă care este numărul de șiruri  $S$  pe care le poate alege inițial astfel încât Argintolac să câștige. *Fiind foarte competitiv, atât Argintolac, cât și Bronzolac joacă optim.* Din moment ce Nitsoc și-a petrecut ultimii 5 ani la Master...Club, el nu este capabil să rezolve problema singur așa că vă cere ajutorul. Sa se afișeze răspunsul modulo  $10^9 + 7$ .

### Date de intrare

Pe primul rând se găsesc 2 numere întregi pozitive  $N$  și  $M$ , reprezentând numărul de linii, respectiv numărul de coloane al tablei.

Pe următoarele  $N$  linii este descrisă configurația tablei, linia  $i$  conținând  $M$  caractere de tipul '.' (reprezentând o celulă liberă) sau '#' (reprezentând o celulă ce conține un obstacol).

### Date de ieșire

Se va afișa pe primul rând un număr întreg reprezentând numărul de șiruri nevide  $S$  care pot fi alese inițial astfel încât Argintolac să câștige, modulo  $10^9 + 7$ .

### Restricții

- $1 \leq N, M \leq 2000$
- Tabla conține cel puțin o celulă liberă.

### Subtask 1 (29 puncte)

- $N = 1$

### Subtask 2 (8 puncte)

- $1 \leq N, M \leq 3$

### Subtask 3 (13 puncte)

- $1 \leq N, M \leq 10$

### Subtask 4 (9 puncte)

- $1 \leq N, M \leq 50$

### Subtask 5 (11 puncte)

- $1 \leq N, M \leq 150$

### Subtask 6 (16 puncte)

- $1 \leq N, M \leq 1000$

### Subtask 7 (14 puncte)

- Fără restricții suplimentare

### Exemple

stdin	stdout
2 4 ##.. .##.	4
2 2 .. ..	12
4 4 .... .#.. .... #...	148

## Explicații

Pentru primul exemplu,

- Dacă Nitsoc alege sirul R, atunci Argintolac alege poziția (1,3) și nu contează dacă Bronzolac reordonează sirul.
- Dacă Nitsoc alege sirul L, atunci Argintolac alege poziția (1,4) și nu contează dacă Bronzolac reordonează sirul.
- Dacă Nitsoc alege sirul D, atunci Argintolac alege poziția (1,4) și nu contează dacă Bronzolac reordonează sirul.
- Dacă Nitsoc alege sirul U, atunci Argintolac alege poziția (2,4) și nu contează dacă Bronzolac reordonează sirul.

Pentru al doilea exemplu,

- Dacă Nitsoc alege sirurile R, D, RD sau DR, atunci Argintolac alege poziția (1,1) și castiga indiferent de cum le poate reordona Bronzolac.
- Dacă Nitsoc alege sirurile L, D, LD sau DL, atunci Argintolac alege poziția (1,2) și castiga indiferent de cum le poate reordona Bronzolac.
- Dacă Nitsoc alege sirurile R, U, RU sau UR, atunci Argintolac alege poziția (2,1) și castiga indiferent de cum le poate reordona Bronzolac.
- Dacă Nitsoc alege sirurile L, U, LU sau UL, atunci Argintolac alege poziția (2,2) și castiga indiferent de cum le poate reordona Bronzolac.

A se observa că ne interesează numărul de siruri distincte, deci cele formate dintr-un singur caracter care sunt aplicabile în mai multe poziții nu se vor număra de mai multe ori.