



33^{ος}

ΠΑΝΕΛΛΗΝΙΟΣ ΔΙΑΓΩΝΙΣΜΟΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΘΕΜΑ Β΄ ΦΑΣΗΣ ΛΥΚΕΙΟΥ

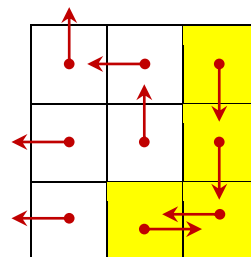
Ο ΔΙΚΑΙΟΣ ΛΑΒΥΡΙΝΘΟΣ

Σε ένα τηλεπαιχνίδι, οι παίκτες τοποθετούνται σε ένα λαβύρινθο από όπου πρέπει να βγουν. Ο λαβύρινθος αποτελείται από $N \times M$ δωμάτια σε διάταξη ορθογωνίου. Σε κάθε δωμάτιο υπάρχει ένας γρίφος και, αν ο παίκτης τον λύσει, τότε ανοίγει μία πόρτα που οδηγεί σε ένα συγκεκριμένο γειτονικό δωμάτιο (πάνω, κάτω, αριστερά ή δεξιά). Αν ένα δωμάτιο βρίσκεται στο περίγραμμα του λαβύρινθου και ανοίξει μία πόρτα που οδηγεί προς τα έξω, τότε ο παίκτης βγαίνει και κερδίζει.

Οι διαστάσεις του λαβύρινθου και η θέση της πόρτας εξόδου για κάθε δωμάτιο είναι γνωστές εξ αρχής — σε κάθε δωμάτιο, η πόρτα είναι σημειωμένη με ένα από τα γράμματα "U" (up), "D" (down), "L" (left) ή "R" (right). Άρα, αν ένας παίκτης ξεκινήσει από κάποιο αρχικό δωμάτιο και λύνει κατά σειρά όλους τους γρίφους στα δωμάτια που επισκέπτεται, η διαδρομή που θα ακολουθήσει είναι συγκεκριμένη.

Δυστυχώς όμως για τους παίκτες, είναι πιθανό οι πόρτες εξόδου να είναι έτσι τοποθετημένες ώστε από κάποια αρχικά δωμάτια να μην είναι δυνατή η έξοδος από το λαβύρινθο! Δείτε το παρακάτω παράδειγμα. Τα βέλη στο δεξιό μέρος του σχήματος δείχνουν την πορεία των παικτών από κάθε δωμάτιο.

U	L	D
L	U	D
L	R	L



Αν ένας παίκτης ξεκινήσει από κάποιο αρχικό δωμάτιο που είναι σημειωμένο με κίτρινο χρώμα, τότε θα κολλήσει σε μια κυκλική διαδρομή και δε θα καταφέρει να βγει ποτέ από το λαβύρινθο, όσους γρίφους και αν λύσει. Αντίθετα, από τα υπόλοιπα δωμάτια θα καταφέρει να βγει, αν φυσικά λύσει τους γρίφους που απαιτούνται.



Οι παρουσιαστές του παιχνιδιού θέλουν να είναι δίκαιο το τηλεπαιχνίδι τους, δηλαδή τα αρχικά δωμάτια από τα οποία οι παίκτες δεν μπορούν να κερδίσουν όσους γρίφους και αν λύσουν να είναι λίγα. Βοηθήστε τους να τα μετρήσουν!

Πρόβλημα

Να αναπτύξετε ένα πρόγραμμα σε μια από τις γλώσσες του IOI (PASCAL, C, C++, Java) το οποίο θα διαβάσει το λαβύρινθο και θα εκτυπώνει το πλήθος των αρχικών δωματίων από τα οποία οι παίκτες δεν μπορούν να κερδίσουν.

Αρχεία εισόδου:

Τα αρχεία εισόδου με όνομα **fairmaze.in** είναι αρχεία κειμένου με την εξής δομή: Στην πρώτη γραμμή υπάρχουν δύο ακέραιοι αριθμοί **N** και **M**, οι διαστάσεις του λαβυρίνθου (γραμμές και στήλες, αντίστοιχα). Κάθε μία από τις επόμενες **N** γραμμές παριστάνει μία γραμμή του λαβυρίνθου και περιέχει ακριβώς **M** χαρακτήρες, κάθε ένας από τους οποίους είναι ένα από τα γράμματα "U", "D", "L" ή "R". Το γράμμα που αντιστοιχεί σε κάθε δωμάτιο συμβολίζει τη θέση της πόρτας εξόδου του δωματίου.

Αρχεία εξόδου:

Τα αρχεία εξόδου με όνομα **fairmaze.out** είναι αρχεία κειμένου με την εξής δομή. Περιέχουν ακριβώς μία γραμμή που περιέχει έναν ακέραιο αριθμό: το πλήθος των αρχικών δωματίων από τα οποία οι παίκτες δεν μπορούν να κερδίσουν, όσους γρίφους και αν λύσουν.

Περιορισμοί:

- $1 \leq N \leq 1.000$
- $1 \leq M \leq 1.000$

Παραδείγματα αρχείων εισόδου-εξόδου:

fairmaze.in	fairmaze.out
3 3 ULD LUD LRL	4

Εξήγηση:

Το παράδειγμα είναι αυτό του παραπάνω σχήματος. Υπάρχουν 4 δωμάτια (σημειωμένα στο σχήμα με κίτρινο χρώμα) από τα οποία οι παίκτες δεν μπορούν να κερδίσουν.



Παρατηρήσεις:

Μορφοποίηση: Στην είσοδο αλλά και στην έξοδο, κάθε γραμμή τερματίζει με έναν χαρακτήρα newline.

Μέγιστος χρόνος εκτέλεσης: 1 sec.

Μέγιστη διαθέσιμη μνήμη: 64 MB.

Επικεφαλίδες στον πηγαίο κώδικα: Στην αρχή του πηγαίου κώδικά σας, θα πρέπει να χρησιμοποιήσετε τις παρακάτω επικεφαλίδες.

```
(* USER: username  
LANG: PASCAL  
TASK: fairmaze *)
```

για κώδικα σε PASCAL

```
/* USER: username  
LANG: C  
TASK: fairmaze */
```

για κώδικα σε C

```
/* USER: username  
LANG: C++  
TASK: fairmaze */
```

για κώδικα σε C++

```
/* USER: username  
LANG: Java  
TASK: fairmaze */
```

για κώδικα σε Java