

Shopping

The old tube screen to your computer turned out to be the cause of your chronic headaches. You therefore decide to buy one of these new flat TFT monitors. At the entrance of the computer shop you see that it is quite full with customers.



In fact, the shop is rather packed with customers and moving inside involves a certain amount of elbowing. Since you want to return home quickly to complete your half finished SPOJ tasks, you want to sidestep the crowd as much as possible. You examine the situation somewhat closer and realise that the crowding is less in some parts of the shop. Thus, there is reason for hope that you can reach your goal in due time, provided that you take the shortest way. But which way is the shortest way?

You sketch the situation on a piece of paper but even so, it is still a tricky affair. You take out your notebook from your pocket and start to write a program which will find the shortest way for you.

Input

The first line of the input specifies the width w and height h of the shop. Neither dimension exceeds 25.

The following h lines contain w characters each. A letter X symbolises a shelf, the letter S marks your starting position, and the letter D marks the destination (i.e. the square in front of the monitors). All free squares are marked with a digit from 1 to 9, meaning the number of seconds needed to pass this square.

There are many test cases separated by an empty line. Input terminates with width and height equal 0 0.

Output

Your program is to output the minimum number of seconds needed to reach to destination square. Each test case in a separate line. Movements can only be vertical and horizontal. Of course, all movements must take place inside the grid. There will always be a way to reach the destination.

Example

Sample input:

```
4 3
X1S3
42X4
X1D2
```

```
5 5
S5213
2X2X5
51248
4X4X2
```

1445D

0 0

Sample output:

4

23