

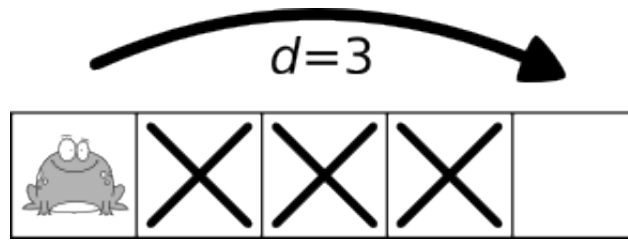
UCF Local Programming Contest — August 30, 2014

Jumping Frog

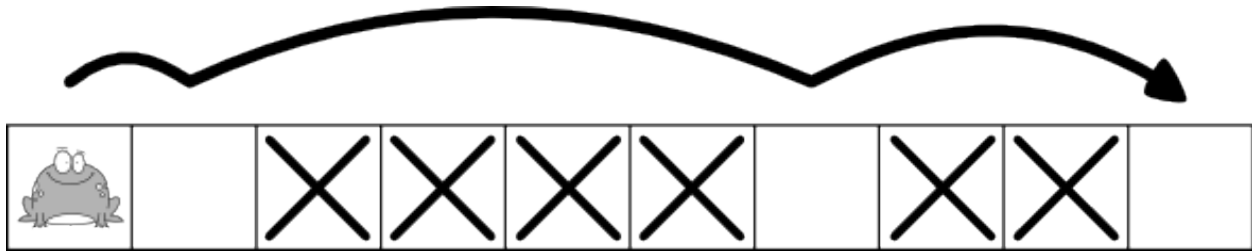
filename: frog

(*Difficulty Level:* Easy)

Freddy the frog is trying to go get a bite to eat. The problem is Freddy lives far away from all the restaurants he likes. Freddy is a capable frog, able to jump over a large number of cells. Unfortunately, on a given day, Freddy may be too hungry to jump very far. On a given day, he can only jump over at most d cells (note that he can also jump over fewer cells).



Another complication for Freddy is the path he jumps across is always under construction. Some of the cells are blocked off! Freddy doesn't want to land in a cell under construction but is allowed to jump over them.



Freddy starts off in the first cell and must travel to the last cell where his destination restaurant resides. He would like to know if he can reach the last cell and how quickly he can reach it. Freddy always jumps towards his destination.

The Problem:

Given a description of cells, determine the minimum number of jumps Freddy can make to reach the restaurant.

The Input:

The first input line contains a positive integer, n , indicating the number of days to check. Each day is represented by two lines. The first line of each day contains two integers, c ($2 \leq c \leq 50$) and d ($0 \leq d \leq 50$), representing (respectively) the number of cells in the path from Freddy's home to the restaurant (including his home and the restaurant) and the maximum number of cells Freddy can jump over in a single jump on that day. The second line is a string consisting of c characters containing only '.' and 'X' characters where '.' represents that the cell is okay for

Freddy to occupy and 'X' represents that the cell is blocked by construction. The first and last characters of the string represent Freddy's home and the restaurant, respectively; these two locations will never be blocked.

The Output:

For each day, first output the heading "Day # d ", where d is the day number, starting with 1. Then, print the input values exactly as they appear in the input. Following the header info, output the minimum number of jumps it takes Freddy to reach the restaurant. If it is not possible to reach the restaurant on a given day, print the number 0 instead.

Leave a blank line after the output for each data set. Follow the format illustrated in Sample Output.

Sample Input:

```
4
8 3
.XX.X.X.
3 50
...
8 1
...XX...
10 4
..XXXX.XX.
```

Sample Output:

```
Day #1
8 3
.XX.X.X.
2

Day #2
3 50
...
1

Day #3
8 1
...XX...
0

Day #4
10 4
..XXXX.XX.
3
```