# stringpath

You are given an $N \cdot M$ grid with characters from $A - Z$. Print the path to be taken to get lexicographically smallest string when moving from top-left to bottom-right by moving either to RIGHT(R) or DOWN(D) and not visiting any cell twice. If there are multiple such paths, print the lexicographically smallest of them.

click the link if you don't know what lexicographic means.

Note: use fast io as input files could be as large as 15MB

**Input Format**

The first line contains $T$, the number of testcases.

The first line of each testcase contains two integers, $N$ and $M$.

The next $N$ lines contain $M$ characters, denoting the characters of that row.

**Constraints**

$1 \leq T \leq 500$

$1 \leq N, M \leq 500$

$\textbf{The total number of characters in a file will not exceed } 15 \cdot 10^7$

**Output Format**

For each testcase print a single string denoting the lexicographically smallest path which when followed can make the lexicographically smallest string from $0, 0$ moving to $n - 1, m - 1$

**Sample Input 0**

```
3
3 3
AAA
AAA
AAA
3 4
DSCG
ZFQL
LMEV
2 5
TSNMX
SRCVK
```

**Sample Output 0**

```
Case #1: DDRR
Case #2: RRRDD
Case #3: RRDRR
```

**Explanation 0**

/

For the 1st testcase, The smallest string is AAAAA, as there are multiple paths to achieve this, we print the lexicographically smallest of them, DDRR. For the 2nd testcase, The smallest string is DSCGLV.

/