Count weighted paths

John likes to take a walk from his house to university. He needs to arrive his university in at most **T** seconds after leaving his home. We can represent the situation as a **N** vertices graph. Vertex 0 of the graph will be John's home and vertex 1 John's university. There can be bidirectional roads connecting pairs of vertices, each road will take John some seconds to cross.

John likes variety. We consider a valid path to be a sequence of vertices that starts with vertex 0 (John's house) and finishes with vertex 1 (The University) and there exists a road connecting each pair of consecutive vertices in the sequences (Note that a vertex may appear multiple times in the path). The total time John needs to traverse a path is equal to the sum of the times needed to cross each individual road in it. Please count the total number of different paths that need at most **T** minutes to be traversed in total. Two paths are different if there is at least one moment at which they visit different vertices.

Given **T**, **N** and the roads between the vertices, ¿How many different paths that need at most **T** seconds exist? Print the result modulo 1000000007 (10^9+7).

Input

The first line consists of a integer **TOTAL**, the total number of test cases (1 <= **N** <= 10).

Each of the following test cases begins with a single line that contains two integers : **N** and **T**. (2 <= **N** <= 5), (1 <= **T** <= 1000000000 (10^9)).

The **N** following lines are indexed from **i**=0 to **N**-1. The **i-th** line will represent the roads that connect vertex **i** with other vertices. The line will consist of **N** character indexed from **j=**0 to **N**-1**.** The **j**-th character of the **i**-th line represents the road connecting vertex **i** with vertex **j**. If the character is '-', this means no road connectes vertices **i** and **j**. Otherwise, the character will be a digit equal to 1,2 or 3, determining the number of minutes it takes John to move between vertices **i** and **j**.

For every pair (**i**,**j**), the road character between **i** and **j** will be the same as the one between **j** and **i**.

For each **i**, there will never be a road cannecting vertex **i** with itself.

Vertex 0 represents John's house and Vertex 1 John's university.

Output

For each test case, show in a single line: "Case #i: R", where R is the total number of valid paths between vertices 0 and 1 donde R that need a quantity of at most **T** segundos .

Example

Input:
3
2 9
-3
3-
5 4
--123
--123
11---
22---
33---
3 100
-21
2-3
13-

Output:
Case #1: 2
Case #2: 4
Case #3: 924247768

Notes

There are two paths in the first case that need 9 minutes or less:

- 0 -> 1 (3 minutes)
- 0 -> 1 -> 0 -> 1 (9 minutes)

The second case contains 4 paths that need at most 4 minutes to be traversed:

- 0 -> 2 -> 1 (2 minutes)
- 0 -> 3 -> 1 (4minutes)
- 0 -> 2 -> 0 -> 2 -> 1 (4 minutes)
- 0 -> 2 -> 1 -> 2 -> 1 (4minutes)

0 -> 4 -> 1 is a path that needs 6 minutes.