

10802 Lex Smallest Drive

“OK. Let’s go right.”
 Bartholomew Furrow

A *graph*, G , consists of a finite set of *vertices*, V , and a set of *edges*, E , where each edge is a set of 2 vertices $\{u, v\}$. A *walk* in G is a finite sequence of vertices (v_1, v_2, \dots, v_k) , such that for each pair (v_{i-1}, v_i) for i in $[2, k]$, $\{v_{i-1}, v_i\}$ is in E . This is called a “walk from v_1 to v_k ”. If V is a set of integers, then any two walks in G can be compared lexicographically; for example, the walk $(3, 5, 6, 2, 8)$ is smaller than the walk $(3, 5, 6, 5, 7)$. A walk, W , from a to b is *lexicographically smallest* if there is no other walk from a to b in G that is smaller than W . A *drive* is a walk (v_1, v_2, \dots, v_k) , where no edge is used twice consecutively. That is, for all i from 2 up to $k - 1$, v_{i-1} is not equal to v_{i+1} .

Given G and a start vertex, s , your task is to find the lexicographically smallest drives from s to each vertex in G .

Input

The first line of input gives the number of cases, N . N test cases follow. Each one starts with a line containing the integers n, m and s . ($0 \leq n \leq 100, 0 \leq m \leq 4950$). The next m lines will list the edges of G . V is the set $\{0, 1, \dots, n - 1\}$. s is in V .

Output

For each test case, output the line ‘Case # x :’, where x is the number of the test case. Then print n lines, line i listing the lexicographically smallest drive from s to i using single spaces to separate consecutive vertices. If there is no such drive, print ‘No drive.’ Put an empty line after each test case.

Sample Input

```
2
6 4 5
5 0
2 5
4 0
3 1
4 4 0
0 1
1 2
0 2
0 3
```

Sample Output

```
Case #1:
5 0
No drive.
5 2
No drive.
5 0 4
```

5

Case #2:

0

0 1

0 1 2

No drive.