Transitive Closure

In mathematics, a set S is transitive if whenever an element a is related to an element b, and b is in turn related to an element c, then a is also related to c. In other words, any set of pairs is transitive if and only if you have (a,b) and (b,c) then you also must have (a,c). Check this example: S = { (1,2),(2,3),(3,4),(2,4) }. Is set S is transitive relation ? No, Because you have (1,2) and (2,3) but you don’t have (1,3) If we add (1,3) will it be transitive ? ((S = { (1,2),(2,3),(3,4),(2,4),(1,3) })) No, Because you have (1,3) and (3,4) but you don’t have (1,4) If we add (1,4) will it be transitive ? ((S = { (1, 2),(2,3),(3,4),(2,4),(1,3),(1,4) })) Yes, Now the set S is now transitive after we added 2 pairs { (1,3),(1,4) } These pairs called transitive closure (which means the minimal pairs that convert set S into a transitive set). Your task is given the set S you have to output the minimal pairs have to be added to make the set S transitive.

Input

The first line of input is the number of test cases T where (0<T<=100), Each test case you'll be given the number of pairs in the set N where (0<N<=100), followed by N pairs (a, b) where (0<=a,b<N).

Output

For each test case print "Case_#i:_X" where "i" is the case number, "X" is the minimal number of pairs have to be added to make the set transitive and "_" is a white space. Each test case should be in a separate line.

Example

Input:

```
3
4
0 1
1 2
2 3
1 3
2
0 1
1 0
1
0 0
```

Output:

```
Case #1: 2
Case #2: 2
Case #3: 0
```