

4246 Tanks a Lot

Imagine you have a car with a very large gas tank — large enough to hold whatever amount you need. You are traveling on a circular route on which there are a number of gas stations. The total gas in all the stations is exactly the amount it takes to travel around the circuit once. When you arrive at a gas station, you add all of that station's gas to your tank. Starting with an empty tank, it turns out there is at least one station to start, and a direction (clockwise or counter-clockwise) where you can make it around the circuit. (On the way home, you might ponder why this is the case — but trust us, it is.)

Given the distance around the circuit, the locations of the gas stations, and the number of miles your car could go using just the gas at each station, find all the stations and directions you can start at and make it around the circuit.

Input

There will be a sequence of test cases. Each test case begins with a line containing two positive integers c and s , representing the total circumference, in miles, of the circle and the total number of gas stations. Following this are s pairs of integers t and m . In each pair, t is an integer between 0 and $c-1$ measuring the clockwise location (from some arbitrary fixed point on the circle) around the circumference of one of the gas stations and m is the number of miles that can be driven using all of the gas at the station. All of the locations are distinct and the maximum value of c is 100,000. The last test case is followed by a pair of '0's.

Output

For each test case, print the test case number (in the format shown in the example below) followed by a list of pairs of values in the form ' $i d$ ', where i is the gas station location and d is either 'C', 'CC', or 'CCC', indicating that, when starting with an empty tank, it is possible to drive from location i around in a clockwise (C) direction, counterclockwise (CC) direction, or either direction (CCC), returning to location i .

List the stations in order of increasing location.

Sample Input

```
10 4
2 3 4 3 6 1 9 3
5 5
0 1 4 1 2 1 3 1 1 1
0 0
```

Sample Output

```
Case 1: 2 C 4 CC 9 C
Case 2: 0 CCC 1 CCC 2 CCC 3 CCC 4 CCC
```