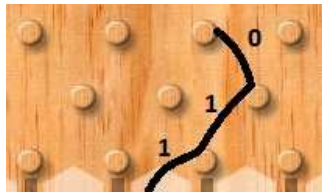


Pachinko Madness

Have you ever heard of Pachinko? There are probably many names for it, but the game is very simple: Drop a ball, a disc, or some object into a thin box. The box is filled with pegs, which bounce the ball left or right. There are usually some buckets at the bottom of the box with different numbers on them, and if your ball lands in that bucket, then you get the number as your points.



This version has a slightly different variation. Your score is not based on any buckets: it is based on the path the ball takes down the box! The path is denoted as a simple binary string, where a 0 means the ball bounced right, and a 1 means the ball bounced left. Each bounce gets appended to the front of the path's string, and then your points are calculated by converting the binary string into decimal representation. For example: Your ball is dropped, and it bounces right, then left, then left again before falling out of the box. This means your binary string is 110, which is 6 in decimal, so you earn 6 points.



Being the genius that you are, you have found a way to manipulate the ball, such that you can control where it bounces. Obviously, the judges of the game found out your strategy and implemented a new kind of box, one where there are walls connecting pegs, such that your ball *must* bounce in one of the directions, or possibly get stuck (this means you get no points)! Now it is time to gain the "advantage" again by determining just which way your ball should bounce to get the most points.

Input

Input begins with a single non-negative integer, B , the number of boxes to score. Each box will begin with a single line containing a single positive integer P , the number of pegs in the box. Each peg is numbered from 0 to $P - 1$. The following P lines will describe the pegs, starting at peg 0, followed by peg 1, and so on. Each peg line will have 2 space separated integers. The first integer denotes which peg the ball will go to if it bounces left. The second integer denotes which peg the ball will go to if it bounces right. If the number is -1, then you cannot bounce in that direction. If the number is P , then the ball will fall out of the box. The game will always start at peg 0, and the ball will always be moving downward, so the ball will never be able to revisit a peg.

Output

For each box, output the maximal score you can achieve in the box on a single line, in decimal representation.

(Sample Input/Output on next page)

Sample Input

4
6
1 2
3 4
4 5
6 6
6 6
6 6
3
-1 -1
2 2
3 3
7
1 4
-1 2
-1 3
-1 7
5 -1
6 -1
7 7
5
3 2
5 5
4 -1
1 4
5 5

Sample Output

7
0
14
7