Checkers

Lea and Bea love to play checkers. Since the game got boring of time they built a giant checkers board, but playing the game with loads of pieces got boring, too. Thus, they decided to create a new game that is played like this: One player plays with the white pieces, the other player takes the black pieces. Per row of the game board one piece of every color is randomly placed. It is never allowed that two pieces are placed on the same field of the game board.

The game itself is played in turns, white always goes first. A turn consists of moving a piece of your own color in the row it is placed in. It can be moved as far to the left or right as possible, but it may never be moved to a field already occupied by another piece and it may never jump over another piece. A player loses the game if he or she can't make a move anymore.

Given the initial placement of the pieces, can you tell Lea which player if any can force a win?

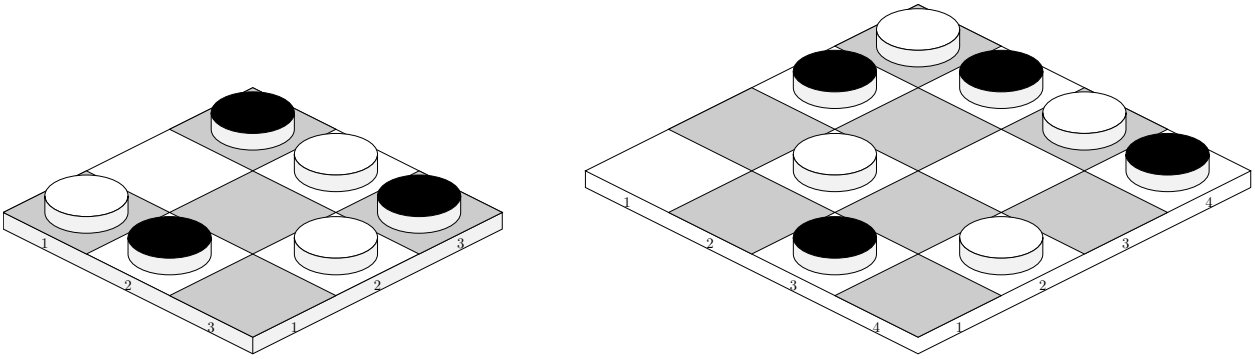

Figure 1: Illustration of the sample inputs.

Input

The first line of the input contains an integer t. t test cases follow, each of them separated by a blank line.

Each test case starts with a single line containing an integer n. The size of the game board is $n \times n$. n lines follow describing the initial placement of the pieces. The i-th line describes the pieces in the i-th row. It consists of two space-separated integers a_i and b_i where a_i is the column of the white piece and b_i is the column of the black piece.

Output

For each test case, print a line containing "Case #i: x" where x is "white" if the white player can force a win, "black" if the black player can force a win, or "neither" if neither of them can force a win.

Constraints

- $1 \leq t \leq 20$
- $1 \leq n \leq 2 \cdot 10^4$
- $1 \leq a_i, b_i \leq n$ for all $1 \leq i \leq n$
- The given board state is always legal.

Sample Input 1	Sample Output 1
2 3 1 3 3 1 2 3 4 4 3 2 4 4 1 2 4	Case #1: black Case #2: white