

Problem A

Backward numbers

Input file: backw.in

Output file: backw.out

Backward numbers are numbers written in ordinary Arabic numerals but the order of the digits is reversed. The first digit becomes the last, and vice versa. For example, the number 1245 becomes 5421. Note that all leading zeroes are omitted. This means that if the number ends with a zero, the zero is lost by the reversal; e.g., 1200 gives 21. Also note that the reversed numbers never have any trailing zeroes.

We need to calculate with backward numbers and your task is to write a program that can add two backward numbers and output their sum as a backward number. Of course, the result is not unique (e.g., 21 could be 12, 120, or 1200 before the reversal). Thus, we assume that no zeroes were lost in the reversal (e.g., we assume that the original number was 12).

Input specifications

The input consists of N cases. The first input line contains only the integer N , and we assume that $N > 0$. The follow the cases. Each case consists of exactly one line with two non-negative integers separated by a space. These are the reversed numbers you are to add.

We assume that all numbers are in the range $0 \leq n < 10^9$.

Output specifications

For each case, print exactly one line containing only one integer—the backward sum of the two backward numbers. Omit any leading zeroes in the output.

Sample input

4
24 1
101 0
4358 754
305 794

Output for sample input

34
101
1998
1