Combination Lock

After hanging out in the sh bowl (also called the ACM room if you're a scrub) for what seemed like an eternity Joon wanted to go grab an ice cold beverage from the refrigerator. Recently, however, a two-digit combination lock was added to the refrigerator door to prevent unauthorized access (Joon); fortunately Joon already knows the two-digit combination that will unlock the refrigerator door!

Joon is good with numbers but is incredibly lazy, therefore he wants to unlock the door with as little effort as possible. To unlock the door he will make a sequence of moves where each move can alter one of the digits on the lock by either incrementing it or decrementing it (increase/decrease by 1). If Joon increments a digit set to 9, then the digit wheel will wrap around to 0. Similarly, if Joon decrements a digit set to 0 it will wrap around to 9.

Given the initial conguration of the combination lock and the unlocked conguration, can you determine the minimum number of moves Joon must make to open the refrigerator door?

Input

The input will begin with a line containing a single positive integer, t, representing the number of test cases to process. Each test case consists of a single line with four integers x_1, y_1, x_2, y_2 ($0 \leq x_1, y_1, x_2, y_2 \leq 9$) where (x_1, y_1) is the initial conguration of the lock and (x_2, y_2) is the unlocked conguration.

Output

For each test case print the minimum number of moves Joon must make to open the refrigerator door, each on its own line.

Example

Input:
2
0 9 1 3
4 5 7 2

Output:
5
6