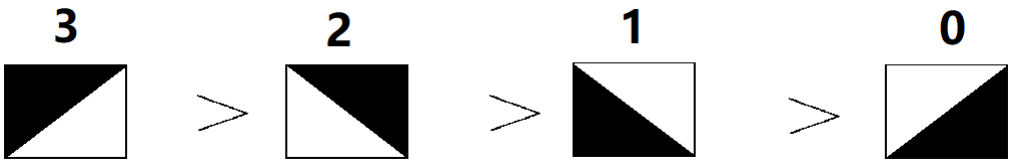
Coloring Rectangles

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Berlinetta has got many white rectangles on a plane in the Cartesian coordinate system. She wants to color half of each white rectangle black along the main diagonal or the second diagonal. After coloring, she wants to have all black triangles not overlapped (but can share edges or points) with each other. Help her find a feasible coloring pattern that is lexicographically **largest** or tell her the sad truth that it is impossible. To rank the coloring patterns lexicographically, we generate a string of digits from the sequence of rectangles, and each half-black rectangle is encoded into a digit 3, 2, 1, or 0 as below



Input

The first line contains an integer n ($1 \le n \le 200$) — the number of rectangles Berlinetta have.

The following n lines give a sequence of n rectangles. Each line contains 4 integers x_1, y_1, x_2, y_2 ($x_1 < x_2$, $y_1 < y_1$, and $|x_1|, |y_1|, |x_2|, |y_2| \le 10^6$) — a rectangle $[x_1, x_2] \times [y_1, y_2]$ in the sequence to be colored.

Output

If there is a solution, output a sequence of integers, each of which shows how we color the i-th rectangle, separated by space. Otherwise, just print "no solution".

Example

standard input	standard output
2 1 1 4 4 3 2 8 8	3 2