

D. Chain Reaction

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Group of Berland scientists, with whom you have a close business relationship, makes a research in the area of peaceful nuclear energy. In particular, they found that a group of four nanobots, placed on a surface of a plate, can run a powerful chain reaction under certain conditions.

To be precise, researchers introduced a rectangular Cartesian coordinate system on a flat plate and selected four distinct points with integer coordinates where bots will be placed initially. Next each bot will be assigned with one of the four directions (up, down, left or right) parallel to the coordinate axes. After that, each bot is shifted by an integer distance (which may be different for different bots) along its direction. The chain reaction starts, if the bots are in the corners of a square with **positive area** with sides parallel to the coordinate axes. **Each corner of the square must contain one nanobot.** This reaction will be stronger, if bots spend less time to move. We can assume that bots move with unit speed. In other words, the lesser is the maximum length traveled by bot, the stronger is reaction.

Scientists have prepared a set of plates and selected starting position for the bots for each plate. Now they ask you to assign the direction for each bot to move after *landing* such that the maximum length traveled by bot is as small as possible.

Input

The first line contains an integer number t ($1 \leq t \leq 50$) — the number of plates.

t descriptions of plates follow. A description of each plate consists of four lines. Each line consists of a pair of integers numbers x_i, y_i ($-10^8 \leq x_i, y_i \leq 10^8$) — coordinates of the next bot. All bots are in different locations.

Note, though, the problem can include several records in one test, you can hack other people's submissions only with the test of one plate, i.e. parameter t in a hack test should be equal to 1.

Output

Print answers for all plates separately. First goes a single integer number in a separate line. If scientists have made an unfortunate mistake and nanobots are not able to form the desired square, print -1 . Otherwise, print the minimum possible length of the longest bot's path.

If a solution exists, in the next four lines print two integer numbers — positions of each bot after moving. Print bots' positions in the order they are specified in the input data.

If there are multiple solution, you can print any of them.

Examples

input
2 1 1 1 -1 -1 1 -1 -1 1 1 2 2 4 4 6 6
output
0 1 1 1 -1 -1 1

-1 -1
-1