



## Pingpong (pingpong)


You are attending a table tennis tournament and you decided to note down the results the players achieved. However, a bad misfortune happened and you lost your detailed list of results so you don't know the way the games went anymore.

There is good news though: In order to avoid a complete loss of data, you also noted how many points each player has scored in total during all the sets they played. In addition, you also know that the first player always won in the end.

Now your goal is to reconstruct the way  $T$  games turned out, if you know the following details:

- The games have been played using normal table tennis rules, but with a key difference. The set always ends when a player reaches 11 points (even when the score is 11-10, so no tiebreaks exist).
- The match is played using a best of 5 system (the first player to win 3 sets wins the game).
- The first player always won in the end.

Since the games can turn out in many different ways, any such sequence of set scores is acceptable.

 Among the attachments of this task you may find a template file `pingpong.*` with a sample incomplete implementation.

## Input

The first line of the input will contain  $T$ , the number of test cases.

Each test case will contain two integers,  $A$  and  $B$ , representing the number of points each player obtained during the game.

## Output

For each test case, you will either print a way the game could have turned out, with each set written on one line, or `-1 -1` if no such way exists.

## Constraints

- $1 \leq T \leq 3000$ .
- $1 \leq A, B \leq 60$  for each test case.

## Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

– **Subtask 1** (0 points)      Examples.



– **Subtask 2** (20 points)      The match can be won by the first player in 3 sets.



- Subtask 3 (30 points)

The match can be won by the first player in 4 sets.
- Subtask 4 (50 points)

No additional limitations.

Examples

input	output
5	11 10
33 15	11 5
40 29	11 0
55 55	11 10
39 16	11 8
29 54	7 11
	11 0
	-1 -1
	11 5
	11 0
	6 11
	11 0
	-1 -1

Explanation

In the **first sample case**, the first player can win in 3 sets, with results 11 – 10, 11 – 5 and 11 – 0. Note that in this case, there are multiple ways in which the first player could have won the game.

In the **last sample case**, the total points won by the first player is 29. It can be seen that there is no way in which they could have won 3 sets (and, therefore, the game) with 29 points total.