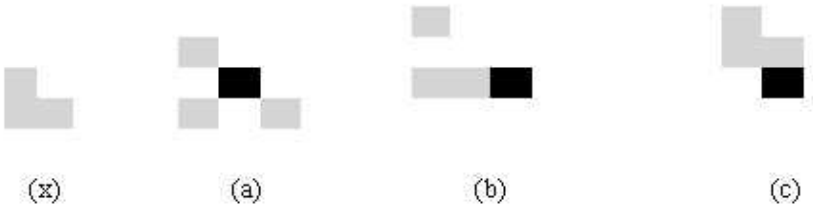
Problem D
Denki Blocks

Input: standard input
Output: standard output
Time Limit: 8 seconds
Memory Limit: 64 MB

'Denki Blocks' is a extremly cool puzzle game. If you have a Game Boy, you may have played this game once. But in this problem, the game is slightly modified, so please read the new rules carefully.

On a infinitely large board, there is a black block at (0,0). There are **n** grey blocks surrounding it, the **i**−th block is at ($\mathbf{x_i}$, $\mathbf{y_i}$). All the $\mathbf{x_i}$ and $\mathbf{y_i}$ are **odd** numbers. The aim of the game is to move the grey blocks together to form a specified piece. The piece can be found anywhere on the board, but cannot be rotated or flipped. Figure(a) shows a legal initial state, there are **3** grey blocks at (**−1**,**−1**),(**1**,**−1**),(**1**,**1**). Figure(x) shows a possible expected piece.



The player can use a sequence of operations to achieve the goal. There are only **4** possible operations.

- 'U': all the grey blocks are moved up(**x** position is decreased by **1**).
- 'D': all the grey blocks are moved down(**x** position is increased by **1**).
- 'L': all the grey blocks are moved left(**y** position is decreased by **1**).
- 'R': all the grey blocks are moved right(**y** position is increased by **1**).

Note that the black block NEVER moves. If the black block lies just in front of one grey block, the grey block cannot be moved. If several grey blocks are connected(two are connected if they share a common edge), they are considered to be **sticked** to each other and can only move together. Therefore, if one of the connected blocks cannot move, the whole component cannot move. For example, In figure(a), if you use operations 'LLUR', you will reach figure(b). If you use 'URD' in figure(b), you will reach figure(c). Notice that in figure(c), the expected piece(see figure(x)) is found, so the game ends successfully.

Write a program to use no more than **10,000** operations to achieve the goal.

Input

The first line of the input is a single integer **t**(**1**<=**t**<=**10**), indicating the number of test cases. Each case begins with a line containing a single integer **n**(**3**<=**n**<=**20**), indicating the number of blocks. The second line contains **n** integer pairs ($\mathbf{x_i}$, $\mathbf{y_i}$), indicating the initial positions of the blocks. The blocks are sorted in ascending order of **x**, then ascending order of **y**. The third line contains **n** integer pairs ($\mathbf{Px_i}$, $\mathbf{Py_i}$), indicating the final(relative) positions of the blocks. The final positions of the blocks are connected. Input data always have solutions. All the position numbers are odd numbers between **−9** and **9**.

Output

For each test case, print in a line the value of **m**, the number of operations you need. In the second line, print **m** characters representing you solution. If multiple solutions are found, print anyone.

Sample Input

```
1
3
-1 -1 1 -1 1 1
1 1 2 1 2 2
```

Sample Output

```
7
LLURURD
```

The 2nd OIBH Online Programming Contest. Author: Rujia Liu