Problem A. Rotating Twins

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	64 megabytes

It is not the case that twins are always exactly the same, there exists tiny differences in most cases and in some rare case either they are perfect or even rivals. Now we will go on a journey to figure out the twin status between two squares of alphabets.

A square of alphabets contains characters A-Z in each of its cell. Now if the two given squares are exactly similar in their original state then they are considered perfect. But if not, then we will try to rotate either of them and check if its a match or not, we will keep on rotating to find a match. If we find a match then the minimum number of rotation required to reach there is the status of their twin hood.

But if there is no such match between them, then it is considered that they are not related.

Input

The first input is an integer n ($1 \leq n \leq 100$), the size of the first square. Following this is a $n * n$ grid of uppercase alphabets.

The next input is an integer m ($1 \leq m \leq 100$), the size of the second square. Following this is a $m * m$ grid of uppercase alphabets.

Output

If no rotation is needed print “Perfect Twins”, if 1 rotation needed print “Normal Twins”, if 2 or more rotation needed print “Rival Twins” and lastly if they are not related then print “Not Related”, without the quotes.

Examples

standard input	standard output
2 AB CD 2 AB CD	Perfect Twins
2 AA AB 2 AB AA	Normal Twins
2 AA BB 2 BB AA	Rival Twins
2 AB CD 2 WX YZ	Not Related