110502 Reverse and Add

The *reverse and add* function starts with a number, reverses its digits, and adds the reverse to the original. If the sum is not a palindrome (meaning it does not give the same number read from left to right and right to left), we repeat this procedure until it does.

For example, if we start with 195 as the initial number, we get 9,339 as the resulting palindrome after the fourth addition:

195	786	1,473	5,214
591	687	3,741	4,125
+ ———	+ ———	+ ———	+ ———
786	1,473	5,214	9,339

This method leads to palindromes in a few steps for almost all of the integers. But there are interesting exceptions. 196 is the first number for which no palindrome has been found. It has never been proven, however, that no such palindrome exists.

You must write a program that takes a given number and gives the resulting palindrome (if one exists) and the number of iterations/additions it took to find it.

You may assume that all the numbers used as test data will terminate in an answer with less than 1,000 iterations (additions), and yield a palindrome that is not greater than 4,294,967,295.

Input

The first line will contain an integer N ($0 < N \leq 100$), giving the number of test cases, while the next N lines each contain a single integer P whose palindrome you are to compute.

Output

For each of the N integers, print a line giving the minimum number of iterations to find the palindrome, a single space, and then the resulting palindrome itself.

Sample Input	Sample Output
3	4 9339
195	5 45254
265	3 6666
750	

