12. Reverse Pals

Program Name: ReversePals.java   Input File: reversepals.dat

There are many numbers that are palindromes. A number that is a palindrome is a number that, when the digits are reversed, the number remains the same. The number 121 is an example of a number that is a palindrome.

Most integers that are not palindromes have an integer that we will call a Reverse Pal that is a palindrome. The Reverse Pal for a number is found by a process that begins by reversing the digits in that number and adding that reverse to the number. If the resulting sum is not a palindrome, take the sum and repeat the above process until the result is a palindrome, the Reverse Pal of the original number.

For example, consider the number 97:

- `97 + 79 = 176`
- `176 + 671 = 847`
- `847 + 748 = 1595`
- `1595 + 5951 = 7546`
- `7546 + 6457 = 14003`
- `14003 + 30041 = 44044`   `//`palindrome found after the 6th iteration

You are to write a program to find the Reverse Pal of a given integer and the number of iterations it took to find its Reverse Pal. Although not all integers have a Reverse Pal, you may assume that each integer in our data set does have a Reverse Pal and that it can be found in 20 iterations or less. The number of iterations is the number of times in the process that you reversed the digits of a number and added the reverse to that number.

Input

The first line of input will contain a single integer `n` that indicates the number of positive integers to follow. Each the following `n` lines will contain a single positive integer less than 1000 that is not a palindrome.

Output

For each integer input, you will print the Reverse Pal for that integer followed by a space and the number of iterations it took to find the Reverse Pal.

Example Input File

```
4
97
125
987
882
```

Example Output to Screen

```
44044 6
646 1
66066 4
1881 2
```