Problem

Palindromic Numbers

A palindrome is an expression which reads the same backwards as forwards. For example: “radar” or 23432.

Pick a positive integer. Reverse its digits and add the resulting number to the original number. If the sum isn’t a palindrome, repeat the process with the sum. Do all positive integers in base 10 eventually become palindromes through this process? Let’s find out.

For example, starting with 48 and applying the process

```
  48
 +84  (reverse)
 ---
 132  (sum)
+231  (reverse)
----
 363  (sum)   (palindrome)
```

The program that you are required to write will determine the palindrome of a given positive integer and how many steps it took to achieve the palindrome. For the sake of this program, if the palindrome is not found within 100 steps, then we will assume that it will not be possible to find the palindrome.

The input file (**pal.txt**) will contain at least 3 lines of data. Each line will contain one positive integer less than or equal to 10000.

The output will be in the following format:

If the palindrome can be found, the line will contain the original number, how many steps it took to find the palindrome, and the palindrome.

If the palindrome can not be found, the line will contain the original number, “NOT POSSIBLE” and the sum formed after the 100^{th} step in the process.

Each item on the output line is separated by a single space.

Sample Input (pal.txt)

```
48
59
196
```

Sample Output

```
48  2  363
59  3  1111
196  NOT  POSSIBLE  44757771534490515617290699271561508443627774644
```