C - Sum of Round Numbers

A positive (strictly greater than zero) integer is called *round* if it is of the form d00...0. In other words, a positive integer is round if all its digits except the leftmost (most significant) are equal to zero. In particular, all numbers from 1 to 9 (inclusive) are round.

For example, the following numbers are round: 4000, 1, 9, 800, 90. The following numbers are **not** round: 110, 707, 222, 1001.

You are given a positive integer n (1≤*n*≤104). Represent the number *n* as a sum of round numbers using the minimum number of summands (addends). In other words, you need to represent the given number *n* as a sum of the least number of terms, each of which is a round number.

**Input**

The first line contains an integer *t* (1≤*t*≤104) — the number of test cases in the input. Then *t* test cases follow.

Each test case is a line containing an integer *n* (1≤*n*≤104).

**Output**

Print t answers to the test cases. Each answer must begin with an integer *k* — the minimum number of summands. Next, *k* terms must follow, each of which is a round number, and their sum is *n*. The terms can be printed in any order. If there are several answers, print any of them.

**Example**

**Input**

5

5009

7

9876

10000

10

**Output**

2

5000 9

1

7

4

800 70 6 9000

1

10000

1

10