



Problem G

(Program filename: G.CPP, G.DPR, or G.PAS)

Skew Binary

It had been a year since Swamp County Computing established a functional programming group. Your (non-functional programming) group is going to throw a surprise party for the anniversary. Now the functional folks really like skew binary numbers for some reason. "Easy to increment and decrement!" they say. Your task is to write a program to convert decimal integers to skew binary in the format they like. This will help in making banners and other party material.

Number representations are made up of a list of digits. Call the lowest order digit the rank 0 digit, the next, rank 1, etc. For example, in decimal representation, digits are 0-9, the rank 0 digit has weight 1, the rank 1 digit has weight 10, and the rank i digit has weight 10^i . In binary representation, the digits are 0 and 1, and the rank i digit has weight 2^i . In skew binary representation, the digits are 0, 1, and 2, and the rank i digit has weight $2^{i+1} - 1$.

Rank	Weight
0	1
1	3
2	7
3	15
4	31
5	63
6	127
7	255
⋮	⋮

Allowing the digit 2 in the skew binary means there may be several ways to represent a given number. However the convention is that the digit 2 may only appear as the lowest rank non-zero digit. This makes the representation unique.

In this problem, you should use a special way to write skew binary numbers as a list of ranks of non-zero digits in the number. The digit 2 is represented by the rank of the digit appearing twice in the list. Note that this means that *only* the first two ranks in the list may be equal.

Each rank is a decimal integer, and is separated from the next rank by a comma (','). A list is started by a '[' and ended by a ']'. For example, the decimal number 5, which has the skew representation 12, should be written as [0, 0, 1]. Decimal 0 is an empty list: [].

Input consists of decimal numbers, one per line,

Input (filename: G.IN)

The first line of the input file contains a single integer t ($1 \leq t \leq 10$), the number of test cases, followed by t lines, each containing a single decimal number with no leading or trailing white space. Each number will be in the range 0...100663270 (inclusive).

Output (filename: G.OUT)

There should be one line per test case containing the input decimal number, with no leading zeros or spaces, a single space, and the skew binary equivalent in list format with no leading or trailing spaces. Within the list each rank should have no extra leading zeros or leading or trailing spaces.

Sample Input

```
5  
0  
1  
2  
3  
4
```

Sample Output

```
0 []  
1 [0]  
2 [0,0]  
3 [1]  
4 [0,1]
```