

Phil Oracle has a unique ability that makes him indispensable at the National Spying Agency. His colleagues can bring him any new binary code and he can tell them immediately whether the code is uniquely decodable or not. A code is the assignment of a unique sequence of characters (a codeword) to each character in an alphabet. A binary code is one in which the codewords contain only zeroes and ones. For example, here are two possible binary codes for the alphabet {a,c,j,l,p,s,v}.

	Code 1	Code 2
a	1	010
c	01	01
j	001	001
l	0001	10
p	00001	0
s	000001	1
v	0000001	101

The encoding of a string of characters from an alphabet (the *cleartext*) is the concatenation of the codewords corresponding to the characters of the cleartext, in order, from left to right. A code is uniquely decodable if the encoding of every possible cleartext using that code is unique. In the example above, Code 1 is uniquely decodable, but Code 2 is not. For example, the encodings of the cleartexts “pascal” and “java” are both 001010101010.

Even shorter encodings that are not uniquely decodable are 01 and 10.

While the agency is very proud of Phil, he unfortunately gives only “yes” or “no” answers. Some members of the agency would prefer more tangible proof, especially in the case of codes that are not uniquely decodable. For this problem you will deal only with codes that are *not* uniquely decodable. For each of these codes you must determine the single encoding having the minimum length (measured in bits) that is ambiguous because it can result from encoding each of two or more different cleartexts. In the case of a tie, choose the encoding which comes first lexicographically.

Input

One or more codes are to be tested. The input for each code begins with an integer m , $1 \leq m \leq 20$, on a line by itself, where m is the number of binary codewords in the code. This is followed by m lines each containing one binary codeword string, with optional leading and trailing whitespace. No codeword will contain more than 20 bits.

The input is terminated by the number zero on a line by itself.

Output

For each code, display the sequential code number (starting with 1), the length of the shortest encoding that is not uniquely decodable, and the shortest encoding itself, with ties broken as previously described. The encoding must be displayed with 20 bits on each line except the last, which may contain fewer than 20 bits. Place a blank line after the output for each code. Use the format shown in the samples below.

Sample Input

```
3
 0
 01
 10
5
0110
 00
  111
001100
  110
5
 1
 001
 0001
00000000000000000001
10000000000000000000
0
```

Sample Output

```
Code 1: 3 bits
010

Code 2: 9 bits
001100110

Code 3: 21 bits
100000000000000000000
1
```