

## Problem B. Bitwise Automaton

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

A *bitwise automaton* is a directed acyclic graph with its vertices called *states*, which has two *terminal states*, called 0 and 1, and an arbitrary amount of other states numbered with consecutive integers starting from 2. The terminal states have no outgoing edges. Each non-terminal state has exactly two outgoing edges, one marked with 0 and one marked with 1. Additionally, each non-terminal state  $i$  has a *bit number*  $b_i$  associated with it, and one of the states (either terminal or non-terminal) is denoted as the *starting state*.

A bitwise automaton can be evaluated on an integer input  $x$  in the following way: first, we put a token to the starting state. While the token is in a non-terminal state  $i$ , we do the following: check the  $b_i$ -th bit of  $x$ , and move the token to a new state using the outgoing edge with the corresponding label. The bits are numbered from 0 from lowest to highest: the 0-th bit is the parity bit of  $x$ , and so on. Eventually the token will reach a terminal state, and the number of this state (either 0 or 1) is the output of the evaluation for the given  $x$ .

You are given the desired result of evaluation of an automaton for all inputs from 0 to  $n - 1$ , and need to construct an automaton with the smallest number of states that gives such evaluation results.

### Input

The input contains multiple test cases. The first line of the input contains the number  $t$  of test cases,  $1 \leq t \leq 510$ . Each test case is then described with two lines. The first of those lines contains one integer  $n$ ,  $1 \leq n \leq 8$ . The second of those lines contains  $n$  integers, each either 0 or 1 — the desired results of evaluation for inputs 0, 1, ...,  $n - 1$ .

### Output

Print the outputs for all test cases in order. For each test case, first print the number  $m$  ( $m \geq 2$ ) of states in your automaton on a line of its own. It must be the minimum possible number of states for an automaton that produces the desired results.

In the next  $m - 2$  lines describe the states from 2 to  $m - 1$  in order. Describe a state with three integers: the bit number  $b_i$  ( $0 \leq b_i \leq 2$ ), the number of the state where the 0 edge leads, and the number of the state where the 1 edge leads.

In the next line print the number of the starting state.

In case there are multiple possible solutions, output any.

### Example

standard input	standard output
1	4
5	0 3 0
1 0 0 0 1	1 1 0
	2