



## **B** Walking the Cube

TIME LIMIT: 3.0s  
MEMORY LIMIT: 256MB

*This is an interactive problem.*

Alice and Bob are playing a game on a  $k$ -dimensional Boolean cube. Its vertices are all binary strings of length  $k$ . There is an edge between two vertices if their strings differ at exactly one position.

Initially, the token is at the vertex  $00\dots 0$ . Alice and Bob make moves alternately. In one move, a player must choose one bit and flip it. In other words, from the current vertex, the token can be moved to any vertex connected by an edge.

It is forbidden to move the token to a vertex that has already been visited (the initial vertex  $00\dots 0$  counts as visited). The player who cannot make a move loses.

You may choose whether you want to play as Alice or as Bob. After that, you have to play the game and win.

### INPUT

Each test contains multiple test cases. The first line contains the number of test cases  $t$  ( $1 \leq t \leq 100$ ). The description of the test cases follows.

The only line of each test case contains a single integer  $k$  ( $3 \leq k \leq 10$ ) — the dimension of the Boolean cube.

Then interaction follows.

### INTERACTION

First, you need to output one of the following lines:

- **Alice** — if you want to play first;
- **Bob** — if you want to play second.

After that, the game starts at the vertex  $00\dots 0$ .

If you chose Alice, you should make the first move. If you chose Bob, the jury will make the first move.

To make a move, output a single integer  $i$  ( $1 \leq i \leq k$ ) — the position of the bit you want to flip.

After each move of the jury, you should read a single integer  $i$  ( $1 \leq i \leq k$ ) — the position of the bit flipped by the jury.

If you cannot make a move, you should print 0. The interactor will print  $-1$  and finish the interaction.



If the jury cannot make a move and loses, you should read 0 and proceed to the next test case.

After printing each move, do not forget to output the end of line and flush the output. Otherwise, you will get *Idleness limit exceeded* verdict.

If, at any interaction step, you read  $-1$ , your solution must terminate immediately. This means that your solution has made an invalid move, you lost, or some other error happened. Failing to terminate can result in an arbitrary verdict.

The jury program is adaptive, meaning it's behavior might be different on the same test based on the participant's choices.

## SAMPLES

Sample input 1	Sample output 1
1	
3	Bob
3	2
1	3
2	3
0	