Problem B. Vanishing Inversions

Alice and Bob are playing a game on a permutation, with Alice moving first.

A sequence of length n is called a **permutation** if and only if each integer from $1, 2, \ldots, n$ appears exactly once in the sequence.

Initially, a permutation $a_1, a_2, \ldots, a_n$ of length n is given. At any moment during the game, the current sequence is always a permutation of some length m. The players take turns, and on each turn the current player must choose one of the following two operations:

- Choose an index i $(1 \le i < m)$ such that $a_i > a_{i+1}$, and swap a_i and a_{i+1};
- Choose an index i $(1 \le i < m)$ such that $a_{i+1} = a_i + 1$, delete a_i and a_{i+1}, and then renumber the remaining elements according to their relative order: the k-th smallest remaining element is renumbered as k, so that the remaining sequence becomes a permutation again.

For example, suppose that the current permutation is $[2, 3, 4, 1]$. The adjacent elements 2 and 3 can be deleted, leaving $[4, 1]$; after renumbering, the permutation becomes $[2, 1]$.

A player who cannot perform any operation loses the game. Assuming that both Alice and Bob play optimally, determine the winner.

Input

Each test contains multiple test cases.

The first line contains an integer T $(1 \le T \le 5000)$, the number of test cases.

For each test case:

- The first line contains an integer n $(1 \le n \le 5000)$.
- The second line contains n integers $a_1, a_2, \ldots, a_n$. It is guaranteed that a is a permutation of length n.

It is guaranteed that the sum of n over all test cases does not exceed 5000.

Output

For each test case, output `Alice` if Alice wins; otherwise, output `Bob`.

Example

standard input	standard output
5 1 1 2 1 2 2 2 1 4 2 4 1 3 5 5 1 4 2 3	Bob Alice Bob Alice Bob