A. LRC and VIP

time limit per test: 1 second
memory limit per test: 256 megabytes

You have an array a of size n — $a_1, a_2, \ldots a_n$.

You need to divide the n elements into 2 sequences B and C, satisfying the following conditions:

- Each element belongs to exactly one sequence.
- Both sequences B and C contain at least one element.
- $\gcd(B_1, B_2, \ldots, B_{|B|}) \neq \gcd(C_1, C_2, \ldots, C_{|C|})$ *

*$\gcd(x, y)$ denotes the greatest common divisor (GCD) of integers x and y.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \leq t \leq 500$). The description of the test cases follows.

The first line of each test case contains an integer n ($2 \leq n \leq 100$).

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^4$).

Output

For each test case, first output `Yes` if a solution exists or `No` if no solution exists. You may print each character in either case, for example `YES` and `yEs` will also be accepted.

Only when there is a solution, output n integers on the second line. The i-th number should be either 1 or 2. 1 represents that the element belongs to sequence B and 2 represents that the element belongs to sequence C.

You should guarantee that 1 and 2 both appear at least once.

Example

input

```
3
4
1 20 51 9
4
5 5 5 5
3
1 2 2
```

output

```
Yes
2 2 1 1
No
Yes
1 2 2
```

Note

In the first test case, $B = [51, 9]$ and $C = [1, 20]$. You can verify $\gcd(B_1, B_2) = 3 \neq 1 = \gcd(C_1, C_2)$.

In the second test case, it is impossible to find a solution. For example, suppose you distributed the first 3 elements to array B and then the last element to array C. You have $B = [5, 5, 5]$ and $C = [5]$, but $\gcd(B_1, B_2, B_3) = 5 = \gcd(C_1)$. Hence it is invalid.