Problem I. Good Subsegments

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

You are given an array $a[1..n]$ consisting of n integers from 1 to n. A *subsegment* $a[\ell..r]$ of the array is its consecutive part from position ℓ to position r, inclusive.

A subsegment $a[\ell..r]$ is *k-good* if the following conditions are satisfied:

- $r - \ell + 1 \geq 2 \cdot k$, so its length is at least $2 \cdot k$;
- $a_\ell = a_{\ell+1} = a_{\ell+2} = ... = a_{\ell+k-1}$, so at least k of its leftmost elements are equal to each other;
- $a_r = a_{r-1} = a_{r-2} = ... = a_{r-k+1}$, so at least k its rightmost elements are equal to each other;
- $a_\ell = a_r$, so its ends are equal.

For each k from 1 to $\lfloor \frac{n}{2} \rfloor$, find the number of k-good subsegments of the given array a.

Input

The first line contains an integer t ($1 \leq t \leq 5 \cdot 10^5$), the number of test cases. The test cases follow.

The first line of each test case contains an integer n ($2 \leq n \leq 5 \cdot 10^5$).

The second line consists of n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq n$).

The sum of n over all test cases does not exceed $5 \cdot 10^5$.

Output

For each test case, print a line with $\lfloor \frac{n}{2} \rfloor$ integers: the number of k-good subsegments for each corresponding k, starting from 1.

Example

standard input	*standard output*
4 10 1 2 2 2 2 2 3 2 2 2 6 1 1 1 2 1 1 9 2 2 1 1 1 2 2 1 1 10 3 2 3 2 4 2 10 10 10 10	28 11 3 0 0 10 2 0 16 3 0 0 10 1 0 0 0