Problem A. Permutations and Cycles (Minimum Version)

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

For a given x, a permutation of size n is called *good* if for each $1 \le i < n$ the condition $p_i + p_{i+1} \le x$ holds. Find any good permutation with the **minimum** number of cycles.

A permutation of size n is a sequence of n distinct integers from 1 to n.

A cycle of a permutation p is a sequence of indices $i_1, i_2, \ldots, i_k$ such that $p_{i_1} = i_2$, $p_{i_2} = i_3$, ..., $p_{i_k} = i_1$. The cycles obtained by a cyclic shifting of the sequence are considered to be the same.

Input

The first line contains an integer t ($1 \le t \le 2 \cdot 10^5$), the number of test cases. The test cases follow.

Each test case is given on a line with two integers n ($2 \le n \le 2 \cdot 10^5$) and x ($n+1 \le x \le 2 \cdot n - 1$). These constraints guarantee that at least one *good* permutation exists.

The sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, print two lines. The first one should contain the minimum number of cycles in a good permutation of length n. The second line should consist of n integers: the permutation itself. If multiple such permutations exist, print any one of them.

Example

standard input	*standard output*
2 2 3 6 10	1 2 1 1 5 4 6 3 2 1