Problem B. Binary Arrays and Sliding Sums

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You are given two integers n, k $(2 \le k < n)$.

For an array $a_1, a_2, \ldots, a_n$, which **consists only of zeros and ones**, we define the array $f(a)$ of length n as follows:

- $f(a)_i = a_i + a_{i+1} + \ldots + a_{i+k-2} + a_{i+k-1}$ (here we assume that $a_{n+i} = a_i$, i.e. numbers are arranged in a circle).

  For example, for $n = 4, k = 2,$ $f([0, 1, 1, 0]) = [1, 2, 1, 0]$.

Consider all 2^n possible arrays a, and for each of them, find $f(a)$. How many different arrays are there among them? Since the answer may be very large, print the number of these arrays modulo 998244353.

Two arrays are considered different if they differ in at least one position.

Input

The first line contains a single integer t $(1 \le t \le 10^5)$ — the number of test cases. The description of test cases follows.

The first line of each test case contains two integers n, k $(2 \le k < n \le 10^6)$.

Output

For each test case print the number of different arrays among $f(a)$, modulo 998244353.

Example

standard input	standard output
4 3 2 4 2 42 3 123123 123	8 15 780086989 126500246

Note

For $n = 3, k = 2$, there are 8 different arrays a of ones and zeros. The corresponding arrays $f(a)$ are pairwise distinct for them.

For $n = 4, k = 2$, there are 16 distinct arrays a of ones and zeros. The only pair of matching arrays $f(a)$ is $[0, 1, 0, 1]$ and $[1, 0, 1, 0]$: for both arrays $f(a)$ is $[1, 1, 1, 1]$.