A. Special Permutation

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

You are given one integer n ($n > 1$).

Recall that a permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[2, 3, 1, 5, 4]$ is a permutation of length 5, but $[1, 2, 2]$ is not a permutation (2 appears twice in the array) and $[1, 3, 4]$ is also not a permutation ($n = 3$ but there is 4 in the array).

Your task is to find a permutation p of length n that there is no index i ($1 \le i \le n$) such that $p_i = i$ (so, for all i from 1 to n the condition $p_i \ne i$ should be satisfied).

You have to answer t independent test cases.

If there are several answers, you can print any. It can be proven that the answer exists for each $n > 1$.

Input

The first line of the input contains one integer t ($1 \le t \le 100$) — the number of test cases. Then t test cases follow.

The only line of the test case contains one integer n ($2 \le n \le 100$) — the length of the permutation you have to find.

Output

For each test case, print n distinct integers $p_1, p_2, \ldots, p_n$ — a permutation that there is no index i ($1 \le i \le n$) such that $p_i = i$ (so, for all i from 1 to n the condition $p_i \ne i$ should be satisfied).

If there are several answers, you can print any. It can be proven that the answer exists for each $n > 1$.

Example

input	Copy
2 2 5	

output	Copy
2 1 2 1 5 3 4	

→ **Virtual participation**

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ **Problem tags**

constructive algorithms   probabilities   *800

No tag edit access

→ **Contest materials**

- Announcement (en)   ✕
- Tutorial   ✕
- Discussion (en)   ✕

Codeforces (c) Copyright 2010-2021 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Mar/16/2021 22:52:57^{UTC-5} (f2).
Desktop version, switch to mobile version.
Privacy Policy

Supported by