En Garde

May 2024
C++ — 2 sec — 512 MB

This month marks the bicentennial fencing match between the Guild of Sorcerers and the Warlock's Cooperative. Swordsmen (and swordswomen) will gather at the ancient Hall of Épée for some corps-à-corps action: attacks, parries, and ripostes.

However, the pen is indeed mightier than the sword, and before the bouts can begin, the logistics must first be sorted. The sorcerers and the warlocks have agreed to field **n** teams, with players on each team numbered from 1 to **n**.

To ensure everyone has sufficient redoublement and remise, each player will fight every single player on the other team. Thus, the match will take place at **n** different time slots (numbered from 1 to **n**) on **n** different days. Both the sorcerers and the warlocks have their foibles, so the teams have agreed that each player must fight exactly once on each day, and exactly once at each different time slot.

En Garde... Allez!

INPUT You will be given an integer, **n**, denoting the number of players on each team.

$3 \leq \mathbf{n} \leq 3{,}000$

OUTPUT You should output the time slot in which each player will fight on each day. You should output the times for the sorcerers first, and then for the warlocks.

For the sorcerers, the **i**th line should contain **n** integers, the **j**th integer giving the time slot when the **i**th player will fight on the **j**th day. The same format should be used for the warlocks. If no valid ordering of fights exists, you should instead output `IMPOSSIBLE`.

SAMPLE For example, suppose each team has 4 swordsmen. Let us label the sorcerers from A to D and the warlocks from α to δ. A valid bout timetable is given below, where the rows represent days and the columns represent time slots.

Aα	Bγ	Cδ	Dβ
Bβ	Aδ	Dγ	Cα
Cγ	Dα	Aβ	Bδ
Dδ	Cβ	Bα	Aγ

INPUT	OUTPUT
4	1 2 3 4 2 1 4 3 3 4 1 2 4 3 2 1 1 4 2 3 4 1 3 2 2 3 1 4 3 2 4 1
6	IMPOSSIBLE
7	1 2 3 4 5 6 7 2 3 4 5 6 7 1 3 4 5 6 7 1 2 4 5 6 7 1 2 3 5 6 7 1 2 3 4 6 7 1 2 3 4 5 7 1 2 3 4 5 6 1 2 3 4 5 6 7 3 4 5 6 7 1 2 5 6 7 1 2 3 4 7 1 2 3 4 5 6 2 3 4 5 6 7 1 4 5 6 7 1 2 3 6 7 1 2 3 4 5