

## C. Monster-Go

Nome do problema	Monster-Go
Tempo limite	1 segundo
Memória limite	1 gigabyte

Helen e seus amigos descobriram um novo jogo incrível para seus celulares. O jogo, chamado *Monster-Go*, consiste em capturar monstros caminhando até diferentes ninhos de monstros ao ar livre. Há um número infinito de monstros de um único tipo disponíveis em cada ninho. Quando os amigos chegam a um ninho de monstros, cada um deles captura e adiciona seu monstro à coleção. Há um total de 50 monstros diferentes que os amigos podem capturar, numerados de  $0, 1, \dots, 49$ .

Para tornar o jogo mais emocionante, os  $N$  amigos decidiram que cada jogador terá uma lista personalizada de exatamente 12 tipos monstros para colecionar. A primeira pessoa a capturar todos os monstros da sua lista vence o jogo. Eles querem criar listas de tal forma que, independente da ordem em que visitem os ninhos dos monstros, sempre haja um único vencedor — nunca um empate.

Você pode ajudá-los a criar as listas? Sua pontuação dependerá do número de valores de  $N$ , o número de pessoas jogando, para os quais você for capaz de resolver o problema.

### Entrada

A primeira e única linha de entrada contém o inteiro  $N$ , o número de jogadores.

### Saída

Produza  $N$  linhas, onde a  $i$  ésima linha contém 12 inteiros distintos  $c_{i,1}, c_{i,2}, \dots, c_{i,12}$  (onde  $0 \leq c_{i,j} \leq 49$ ) representando os monstros na lista da pessoa  $i$ . Caso haja múltiplas soluções, você pode imprimir qualquer uma delas.

### Restrições e pontuação

- $1 \leq N \leq 50$ .

Sua solução será testada em um conjunto de grupos de teste, cada um valendo um certo número de pontos. **O  $i$  ésimo grupo de testes contém um único caso de teste com  $N = i$  e vale 2 pontos.** Ou seja, há um total de 50 testes (um para cada  $N = 1, 2, \dots, 50$ ), e sua pontuação neste problema é o dobro do número de testes que seu programa resolve.

Grupo	Pontuação	Limites
1	2	$N = 1$
2	2	$N = 2$
3	2	$N = 3$
$\vdots$	$\vdots$	$\vdots$
49	2	$N = 49$
50	2	$N = 50$

## Exemplo

No exemplo, onde há  $N = 2$  amigos, o programa deve gerar duas listas. De fato, para as duas listas na saída do exemplo, os amigos não podem vencer ao mesmo tempo, independentemente da ordem em que visitam os ninhos dos monstros. Observe que há muitas outras respostas válidas.

Entrada	Saída
2	0 1 2 3 4 5 6 7 8 9 10 11 38 39 40 41 42 43 44 45 46 47 48 49