Crazy domino

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Joker, being treated in the Arkhem medical facility, feels bored at times because patients don't get much entertainment — mostly chess or domino. Today he was sitting there and moving the domino pieces randomly on the chess board when a great self-entertainment idea came to his mind. His idea was to take a puzzle consisting of a square chess board $n \times n$, and a set of domino pieces. Luckily, one domino pies has the same size as two chess squares with a shared side. Now Joker intends to put some checkers so that:

- The total number of checkers is less or equal to n
- All remaining vacant chess squares could be covered by domino pieces. Furthermore, each domino piece must cover 2 squares with a shared side, no 2 pieces cover the same square. Each square must be covered.
- Such domino covering arrangement is unique. There can't be 2 different arrangements that satisfy the criteria.

Joker has already solved this problem and found such an arrangement. Now it's your turn to do the same.

Input

The first line contains one integer number n ($2 \leq n \leq 100$).

Output

Output n lines containing n symbols each — the target checkers' arrangement. Vacant squares are represented by "`.`", squares with checkers are represented by "`#`".

If there exist several arrangements output any. It is guaranteed that there exists at least one suitable arrangement — because Joker found one!

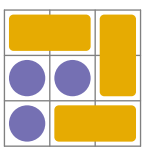
Examples

standard input	standard output

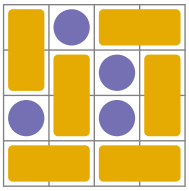
3	`...`
	`##.`
	`#..`

4	`.#..`
	`..#.`
	`#.#.`
	`....`

Note



(a) First test case



(b) Second test case

Рис. 1: Pictures to answers to test cases with a unique arrangements of vacant squares.