# Tiling Madness (`tiling`)

You want to cover a $N \times N$ grid with $N$ non-overlapping identical $2N$-minoes.

The $2N$-minoes are not required to be entirely inside the $N \times N$ grid.

More formally, every solution to this problem must fix a $2N$-mino, and then place $N$ copies of it on a grid (without rotating or reflecting it) so that:

- every cell of the grid is part of at most one of the $2N$-minoes.
- there exists an $N \times N$ subgrid entirely covered by the $2N$-minoes.

A $2N$-mino is a connected set of $2N$ squares; you can find an example of a valid and of an invalid $2N$-mino in Figure 1.

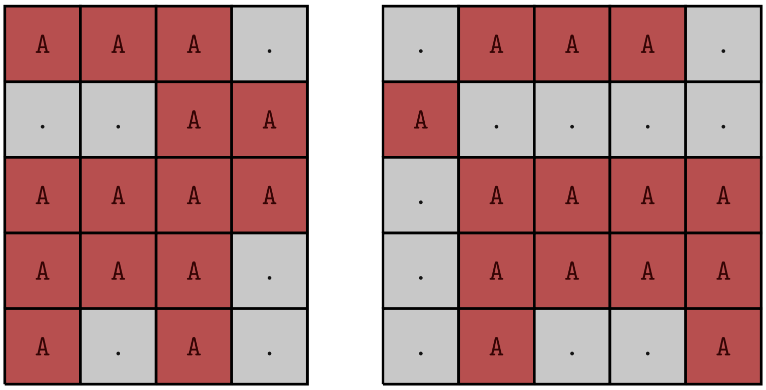



Figure 1: The figure on the left is a valid 14-mino. The one on the right is not since it is not connected.

We want to know many ways to tile the grid, each of which uses a **unique** $2N$-mino; your score will depend on how many valid $2N$-minoes that tile the $N \times N$ square you provide.

Note that $2N$-minoes that can be obtained from each other by rotation or reflection are considered **distinct**.

## Implementation

This is an output-only task. You will have to submit exactly one output file.

### Input format

The only input file consists of a single line, containing the integer $N$.

### Output format

The only output file should be in the following format:

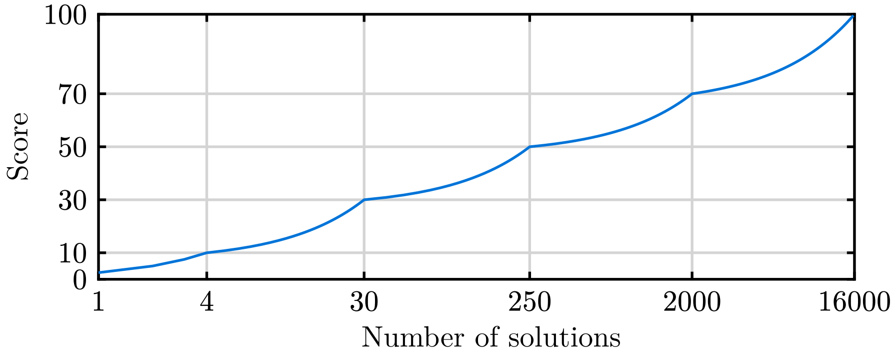
- The first line should contain a single integer $C$ ($0 \le C \le 16000$): the number of different solutions contained in your output.
- Then $C$ solution blocks should follow. Each block should be in the following format:
  - The first line should contain two integers $h$ and $w$ ($0 \le h, w \le 5N$): the height and the width of the grid where you are going to place the $2N$-minoes in.
  - The next $h$ lines should each contain a string of length $w$, made up of the first $N$ uppercase letters of the latin alphabet and the dot (`.`) character. The $i$-th letter of the alphabet indicates that the cell is occupied by the $i$-th copy of the $2N$-mino, while the dot indicates that the cell is left empty.

For each solution block, the grid must contain a $N \times N$ sub-grid that doesn't contain any `.` character. All the $N$ copies of the $2N$-mino must be identical.

## Scoring

This task has exactly 1 test case, where $N = 7$. The score $S$ for your solution is determined according to the following table. Between the values specified in the table, the score will be assigned by **linear interpolation**. A malformed output always scores zero points.

| Solutions | Score |
|---|---|
| 0 | 0 |
| 4 | 10 |
| 30 | 30 |
| 250 | 50 |
| 2000 | 70 |
| 16000 | 100 |

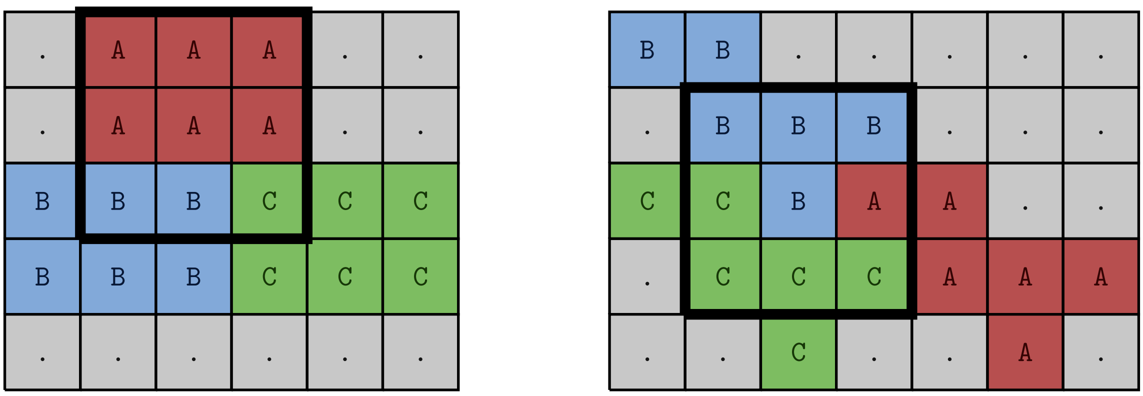



## Examples

| input | output |
|---|---|
| `3` | `2`<br>`5 6`<br>`.AAA..`<br>`.AAA..`<br>`BBBCCC`<br>`BBBCCC`<br>`......`<br>`5 7`<br>`BB.....`<br>`.BBB...`<br>`CCBAA..`<br>`.CCCAAA`<br>`..C..A.` |

## Explanation

In **the sample case** we are asked to use 6-minoes to cover a $3 \times 3$ square: note that this is not a valid input, since in the only input $N = 7$.
The output shows two of the many possible solutions, shown in the image below.




In both cases, we can see that there are 3 identical non overlapping 6-minoes and that a $3 \times 3$ square is covered.