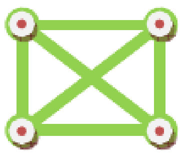
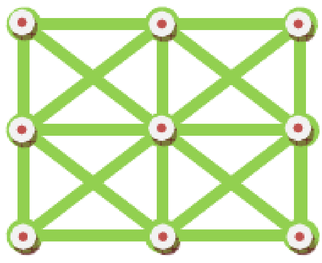
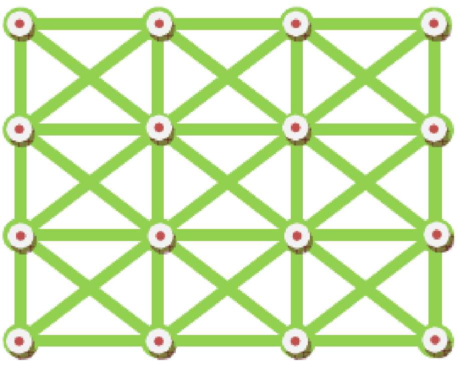
Screen lock on a smart phone can be secured with pattern. Unlock pattern may have 9 nodes on the 3×3 grid. In this problem, we modify this arrangement slightly such that the grid size can be varied from 2×2 to $m \times m$, and only a limited number of connections between adjacent nodes are permitted. Let call this a *modified* mesh graph G and V is a set of vertices (nodes) and E is a set of edges (connections). Examples of 2×2, 3×3 and 4×4 modified mesh graphs are given in the figures below. Note that the corner vertices have 3 connections at most, while the middle vertices can have up to 8 connections. Only a connection to adjacent vertices is allowed, a constraint of the problem.



$m = 2$



$m = 3$



$m = 4$

We define a *spanningtree* in our problem as a collection of lines in the graph forming no closed loops, containing a path between any two points of the graph covering $m^2 = n$ vertices and $n - 1$ edges. Spanning trees in a graph can have many shapes or patterns.

To calculate a number of spanning trees in G, let G be a graph with vertices labeled $v_1, \ldots, v_n$. We form an $n \times n$ matrix tree $T = [t_{ij}]$ as follows.

- If $i = j$, then t_{ii} is the number of lines to v_i in the graph.
- If $i \neq j$, then $t_{ij} = 0$ if there is no line between v_i and v_j in G.
- If $i \neq j$, then $t_{ij} = -1$ if there is a line between v_i and v_j in G.

Then, the number of spanning trees in G is given by

$$\mathit{cofactor\ of\ } a_{ij} = (-1)^{i+j} M_{ij}$$

where M_{ij} is the determinant of the $(n-1) \times (n-1)$ matrix obtained by deleting row i and column j of the matrix tree T. Evaluate any cofactor of T yields the same result.

Example 1: A matrix tree T of 2×2 modified mesh graph is

$$T = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{pmatrix}.$$

Evaluate any cofactor of T. For example, covering up row 1 and column 1, we have

$$(-1)^{1+1} M_{11} = \begin{vmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{vmatrix} = 16.$$

Therefore, a number of spanning tree is 16.

Example 2: A matrix tree T of 3×3 modified mesh graph is

$$T = \begin{pmatrix} 3 & -1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 5 & -1 & -1 & -1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 3 & 0 & -1 & -1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 5 & -1 & 0 & -1 & -1 & 0 \\ -1 & -1 & -1 & -1 & 8 & -1 & -1 & -1 & -1 \\ 0 & -1 & -1 & 0 & -1 & 5 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 & 0 & 3 & -1 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & -1 & 5 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & -1 & 3 \end{pmatrix}.$$

Evaluate any cofactor of T will yield the same value which is the number of spanning trees.

Hint: Determinant calculation using elementary row operation method:

To calculate a determinant you need to transform an original matrix into an upper triangular matrix (all elements below main diagonal are zero). Reduce the matrix to row echelon form using elementary row operations so that all the elements below diagonal are zero. Then multiply the main diagonal elements of a matrix to get the determinant value.

Your task is to write a program to find the number of spanning trees for a given size $m \times m$ of a modified mesh graph.

Input

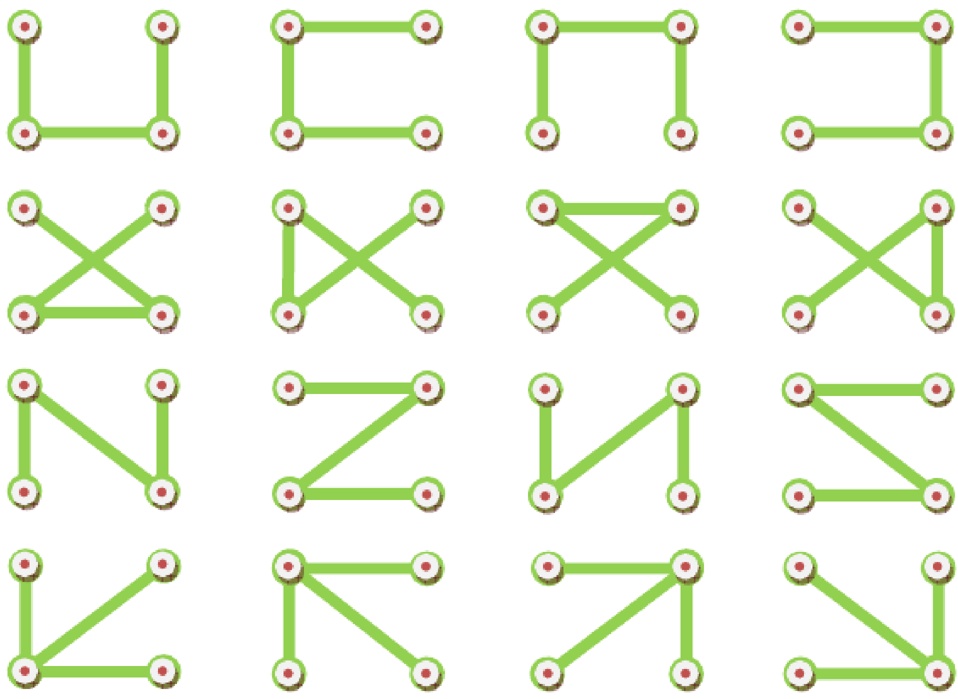
The first line of the input contains an integer N $(1 \leq N \leq 5)$ denoting the number of test cases. After that N test cases follow. Each test case contains a single integer m $(2 \leq m \leq 6)$ denoting a size of an $m \times m$ modified mesh graph.

Output

For each test case, your program should output the number of spanning trees, which indicates how many patterns of spanning trees there are for a given modified mesh graph. Print out one line per test case, respectively.

Explanation

In the sample input below, there is only 1 test case ($N = 1$). For example, in the first test case there are 16 different spanning trees for $m = 2$. These spanning trees are listed below. Note that your program is NOT required to draw all possible patterns.



Sample Input

```
1
2
```

Sample Output

```
16
```