A bishop is a piece used in the game of chess which is played on a board of square grids. A bishop can only move diagonally from its current position and two bishops attack each other if one is on the path of the other. In the following figure, the dark squares represent the reachable locations for bishop B_1 form its current position. The figure also shows that the bishops B_1 and $_B2$ are in attacking positions whereas B_1 and B_3 are not. B_2 and B_3 are also in non-attacking positions.
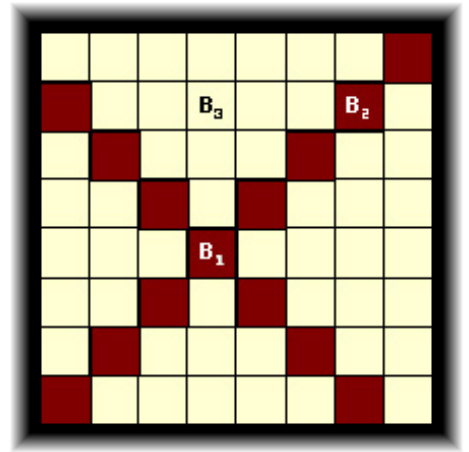
Now, given two numbers n and k, your job is to determine the number of ways one can put k bishops on an $n \times n$ chessboard so that no two of them are in attacking positions.

Input

The input file may contain multiple test cases. Each test case occupies a single line in the input file and contains two integers n ($1 \leq n \leq 30$) and k ($0 \leq k \leq n_2$).

A test case containing two zeros for n and k terminates the input and you won't need to process this particular input.

Output

For each test case in the input print a line containing the total number of ways one can put the given number of bishops on a chessboard of the given size so that no two of them are in attacking positions. You may safely assume that this number will be less than 10^{15}.

Sample Input

```
8 6
4 4
20 40
30 5
0 0
```

Sample Output

```
5599888
260
0
3127859642656
```