Problem F. Neural Network Counting

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	512 megabytes

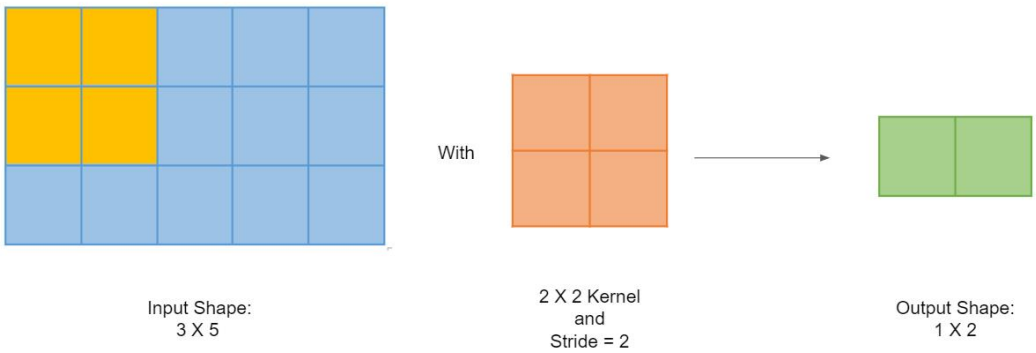
An artificial neural network could be modelled as a directed acyclic graph. The edges in the graph represent bundles of artificial neural signals, where a bundle of neural signals can be represented by a tenor in mathematics (or informally, a multi-dimensional array). The nodes in the graph represent the processing operations for the bundles of neural signals.

In this problem, the neural network only processes bundles of neural signals in two-dimensional tensors (i.e., matrices), and it only contains three types of processing operations: input operation, multiply operation, and convolution operation. We will explain them in detail. A processing operation consumes one or two bundles of signals and generates one bundle of signals. In our problem, the whole neural network contains one or more input operations and only allows one single bundle of output signals (neural signals not processed by any operations).

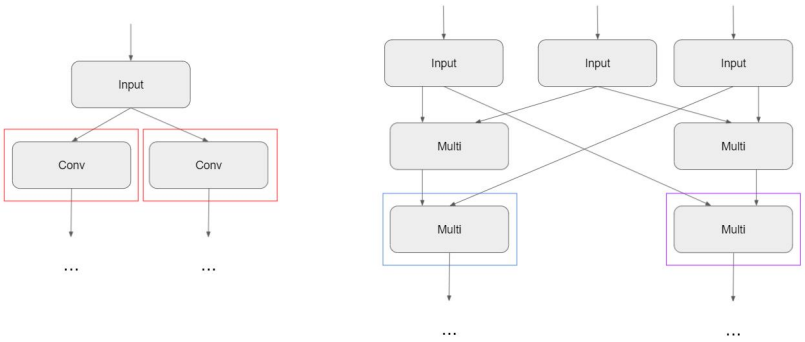
The input operation generate a tensor from the actual input stream of the neural network (which is a sequence of tensors). The actual input data are not modelled as an edge in the graph; instead, the node for the input operation has a zero in-degree and has one out-edge to model or replicate the bundle of input signals as a tensor. The input operation will produce the tensor one by one from the input stream.

The multiply operation takes two tensors as its inputs, and then it multiplies these two tensors and generates a product tensor. In our problem, all tensors are matrices, so the sizes of the two input matrices must be legal for the matrix multiplication. Note that the order of inputs matters, unless both input tensors are virtually the same one; in other words, A multiplies B is likely different from B multiplies A, if A and B are not the same.

The convolution operation takes one tensor as its input and contains some attributes such as kernel size and stride. To simplify the problem, here we only allow the convolution operation with a 2×2 kernel size, a stride of 2, and no padding (do not add 0 outside). What we concern about is the tensor shape: if the input tensor's shape is (x, y) (the size of the first dimension is x and the size of the second dimension is y), then its output tensor's shape is $(\lfloor \frac{x}{2} \rfloor, \lfloor \frac{y}{2} \rfloor)$. Note that we do not allow the tensor with empty size, so each dimension of the output tensor should be greater than 0.



Besides, in this problem, we do not allow the neural network contains repeated nodes. Two nodes are called repeated only if their input(s) are actually the same node(s). For example, the left neural network contains repeated nodes (the nodes in red box). Note, as another an example in right neural network, the node in blue box and the node in purple box are different, because their inputs are different nodes.



We call the input format of a neural network the shape of its input tensor(s). For example, if a neural network has two input tensor(s), the shape of the first tensor is (x_1, y_1), and the shape of the second tensor is (x_2, y_2), then we call the input format of this neural network as $(x_1, y_1), (x_2, y_2)$. We call two neural networks are different if they have different input formats or their graphs models are different.

For a given number N and D, assuming the upper bound of tensor's dimension size is 2^D-1, we would like to know the number of different neural networks with N nodes. As this number may be large, output its remainder when divided by 1000000007.

Input

The first line of the input contains one integer T ($1 \le T \le 10$) — the number of test cases. Then T test cases follow. Each test case contains one line with two integers N and D ($N \le 7$ and $D \le 50$).

Output

For each test case, print one line as its corresponding answer.

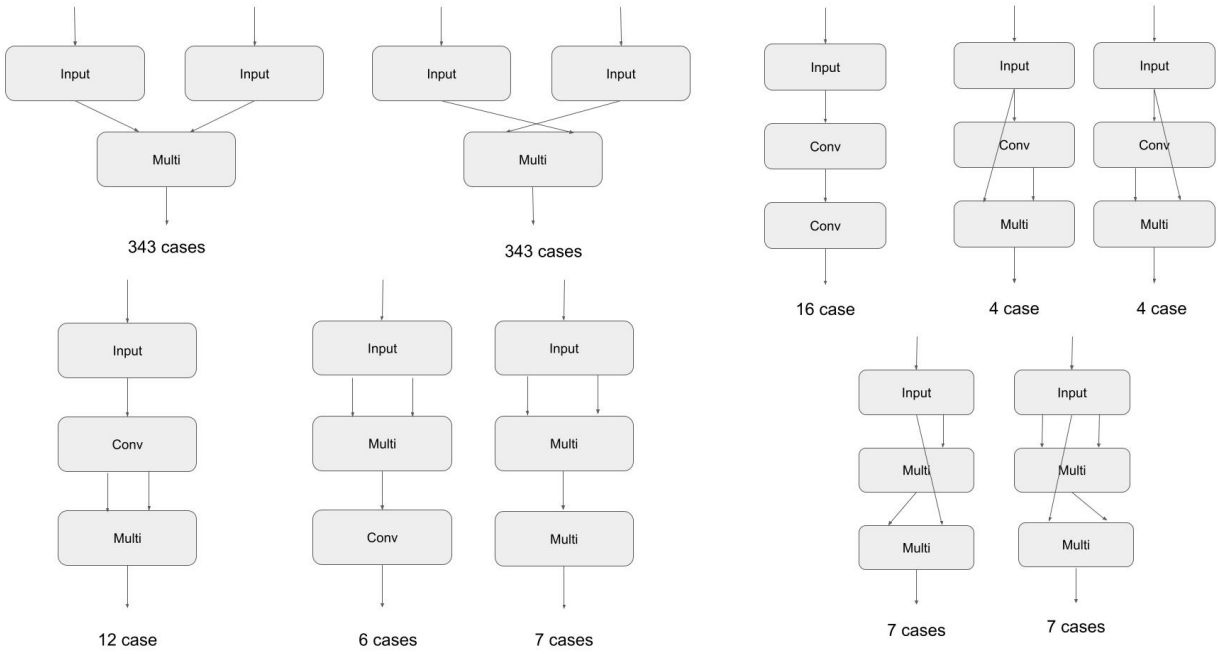
Example

standard input	standard output
3 2 2 3 3 4 3	7 749 2391

Note

For the first test case, it contains the following different neural networks: an input node connects with a multiply node with input tensor's shape as 1×1, or 2×2, or 3×3; an input node connects with a convolution node with input tensor's shape as 2×2, or 2×3, or 3×2, or 3×3.

For the second test case, it contains the following different neural networks, the number below each network represent the number of input formats on that graph.


Input
Input
Multi
343 cases
Input
Input
Multi
343 cases
Input
Conv
Conv
16 case
Input
Conv
Multi
4 case
Input
Conv
Multi
4 case
Input
Conv
Multi
12 case
Input
Multi
Conv
6 cases
Input
Multi
Multi
7 cases
Input
Multi
Multi
7 cases
Input
Multi
Multi
7 cases