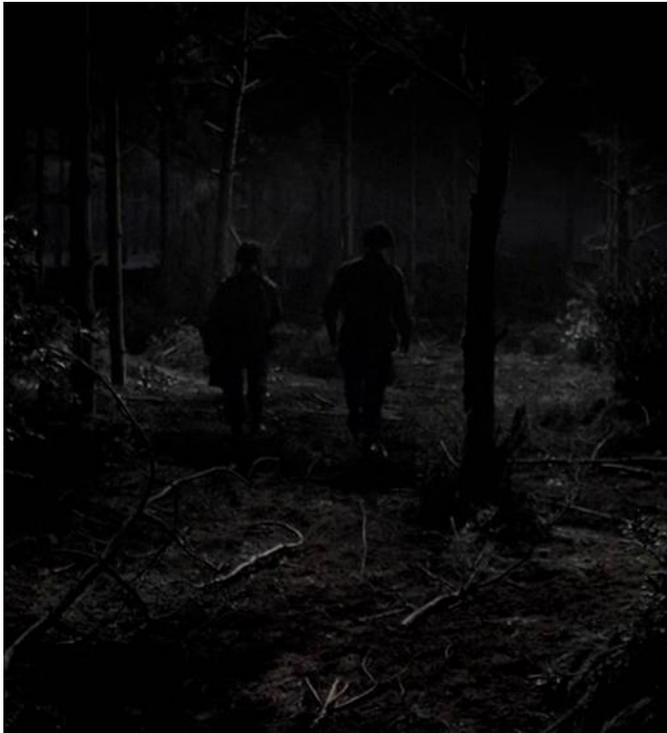


WW2: World War 2**Problem Statement :**

Richard Winters: We're not lost, Private... we're in Normandy.



In WW2, allies have placed their mines on the ground which can be considered as a grid of size $N \times M$. Rows are numbered by integers from **1** to **N** and columns are numbered by integers from **1** to **M**. Each cell (i,j) has a mine in it if $(i+j)$ is divisible by **2**. All the other cells are empty.

Two cells are neighbors if they share a boundary or a corner. Your aim is to move from the first row to the last row using some special rules that the automatic mine placing system follows. The mines won't blow if the following rules are followed:

- If your current row number is odd, from a 'mined' cell (ie. cell having a mine in it) you can move only to a neighboring 'mined' cells in the next row, and similarly, from a 'not mined' (ie. an empty cell) cell you can only move to a neighboring 'not mined' cell in the next row.
- If your current row number is even, you can move to any neighboring cell in the next row, irrespective of the mining.

You have to find the number of ways to reach row number **N** starting

from row number **1**, modulo 10^9+7 . Two ways are different, if there is at least one different cell on the paths.

Input

First line contains **T**, the number of testcases. Each testcase consists of **N** and **M** in one line.

Output

For each testcase, print the required answer in one line.

Constraints

- $1 \leq T \leq 10$
- $1 \leq N \leq 10^9$
- $1 \leq M \leq 30$

Example

Input:

```
2
1 5
2 5
```

Output:

```
5
8
```

Explanation

Example case 1. You can start at any of the 5 cells available.

Example case 2.

8 ways are:

```
(1,1) to (2,2)
(1,2) to (2,1)
(1,2) to (2,3)
(1,3) to (2,2)
(1,3) to (2,4)
(1,4) to (2,3)
(1,4) to (2,5)
(1,5) to (2,4)
```