Problem 6: Fib-Query Fiasco
6+7=13 Points

Problem ID: `fibsums`
Rank: `2+3`

Introduction

After messing up some configurations for time traveling, Youngmin finds himself in 13th century Italy! He meets Fibonacci there, who gives him a hard task (and smallpox).

Problem Statement

You are given a length $\mathbf{N}$ array $\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_N}$ with all elements initialized to zero. You are also given $\mathbf{Q}$ fib-queries, the i^{th} of which involves values $\mathbf{L_i}$ and $\mathbf{R_i}$. Formally, the i^{th} fib-query involves us setting:

$$\mathbf{A_k} = \mathbf{A_k} + f(k - \mathbf{L_i} + 1) \pmod{998244353}$$

for all $\mathbf{L_i} \le k \le \mathbf{R_i}$, where $f(k - \mathbf{L}_i + 1)$ denotes the $(k - \mathbf{L_i} + 1)^{\text{th}}$ Fibonacci number. In other words, the i^{th} fib-query involves us adding the k^{th} Fibonacci number to the k^{th} element starting from $\mathbf{L_i}$, up to and including element $\mathbf{R_i}$.

The i^{th} Fibonacci number $f(i)$ is defined recursively as $f(1) = f(2) = 1$ and $f(i) = f(i - 1) + f(i - 2)$ for $i \ge 3$. These form the Fibonacci sequence 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, and so on.

Find and output the resulting array $\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_N}$ after all $\mathbf{Q}$ queries.

Input Format

The first line of the input contains an integer $\mathbf{T}$ denoting the number of test cases that follow.
For each test case:

- The first line contains two space-separated integers $\mathbf{N}$ $\mathbf{Q}$, where:
  - $\mathbf{N}$ denotes the length of the array $\mathbf{A_1}$, $\mathbf{A_2}$, … , $\mathbf{A_N}$
  - $\mathbf{Q}$ denotes the number of fib-queries.
- The next $\mathbf{Q}$ lines each contain two space-separated integers $\mathbf{L_i}$ $\mathbf{R_i}$ representing the i^{th} fib-query.

Output Format

For each test case, output a space-separated list $\mathbf{N}$ integers $\mathbf{A_1}$ $\mathbf{A_2}$ … $\mathbf{A_N}$.

Constraints

$1 \le \mathbf{L_i} \le \mathbf{R_i} \le \mathbf{N}$ for all fib-queries.

Main Test Set

$1 \le \mathbf{T} \le 10$
$1 \le \mathbf{N} \le 30$
$1 \le \mathbf{Q} \le 100$

Bonus Test Set

$1 \le \mathbf{T} \le 100$
$1 \le \mathbf{N}, \mathbf{Q} \le 10^5$
The sum of $\mathbf{N}$ across all test cases in an input file does not exceed 10^5.
The sum of $\mathbf{Q}$ across all test cases in an input file does not exceed 10^5.

Sample Test Cases

Main Sample Input *Download*

```
3
5 3
1 5
3 5
1 2
10 7
3 9
9 10
1 7
5 8
1 10
3 9
2 4
4 5
1 4
2 2
2 2
1 3
3 3
```

Main Sample Output *Download*

```
2 2 3 4 7
2 3 7 10 15 23 38 40 61 56
2 4 5 3
```

Main Sample Explanations

Test Case #1:

The sequence of fib-queries plays out as follows:

L_i	R_i	Values Added				
1	5	1	1	2	3	5
3	5			1	1	2
1	2	1	1			
Total		2	2	3	4	7

The final values of $A_1, A_2, \ldots, A_N$ after all fib-queries have completed are 2, 2, 3, 4, and 7. Therefore, the output is: `2 2 3 4 7`

Test Case #2:

The values of $\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_N}$ after each fib-query has completed are as follows:

$\mathbf{L_i}$	$\mathbf{R_i}$	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$	$\mathbf{A_4}$	$\mathbf{A_5}$	$\mathbf{A_6}$	$\mathbf{A_7}$	$\mathbf{A_8}$	$\mathbf{A_9}$	$\mathbf{A_{10}}$
-	-	0	0	0	0	0	0	0	0	0	0
3	9	0	0	1	1	2	3	5	8	13	0
9	10	0	0	1	1	2	3	5	8	14	1
1	7	1	1	3	4	7	11	18	8	14	1
5	8	1	1	3	4	8	12	20	11	14	1
1	10	2	2	5	7	13	20	33	32	48	56
3	9	2	2	6	8	15	23	38	40	61	56
2	4	2	3	7	10	15	23	38	40	61	56

*The highlighted cells outline which elements were updated during the fib-query.

The final values of $\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_N}$ after all fib-queries have completed are 2, 3, 7, 10, 15, 23, 38, 40, 61, and 56. Therefore, the output is: `2 3 7 10 15 23 38 40 61 56`

Test Case #3:

The values of $\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_N}$ after each fib-query has completed are as follows:

$\mathbf{L_i}$	$\mathbf{R_i}$	$\mathbf{A_1}$	$\mathbf{A_2}$	$\mathbf{A_3}$	$\mathbf{A_4}$
-	-	0	0	0	0
1	4	1	1	2	3
2	2	1	2	2	3
2	2	1	3	2	3
1	3	2	4	4	3
3	3	2	4	5	3

*The highlighted cells outline which elements were updated during the fib-query.

The final values of $\mathbf{A_1}, \mathbf{A_2}, \ldots, \mathbf{A_N}$ after all fib-queries have completed are 2, 4, 5, and 3. Therefore, the output is: `2 4 5 3`

Bonus Sample Input *Download*

```
1
40 10
1 40
1 40
1 40
1 40
1 40
1 40
1 40
1 40
1 40
1 40
```

Bonus Sample Output *Download*

```
10 10 20 30 50 80 130 210 340 550 890 1440 2330 3770 6100 9870 15970
25840 41810 67650 109460 177110 286570 463680 750250 1213930 1964180
3178110 5142290 8320400 13462690 21783090 35245780 57028870 92274650
149303520 241578170 390881690 632459860 25097197
```

Note that while the output appears to span multiple lines on this page, the output for this test case should only be contained within one line.

Bonus Sample Explanations

For test case 1, note that the output is taken modulo 998244353, causing $\mathbf{A_{40}}$ to equal 25097197. If we didn't take the modulus of our answers, $\mathbf{A_{40}}$ would instead equal 1023341550.