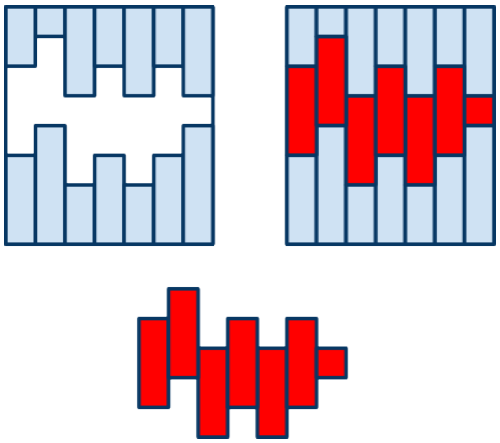
COCI 2009/2010 **Task KLETVA**

5th round, 6. March 2010. 1 second / 32 MB / 70 points

As punishment for destroying half of his city with his monster truck, Mirko now has to pay off his dept to society. He works as an assistant for a famous archaeologist. One of his duties include crafting keys for ancient document boxes.

In ancient times document boxes were locked using elaborate mechanisms with interesting locks. Each lock is **L** centimeters long and **W** centimeters wide and consists of three parts, the upper edge, the lower edge and the empty area between them. Both edges can be represented as a sequence of **L nonnegative integers**: r_1 r_2 r_3 ... r_L. Each number in sequence represents the width of edge at that point.

The key for each lock is a small clay tab, fitting perfectly in the area between edges. This image shows a 7 cm long, 8 cm wide lock along with the corresponding key.



The sequence representing the upper edge is [2, 1, 3, 2, 3, 2, 3], and the sequence representing the lower edge is [3, 4, 2, 3, 2, 3, 4]. Mirko noticed that some keys open more than one lock. Making keys is tedious work so Mirko asked you to find out what is the **minimal number** of different keys he needs to make and still be able to open all of the locks.

INPUT

First line of input contains three integers, **W** ($1 \leq \mathbf{W} \leq 10^8$), width of all locks, **L** ($1 \leq \mathbf{L} \leq 1000$) length of all locks, and **N** ($1 \leq \mathbf{N} \leq 100$), number of different locks.

Next 2**N** lines describe all locks. Each line contains exactly **L** numbers smaller than **W**. Each pair of lines describes one lock. The first line in one pair describes the upper edge, and the second line the lower edge. **There shall always be at least 1 cm of empty space between both edges on all locks.**

OUTPUT

The first and only line of input should contain a single integer, the minimal number of different keys Mirko needs to craft.

SAMPLE TEST CASES

Input:	**Input:**	**Input:**
8 7 2 2 1 3 2 3 2 3 3 4 2 3 2 3 4 3 2 4 3 4 3 4 2 3 1 2 1 2 3	8 4 4 3 3 3 3 3 3 3 3 2 2 2 2 4 4 4 4 1 2 3 4 4 3 2 1 1 1 1 1 5 5 5 5	100000000 2 2 88888888 88888888 4 4 4 4 88888888 88888888
Output: 1	**Output:** 2	**Output:** 1