

# Problem B - ARChitecture

## Description

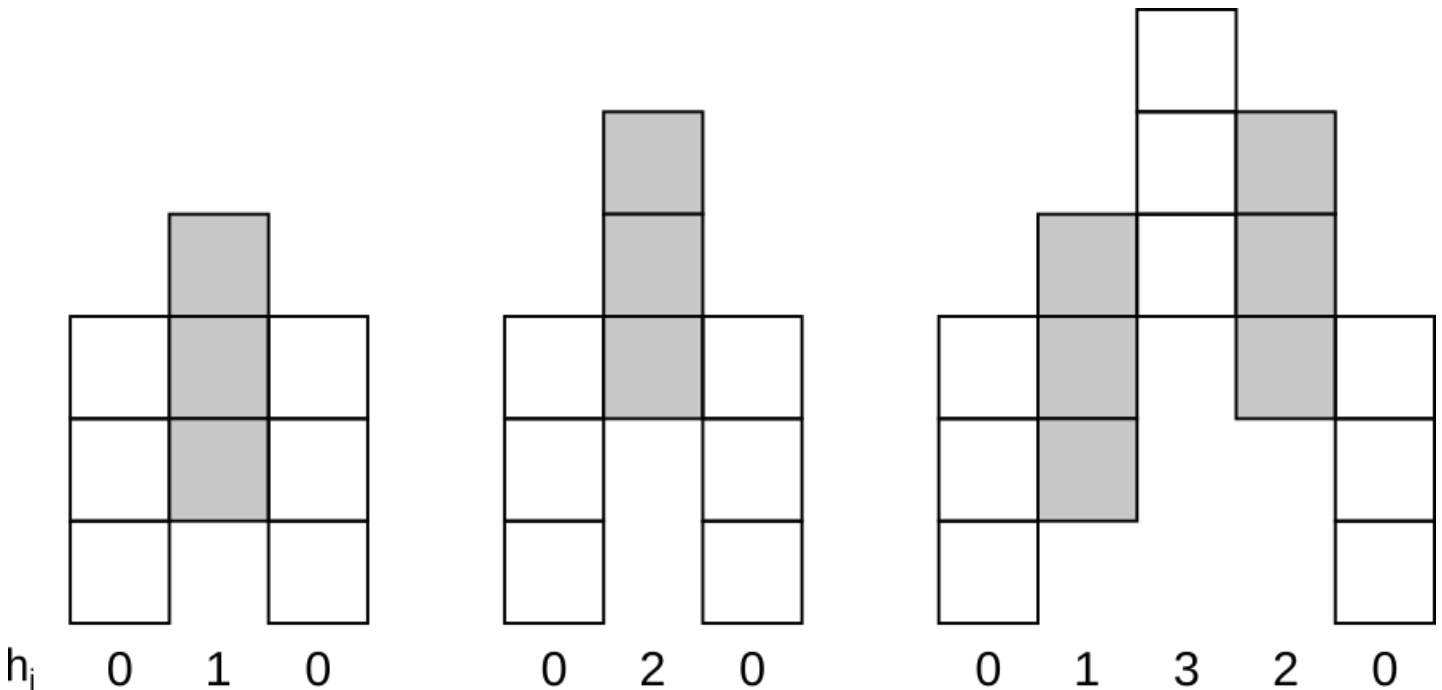
You are given  $n$  Lego blocks of height  $h$ , and you want to build an arc using some (or all) of those blocks in your room which has a ceiling of height  $H$ , so the top of your arc cannot be higher than that. Assume that the floor starts at height 0 and that width is not a problem.

For your Lego construction to be considered an arc it needs to conform to the following rules:

1. An arc consists of  $3 \leq k \leq n$  blocks placed consecutively, such that the index  $i = 1$  denotes the first block,  $i = 2$  the second block, and so on until the last block  $i = k$ ;
2. The first and last blocks must be placed on the floor, that is  $h_1 = 0$  and  $h_k = 0$ , where  $h_i$  denotes the height from the floor to the base of the block;
3. The  $i$ th block must share at least 1 height with its neighbors, formally  $h_{i+1} - h < h_i < h_{i+1} + h$  for  $i > 1$ ;
4. The values of  $h_i$  should be monotonically increasing, up to a certain block, and then monotonically decreasing.

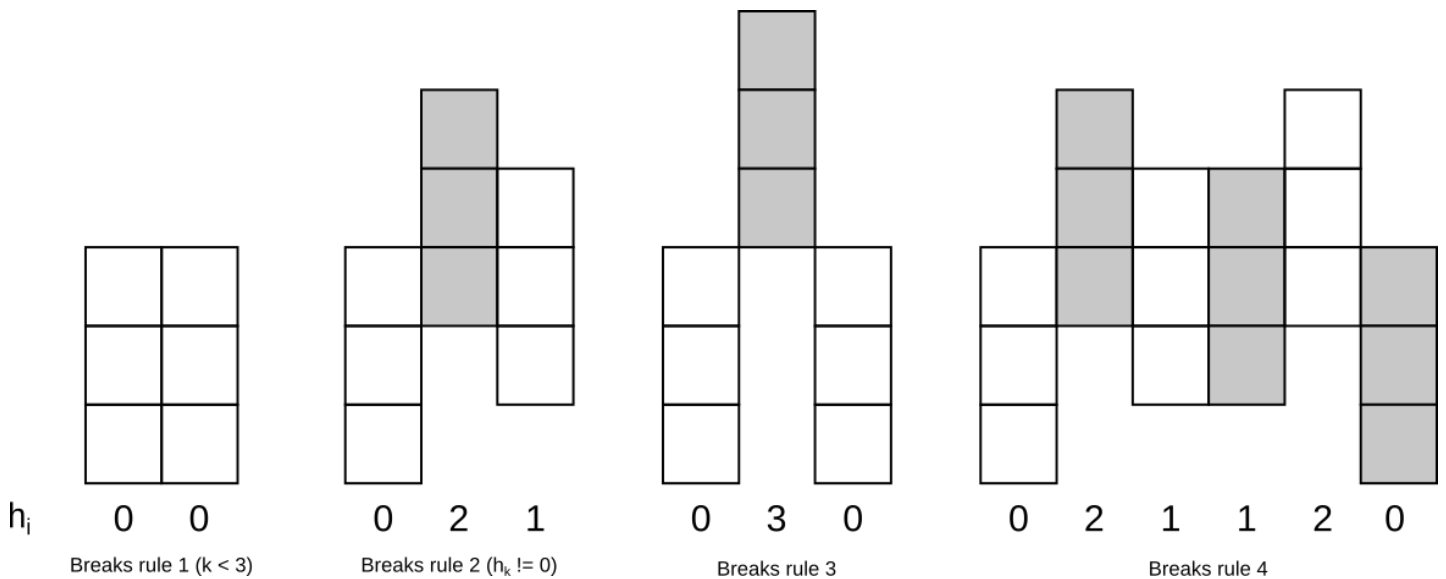
Formally, there exists a block  $l$  such that  $h_1 < \dots < h_l > \dots > h_k$ . Note that two consecutive blocks cannot have the same value of  $h_i$ .

To give some examples the following three constructions are valid arcs for  $n = 5$ ,  $h = 3$  and  $H = 6$ .



However, if  $H = 4$  only the left-most construction would be valid.

Moreover the following constructions are **not** valid arcs since they break at least one rule.



When building your arc you started wondering how many distinct arcs you could build given your constraints. Your task, is to compute the number of distinct arcs that can be built for the given values  $n$ ,  $h$  and  $H$ . Since this number can be very large, you need to report your answer in modulo 1000000007.

You may consider the following modulo arithmetic operations:

```
int mod_abs(int a, int mod) {
    return ((a % mod) + mod) % mod;
}

int mod_add(int a, int b, int mod) {
    return (mod_abs(a, mod) + mod_abs(b, mod)) % mod;
}

int mod_sub(int a, int b, int mod) {
    return mod_add(a, -b, mod);
}
```

## Input

The first line contains the number of test cases  $t$ . Then,  $t$  test cases follow.

For each test case, you are given three space-separated integers in a single line denoting the variables  $n$ ,  $h$  and  $H$  respectively.

## Limits

- $t \leq 20$
- $n \leq 500$
- $h \leq 500$
- $H \leq 60000$

## Output

For each test case you need to output the number of distinct arcs you can build modulo 1000000007.

## Examples

### Input (Example 1)

```
7
3 3 3
3 3 4
3 3 5
3 3 6
4 3 7
```

5 5 10  
8 4 30

### Output (Example 1)

0  
1  
2  
2  
4  
54  
819

### Input (Example 2)

1  
32 3 100

### Output (Example 2)

431655757