

# Jellyfish and Undertale

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         256 megabytes

Flowey has planted a bomb in Snowdin!

The bomb has a timer that is initially set to  $b$ . Every second, the timer will decrease by 1. When the timer reaches 0, the bomb will explode! To give the residents of Snowdin enough time to evacuate, you will need to delay the bomb from exploding for as long as possible.

You have  $n$  tools. Each tool can only be used **at most** once. If you use the  $i$ -th tool, the timer will increase by  $x_i$ . However, if the timer is changed to an integer larger than  $a$ , the timer will be set to  $a$  due to a bug.

More specifically, the following events will happen every second in the following order:

1. You will choose some (possibly none) of your tools that have not been used before. If you choose the  $i$ -th tool, and the bomb's timer is currently set to  $c$ , the timer will be changed to  $\min(c + x_i, a)$ .
2. The timer decreases by 1.
3. If the timer reaches 0, the bomb explodes.

Jellyfish now wants to know the maximum time in seconds until the bomb explodes if the tools are used optimally.

## Input

Each test contains multiple test cases. The first line contains the number of test cases  $t$  ( $1 \leq t \leq 2000$ ). The description of the test cases follows.

The first line of each test case contains three integers  $a$ ,  $b$  and  $n$  ( $1 \leq b \leq a \leq 10^9$ ,  $1 \leq n \leq 100$ ) — the maximum value of the bomb's timer, the initial value of the timer of the bomb and the number of tools.

The second line of each test contains  $n$  integers  $x_1, x_2, \dots, x_n$  ( $1 \leq x_i \leq 10^9$ ) — the number the timer can increase by using the  $i$ -th tool.

Note that the sum of  $n$  over all test cases is not bounded.

## Output

For each test case, output a single integer — the maximum time in seconds until the bomb explodes.

## Example

standard input	standard output
2	9
5 3 3	21
1 1 7	
7 1 5	
1 2 5 6 8	

## Note

Let  $c$  denote the value of the bomb's timer. In the first test case:

- Second 1: choose tool 1 and 2 at this second, then  $c = 5$ ; the timer decreases by 1, then  $c = 4$ .
- Second 2: the timer decreases by 1, then  $c = 3$ .

- Second 3: the timer decreases by 1, then  $c = 2$ .
- Second 4: the timer decreases by 1, then  $c = 1$ .
- Second 5: choose tool 3, then  $c = 5$ ; the timer decreases by 1, then  $c = 4$ .
- Second 6: the timer decreases by 1, then  $c = 3$ .
- Second 7: the timer decreases by 1, then  $c = 2$ .
- Second 8: the timer decreases by 1, then  $c = 1$ .
- Second 9: the timer decreases by 1, then  $c = 0$ . The bomb explodes.

It can be proved that there is no way to use the tools such that the bomb explodes after more than 9 seconds.