
Jump

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

Languages allowed: C, C++, Java

Construction work is ongoing at IIIT-Delhi, with new buildings in process of being built. Being the 'engineer' you are, you are tasked with inspecting the construction work in one of the buildings. During one such inspection, you are examining a flight of stairs and you're trying to figure out if it's too long to climb up.

There are n stairs on this flight (from 1 to n), and every stair has a number assigned to it. At any stair s , two possible moves can be made: go to the next one, or directly jump as many stairs as the number on that stair, i.e if the number on the stair is p , you can either go to $s + 1$, or jump directly to $s + p$ (if $p < 0$ then obviously you jump down instead of up).

Return the number of jumps required in order to get to the next floor (i.e get to step $n + 1$) and expend the least energy (assuming any move you make takes the same effort). If there is no way to reach the next floor, return -1. Note that any jump which takes you to a step $> n + 1$ is not allowed.

Input

The first line consists of t , the number of test cases ($1 \leq t \leq 500$). For every case, the first line takes n (the number of stairs, $1 \leq n \leq 1500$), followed by a line consisting of space separated integers denoting the numbers p_i assigned to each of the n steps ($-1000 \leq p_i \leq 1000$, $p_i \neq 0$).

Output

Return a single number, denoting the number of jumps as mentioned.

Example

standard input	standard output
2	3
3	4
1 -1 3	
10	
2 4 3 -3 4 2 4 1 1 2	

Note

In Case 1, a solution can be Stair 1 \rightarrow Stair 2 \rightarrow Stair 3 \rightarrow Stair 4, i.e 3 moves.

In Case 2, a solution can be Stair 1 \rightarrow Stair 2 \rightarrow Stair 6 \rightarrow Stair 7 \rightarrow Stair 11, i.e 4 moves.