

Walky Walk

Bob is developing a new game. In the game, there will be N blocks, numbered from 1 to N . Initially a player will be standing at block numbered 1 and can jump to other blocks. The player's first jump should be from block 1 to block 2. Then the next jumps can be:

1. If the jump is in forward direction, the length must be one block longer than the previous jump.
2. If the jump is in backward direction, the length must be same as the previous jump.

For example, after the first jump (when the player is at block 2), s/he can jump back to block 1 or block 4.

Now each block contains a surcharge, if you jump on a certain block you have to pay that block's surcharge (it doesn't matter whether s/he visited the block before). Now the player's target will be to go from block 1 to block N by paying minimum surcharge possible. Can you find that minimum surcharge? As the player is initially at block 1 and didn't jump into the block 1 so s/he will not pay the surcharge for block 1. But if in any subsequent jump one player jumps to block 1, s/he must pay.

Input:

Input starts with an integer T (≤ 10), denoting the number of test cases.

The first line of each case contains the integer N , $2 \leq N \leq 1000$, the number of blocks. Then the next line contains N integers separated by space, where i 'th integer ($1 \leq i \leq N$) denotes the surcharge of i 'th block. These integers will be between 1 and 500

Output:

For each case, print the case number and the minimum surcharge a players needs to give to go from block 1 to block N . See the samples for exact formatting.

Sample Input	Sample Output
2 6 1 2 3 4 5 6 8 2 3 4 3 1 6 1 4	Case 1: 12 Case 2: 14

Explanations

In the first test case, after jumping to square 2, Player jumps back to square 1. From there s/he can jump to square 3 and then to 6.

Limits:

Language	Time	Memory
C	1 Second	50MB
C++	1 Second	50MB
Java	2 Second	50MB
C#	2 Second	50MB