5926. Time Needed to Buy Tickets

My Submissions (/contest/weekly-contest-267/problems/time-needed-to-buy-tickets/submissions/) Back to Contest (/contest/weekly-contest-267/)

There are `n` people in a line queuing to buy tickets, where the 0^{th} person is at the **front** of the line and the $(n-1)^{th}$ person is at the **back** of the line.

You are given a **0-indexed** integer array `tickets` of length `n` where the number of tickets that the i^{th} person would like to buy is `tickets[i]`.

Each person takes **exactly 1 second** to buy a ticket. A person can only buy **1 ticket at a time** and has to go back to **the end** of the line (which happens **instantaneously**) in order to buy more tickets. If a person does not have any tickets left to buy, the person will **leave** the line.

Return *the **time taken** for the person at position* `k` ***(0-indexed)*** *to finish buying tickets*.

User Accepted:	8
User Tried:	14
Total Accepted:	8
Total Submissions:	14
Difficulty:	Easy

Example 1:

```
Input: tickets = [2,3,2], k = 2
Output: 6
Explanation:
- In the first pass, everyone in the line buys a ticket and the line becomes [1, 2, 1].
- In the second pass, everyone in the line buys a ticket and the line becomes [0, 1, 0].
The person at position 2 has successfully bought 2 tickets and it took 3 + 3 = 6 seconds.
```

Example 2:

```
Input: tickets = [5,1,1,1], k = 0
Output: 8
Explanation:
- In the first pass, everyone in the line buys a ticket and the line becomes [4, 0, 0, 0].
- In the next 4 passes, only the person in position 0 is buying tickets.
The person at position 0 has successfully bought 5 tickets and it took 4 + 1 + 1 + 1 + 1 = 8 seconds.
```

Constraints:

- `n == tickets.length`
- `1 <= n <= 100`
- `1 <= tickets[i] <= 100`
- `0 <= k < n`

JavaScript

```
/**
 * @param {number[]} tickets
 * @param {number} k
 * @return {number}
 */
var timeRequiredToBuy = function(tickets, k) {

};
```