Linear search algorithm

Consider the searching problem:

- Input: A sequence of n numbers $A = \{a_1, a_2, ..., a_n\}$ and a value v.
- Output: An index i such that v = A[i] or the special value N if v does not appear in A.

You can find the implementation **here** or go to the next url: https://github.com/DiegoMendezMedina/C_Algorithms/blob/master/Search/linear_searching/implementations/searching_problem.c.

Pseudocode

1. **for** i = 0 **to** n-1
2. **if** $v == A[i]$
3. return i
4. return 'N'

Proof

Loop invariant:
At the start of each iteration of the **for** loop (lines 1-3), v was not found on the previous **i** values. **if** $v == A[i]$, **i** is returned and the **for** loop breaks. Otherwhise at the end of the loop **'N'** is returned.

Initialization:
When $i = 0$ there are no previous **i** values.
if $A[0] = v$; then **i** is return and the **for** loop breaks.

Maintenance:
There's another iteration which means that for all the previous value of **i** v was not found. If for the current value of **i** happens that $A[i] = v$ then **i** is returned and the **for** loop breaks.

Termination:
When the loop finishes **i** had browsed all the possible positions of the array and v was not found; then **'N'** is returned.