

HANDS-ON 2

2. Argue to selection sort correctness.

Ans:- Selection sort is a simple sorting algorithm that works by repeatedly finding the minimum element from the unsorted part of the array and swapping it with the first unsorted element.

Q. LOOP INVARIANT:- It's a condition that holds true before and after each iteration of a loop. For selection sort, we can define the loop invariant as follows:-

INVARIANT:- At the beginning of each iteration of the outer loop, the first $i-1$ elements are in sorted order.

⇒ Proof of Correctness:-

1. Initialization:- Before the first iteration, $i=1$, and there are no elements in the sorted part of the array.

Therefore, the loop invariant holds.

2. Maintenance:- Here the inner loop finds the minimum element among the elements from index i to $n-1$.

⇒ We then swap the minimum element with the element at index ' i ', that in turn places the smallest remaining element in its correct position.

\Rightarrow Now, as the first $i-1$ elements are already sorted, & the smallest remaining element is now placed at i^{th} position, the first i elements are now in sorted order. Therefore, the loop invariant holds at the beginning of the next iteration.

3. Termination:- After $n-1$ iterations of the outer loop, $i=n$. Now, the entire array is sorted, and the loop invariant holds.

\Rightarrow Hence, by using the loop invariant, we can argue that selection sort correctly sorts the array.

Proof of correctness

- **Initialization:** Before the first iteration, $i=1$ and there is one element in the array. Therefore, the loop invariant holds.

- **Maintenance:** Here the inner loop finds the minimum element among the elements from index i to $n-1$. We then swap the minimum element with the element at index i . This maintains the loop invariant.