**2.4 OPTIMIZED FUNCTION**

**AIM**

To implement and test an optimized sorting algorithm (Selection Sort) on a given list and analyze its performance.

**ALGORITHM**

1. Start with the entire unsorted list.

2. For each index i from 0 to n-1:

* Assume the element at i is the minimum.
* Traverse the rest of the list (i+1 to n-1) to find the true minimum element.
* If a smaller element is found, update the minimum index.
* Swap the found minimum with the element at index i.

3. Repeat until the whole list is sorted.

**PROGRAM**



Input:

[64, 25, 12, 22, 11]

Output:

A screenshot of a computer

AI-generated content may be incorrect.

**RESULT:**

Thus the program is successfully executed and the output is verified.

**PERFORMANCE ANALYSIS:**

* Time Complexity:
  + Best Case: comparisons, swaps (if already sorted).
  + Worst Case: comparisons and swaps.
* Space Complexity: (in-place sorting).