**2.3 OPTIMIZED BUBBLE SORT FOR A RANDOM ARRAY**

**AIM**

To sort a given array using Bubble Sort with an early stopping condition if the array becomes sorted before completing all passes.

**ALGORITHM**

1. Start from the first element.

2. Compare adjacent elements, swapping them if they are in the wrong order.

3. After each pass, the largest element in the unsorted region moves to its correct position.

4. Keep track of whether any swaps occurred in the current pass.

5. If no swaps are made, stop early since the list is already sorted.

**PROGRAM**

A screenshot of a computer program

AI-generated content may be incorrect.

Input:

[5, 2, 9, 1, 5, 6]

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 (already sorted): O(n)
* Worst/Average Case: O(n²)
* Space Complexity:
  + Uses only a few variables, so O(1) (constant extra space).