Analysis of Sorting Algorithms for Customer Orders

# 1. Introduction

This document analyzes two sorting algorithms—Bubble Sort and Quick Sort—for sorting customer orders based on total order price. The goal is to prioritize high-value orders in an e-commerce platform.

# 2. Sorting Algorithm Description

## 2.1 Bubble Sort

Bubble Sort is a simple comparison-based algorithm that repeatedly compares and swaps adjacent elements if they are in the wrong order. It is easy to implement but inefficient for large datasets.

## 2.2 Quick Sort

Quick Sort is a divide-and-conquer algorithm that selects a pivot and partitions the array into sub-arrays, which are then sorted recursively. It is more efficient and widely used in practice.

# 3. Complexity Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Best Case | Average Case | Worst Case | Space Complexity |
| Bubble Sort | O(n) | O(n²) | O(n²) | O(1) |
| Quick Sort | O(n log n) | O(n log n) | O(n²) | O(log n) |

# 4. Recommendation

For sorting customer orders based on price, Quick Sort is generally preferred due to its superior performance on average. Bubble Sort should be avoided in production environments with large datasets, though it may be used for educational purposes or very small datasets.