Abstract

Insertion sort gives a time complexity of O(n) for the best case. In the worst case where the input is in the descending order fashion, the time complexity is O(n^2). In the case of arrays, shifting takes O(n^2) while in the case of linked lists comparison comes to O(n^2). Here a new way of sorting for the worst case problem is proposed by using arrays as data structure and taking more space. 2n spaces is taken where n is the number of elements and starts the
Increasing Time Efficiency of Insertion Sort for the Worst Case Scenario

insertion from (n-1)th location of the array. In this proposed technique the time complexity is \( O(n\log n) \) as compared to \( O(n^2) \) in the worst case.

References

- Mark Allen Weiss, Data Structures and Algorithm Analysis in C++: Pearson Addison-Wesley, 2006
- Michael A. Bender, "Insertion Sort is \( O(n\log n) \)," Third International Conference on Fun With Algorithms (FUN), Pages 16-23, 2004

Index Terms

Computer Science

Algorithm

Keywords

Insertion Sort Time Complexity Space Complexity.