

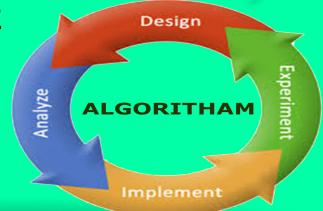
MADE BY

MUHAMMAD UNAIS
SHOAIB AHMED
HASAN SIDDIQUE



CS-211 Data Structures & Algorithms

17B-017-SE
17B-033-SE
17B-049-SE



COCKTAIL SHAKER SORT

INTRODUCTION

Cocktail shaker sort, also known as **Bidirectional Bubble Sort**, **cocktail sort**, **shaker sort** (which can also refer to a variant of selection sort), **ripple sort**, **shuffle sort**, or **shuttle sort**.

COMPLEXITY

Worst complexity: $O(n^2)$

Average complexity: $O(n^2)$

Best complexity: $O(n)$

ADDITIONAL INFORMATION

DIFFERENCES FROM BUBBLE SORT

Cocktail Sort is a variation of Bubble sort. The Bubble sort algorithm always traverses elements from left and moves the largest element to its correct position in first iteration and second largest in second iteration and so on. Cocktail Sort traverses through a given array in both directions alternatively.

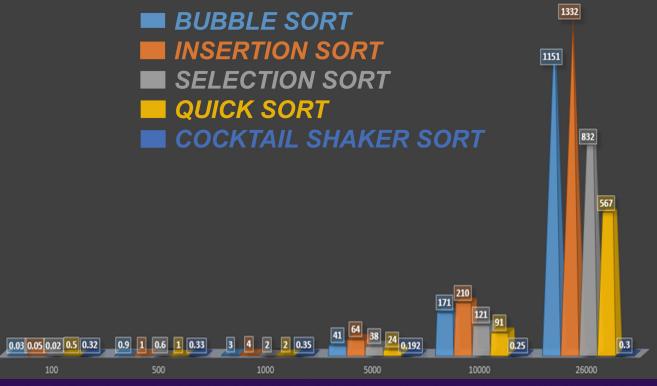
ADVANTAGES

- It solves the problem of **turtles** in bubble sorts.
- It provides only **marginal performance improvements**, and does not improve asymptotic performance; like the bubble sort.

GRAPH

SORTING ALGORITHM GRAPH

- BUBBLE SORT
- INSERTION SORT
- SELECTION SORT
- QUICK SORT
- COCKTAIL SHAKER SORT



PSEUDOCODE

```

Procedure CocktailShakerSort
(A : list of sortable items ) defined as:
while swapped==true
    swapped := false
    for each i in 0 to end -1 do:
        if A[i] > A[i + 1] then
            swap( A[i], A[i + 1] )
            swapped := true
        end if
    end for
    if not swapped then
        break while loop
    end if
    swapped := false
    for each i in end - 1 to start do:
        if A[i] > A[i + 1] then
            swap( A[i], A[i + 1] )
            swapped := true
        end if
    end for
end while
end procedure

```

WORKING

