



SNS COLLEGE OF TECHNOLOGY

Coimbatore-35

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

COURSE NAME : 23ITT201 DATA STRUCTURES

II YEAR/ III SEMESTER

UNIT - V SORTING AND HASHING

Topic: *INSERTION SORT*

Brute Force Design Technique

- General problem solving technique
- Straight forward approach
- Every possibilities
- Test and error
- Example : 4 digit pattern lock

Try for all the possibilities – 0001,0002,0003,..... – in worst case 10^4

Bubble Sort

- Compare the adjacent elements of list and swap if they are out of order
- Doing it repeatedly will bubble up largest element to the last position of the list
- *Example, $n=7, n-2=5, n-2-i$*

i	Pass	89	45	68	90	29	34	17
0	0	45	89	68	90	29	34	17
	1	45	68	89	90	29	34	17
	2	45	68	89	90	29	34	17
	3	45	68	89	29	90	34	17
	4	45	68	89	29	34	90	17
	5	45	68	89	29	34	17	90

i	Pass j	45	68	89	29	34	17	90
1	0	45	68	89	29	34	17	90
	1	45	68	89	29	34	17	90
	2	45	68	29	89	34	17	90
	3	45	68	29	34	89	17	90
	4	45	68	29	34	17	89	90

i	Pass j	45	68	29	34	17	89	90
2	0	45	68	29	34	17	89	90
	1	45	29	68	34	17	89	90
	2	45	29	34	68	17	89	90
	3	45	29	34	17	68	89	90

i	Pass j	45	29	34	17	68	89	90
3	0	29	45	34	17	68	89	90
	1	29	34	45	17	68	89	90
	2	29	34	17	45	68	89	90

i	Pass j	29	34	17	45	68	89	90
4	0	29	34	17	45	68	89	90
	1	29	17	34	45	68	89	90

i	Pass j	29	17	34	45	68	89	90
5	0	17	29	34	45	68	89	90

Bubble Sort

Algorithm

```
ALGORITHM BubbleSort(A[0..n-1])
//Sorts a given array by bubble sort
//Input: An array A[0..n-1] of orderable elements
//Output: Array A[0..n-1] sorted in nondecreasing order
for i ← 0 to n-2 do
    for j ← 0 to n-2-i do
        if A[j+1] < A[j] swap A[j] and A[j+1]
```

Analysis

1. Input size ..
2. Basic operation – Key Comparison $A[j] < A[j+1]$
3. Count of basic operation – **summation formulas** - $O(n^2)$

$$\begin{aligned} C(n) &= \sum_{i=0}^{n-2} \sum_{j=0}^{n-2-i} 1 = \sum_{i=0}^{n-2} [(n-2-i) - 0 + 1] \\ &= \sum_{i=0}^{n-2} (n-1-i) = \frac{(n-1)n}{2} \in \Theta(n^2). \end{aligned}$$

1. No . of Swap is $O(n^2)$
2. Efficiency – worst / Best / Average

References

1. M. A. Weiss, “Data Structures and Algorithm Analysis in C”, Pearson Education, 8TH Edition, 2008.
2. A. V. Aho, J. E. Hopcroft and J. D. Ullman, “Data Structures and Algorithms”, Pearson Education, 2nd Edition, 2007
3. Ashok Kamthane, " Data Structures Using C ", Pearson Education, 2nd Edition, 2012.
4. Sahni Horowitz, “Fundamentals of Data Structures in C”Universities Press; Second edition 2008