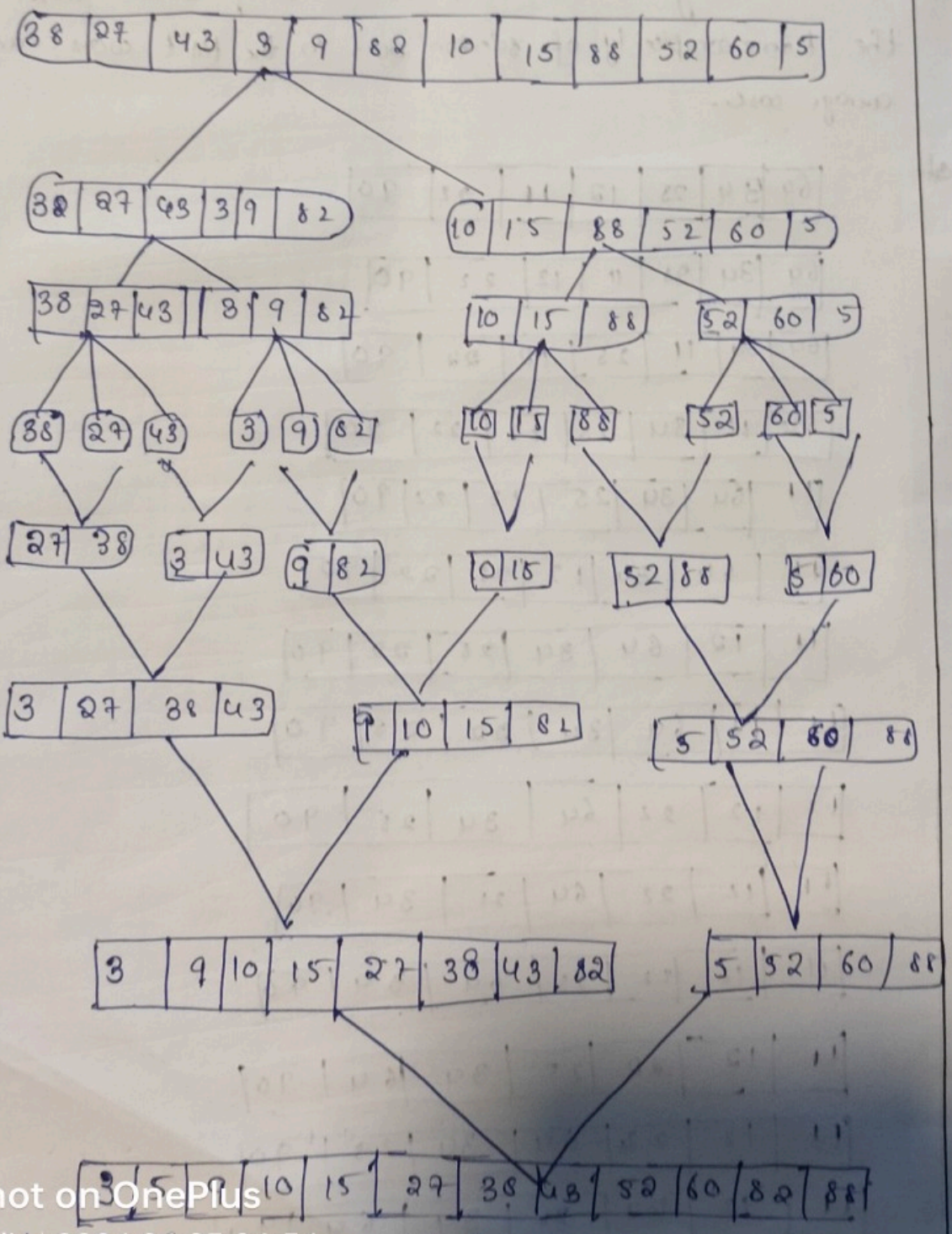
ASSIGNMENT-12

M. polivedday
19A32S056.

(6) Sort the array using merge sort data and tenquer stage by [38, 27, 43, 3, 9, 82, 10, 15, 88, 52, 60, 5] using and analize complexity of the algorithm?

Sol) Given array: merge sort



∴ Sorted list = (3, 5, 9, 10, 15, 27, 38, 43, 52, 60, 82, 88).

The time complexity of merge sort is O(nlogn) where n is the num of elements in the list is split into halves logn times and n.

merging is all the elements at each level takes O(n) time.

(2) Sort the array 64, 34, 25, 12, 22, 11, 90 using bubble sort. What is the time complexity of selection sort in the list worst and average case.

Sol

64	34	25	12	11	22	90

64	34	25	11	12	22	90

64	34	11	25	12	22	90

64	11	34	25	12	22	90

11	64	34	25	12	22	90

11	64	34	12	25	22	90

11	12	64	34	25	22	90

11	12	64	34	22	25	90

11	12	22	64	34	25	90

11	12	22	64	25	34	90

11	12	22	25	64	34	90

11	12	22	25	34	64	90

11	12	22	64	34	25	90

11	12	22	25	34	64	90

③ Sort the array 64, 25, 12, 22, 11 Using selection sort. Whatis the T.C of selection sort in the best, worst and average cases?

64	25	12	22	11

In the selection we will fix that from the longest element in there correct position first - 50

25	64	12	22	11
25	12	64	22	11
25	12	22	64	11
25	12	22	11	64
12	25	22	11	64
12	22	25	11	64
12	22	11	25	64
12	11	22	25	64
11	12	22	25	64

The sorted list is 11, 19, 22, 25, 64

Time complexity:

Selection sort is an another simple comparision sorted algorithm.

Best case: $O(n^2)$

Average case: $O(n^2)$

Worst case: $O(n^2)$.

(4) given an array of [4, -2, 5, 3, 10, -5, 2, 3, -3, 6, 7, -4, 1, 9, -1, 0, -6, -3, 11, -9] integers sort the following elements using analyse time complexity.

Given array :-

4, -2, 5, 3, 10, -5, 2, 3, -3, 6, 7, -4, 1, 9, -1, 0, -6, -3, -9

Insert 4, -2

-2	4

Insert 5

-2	4	5

Insert 10

-2	3	4	5	10

Insert -5

-5	-2	3	4	5	10

Insert 2

-5	-2	2	3	4	5	10

Insert 8

-5	-2	2	3	4	5	8	10

Insert -3

-5	-3	-2	2	3	4	5	8	10

Insert 6

-5	-3	-2	2	3	4	5	6	8	10

Insert 7

-5	-3	-2	2	3	4	5	6	7	8	10

Insert -4

-5	-4	-3	-2	2	3	4	5	6	7	8	10

Insert 9

-5	-4	-3	-2	1	2	3	4	5	6	7	8	9	10

Insert -1

-5	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10

Insert -6

-6	-5	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10

Insert -8

-8	-6	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10

Insert -9

-9	-6	-4	-3	-2	-1	1	2	3	4	5	6	7	8	9	10

Time complexity:-

Best case:- O(n) This occurs when the array is already sorted the inner loop runs zero times for every element

Average case:-

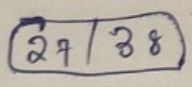
This happens because on average the algorithm will have to move half of the element for each insertion.

Worst case:- $O(n^2)$

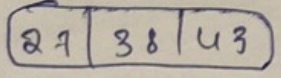
This occurs when the array is sorted in reverse order each insertion takes O(n) times.

(5) sort the following elements using insertion sort using brute fore approch strategy [38, 27, 03, 3, 9, 82, 10, 15, 84, 52, 60, 5] and analyse complexity of the algorithm.

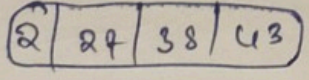
Insert 38, 27:-



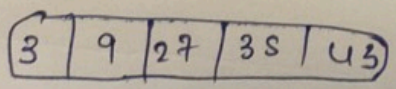
Insert 43:-



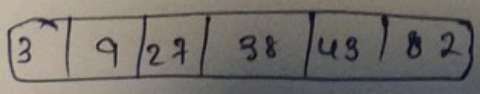
Insert 3:-



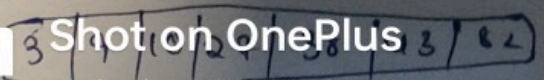
Insert 9:-



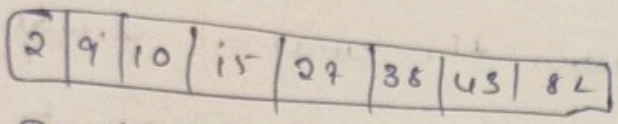
Insert 82:-



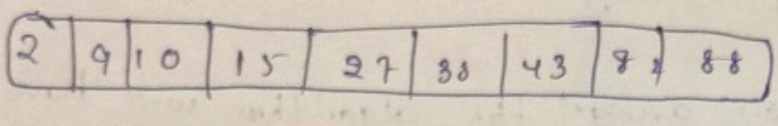
Insert 10:-



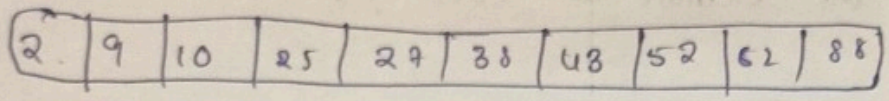
Insert 15:-



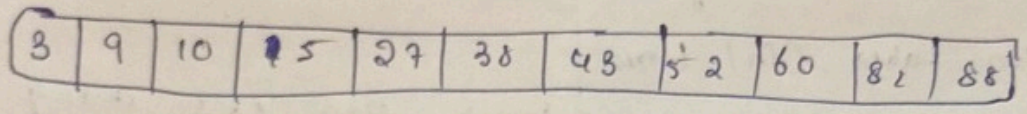
Insert 88:-



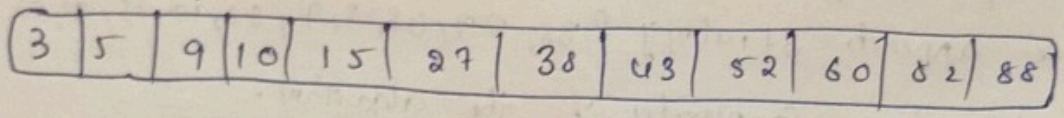
Insert 52:-



Insert 60:-



Insert 5:-



Time complexity:-

Best case:- $O(n)$

Average case:- $O(n \log n)$

Worst case:- $O(n^2)$.