Assignment - 5

NAME : Gayam Rahul.

REG NO : 192311244

DEPT : CSE

COURSE CODE : CSA0389

COURSE NAME : DATA STRUCTURE

FACULTY NAME : Dr. Ashok Kumar

ASSIGNMENT NO : 05

DATE OF SUBMISSION :

NO. OF PAGES : 4.

~~Assignment-5~~ NAME:-G.Rahul

DATA STRUCTURE. Reg no:-192311244

i> Write the algarithm for insertion sort and sort the following Sequence 3,1,4,1,5,9,2,6,5

ii> Explain the procedure for merge sort and perform merge sort for following inputs. Also show the result for each step of iteration.

64, 8, 216, 512, 27, 729, 0, 1, 343, 125.

(i) Algorithm

```
Algorithm intersection sort (a,n);
  for (i=1; i<n-1; i++)
        Key = a[i]
          j=i-1
        While j>0 and a[i] > key;
              a[j+1] = a[j]
                j=j-1
              a[j+1] = Kay.
```

Initialize temp variable

temp □

3	1	4	1	5	9	2	6	5

Step-1:

a[0] = 3, a[1] = 1

a[0] > a[1]

1 goes to temp

1

3		4	1	5	9	2	6	5

1 is 3, a[0] = 1.

□

1	3	4	1	5	9	2	6	5

Step-2:-

a[1]=3 o[2] = 4

3<4 nochange.

Step-3:-

a[2]=4 o[3] = 1

1<4, goes to temp.

1

1	3	4		5	9	2	6	5

1=1, 1<3, a[0] = 1

□

1	1	3	4	5	9	2	6	5

Step-4:

a[3] = 4 a[4] = 5

a[3] < a[4]

No Change.

Step-5:

a[4] = 5 a[5] = 9

a[4] < a[5]

no Change.

Step-6:

a[5] = 9 a[6] = 2

a[5] > a[6] 2 goes to temp.

[2] | 1 | 1 | 3 | 4 | 5 | | 6 | 5 |

2 > 1, 2 > 1, 2 < 3.

[ ] | 1 | 1 | 2 | 3 | 4 | 5 | 9 | 6 | 5 |

Step-7:

a[6] = 9 a[7] = 6.

a > 6, 6 goes to temp

[6] | 1 | 1 | 2 | 3 | 4 | 5 | 9 | | 5 |

6 < 9

[ ] | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 9 | 5 |

Step-8:

a[7] = 9 a[8] = 5

9 > 5, 5 goes to temp

[5] | 1 | 1 | 2 | 3 | 4 | 5 | 6 | 9 |

5 < 6

[ ] | 1 | 1 | 2 | 3 | 4 | 5 | 5 | 6 | 9 |

Sorted Array

a[ ] = {1, 1, 2, 3, 4, 5, 5, 6, 9}

(ii) Merge Sort:

Initial array: [64, 8, 216, 512, 27, 729, 0, 1, 343, 125]

| 64 | 8 | 216 | 512 | 27 | 729 | 0 | 1 | 343 | 125 |

Left Right.

| 64 | 8 | 216 | 512 | 27 | | 729 | 0 | 1 | 343 | 125 |

Left Right Left Right

| 64 | 8 | | 216 | 512 | 27 | | 729 | 0 | | 1 | 343 | 125 |

| 64 | | 8 | | 216 | | 512 | | 27 | | 729 | | 0 | | 1 | | 343 | | 125 |

| 8 | | 64 | | 27 | | 216 | | 512 | | 0 | | 729 | | 1 | | 125 | | 343 |

| 8 | 64 | ~~64~~ | 27 | 216 | 512 | | 0 | 729 | | 1 | 125 | 343 |

| 8 | 27 | 64 | 216 | 512 | | 0 | 1 | 125 | 343 | 729 |

| 0 | 1 | 8 | 27 | 64 | 125 | 216 | 343 | 512 | 729 |

Draw the Concept map of partioning in quick sort, try to write an algorithm for it, which is as follow & develop a program considering these steps.

Step-1:- Choose the highest inden value pivot.

Step-2:- Take two variable to point left and Right of the list Excluding ~~point~~.

Step-3:- Left points to the Low inden Using elements you own.

a[ ] = {27, 10, 36, 18, 25, 45}

left ↓

27	10	36	18	25	45

Pivot ↑Right

Compare a[Pivot] & a[right]

a[Pivot] < a[right], So right moves forward one position.

↓ Left

27	10	36	18	25	45

Pivot ↑right

a[Left] = a[Pivot] = 27, a[right] = 25

a[Pivot] > ~~a~~[right], So Swap

↓ pivot

25	10	36	18	25	45

Left ↓ ↑ right.

a[pivot] = a[right] = 27 a[left] = 25.

* Since, Pivot is at right, so algorithm starts from left and moves to right.

a[Pivot] > a[left] So also moves one position to right.

↓ left ↓ Pivot

25	10	36	18	27	45

↑ right.

a[left] = 10, a[Pivot] = a[right] = 27

a[left] < a[Pivot], So left moves forward

↓ left ↓ Pivot

25	10	36	18	27	45

↑ right

a[left] = 36, a[Pivot] = a[right] = 27, a[Pivot] < a[left].

So Swap.

25	10	27	18	36	45

↑ Pivot ↑ right.

a[left] = a[Pivot] = 27, a[right] = 36.

* Since, pivot in at left, So algorithm Starts from right and move to left.

a[Pivot] < a[right] right moves one position forward

25	10	27	18	36	45

a[left] = 18 a[pivot] = a[right] = 27.

a[pivot] > a[left], So left moves one position forward.

* Elements that are right side of element 27 greater than it and the elements that are left side of 24 are smaller.

25	10	18	27	36	45

left Subarray right Subarray.

* Now in a Similar manner, quick Sort algorithm is seperately applied to the left and right Subarray.

10	18	25	27	36	45