SIMATS

ASSIGNMENT NO. 5

COURSE CODE - CSA0389.

COURSE NAME - DATA STRUCTURE.

NAME - K. GOWTHAM REDDY.

REG NO - 192311220.

i) write the algorithm 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 the following inputs. Also, show the result for each step of iteration. 64, 8, 216, 512, 27, 729, 0, 1, 343, 125.

i) **Algorithm**

```
insertion 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+i] = a[j]
       j = j-1
      a[j+1] = key
```

Sorting:

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 < 3, a[0] = 1

☐

1	3	4	1	5	9	2	6	5

step-2 :

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

3 < 4 No change

step 3 :

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

1 < 4, 1 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 | 9 | | 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

9 > 6, 6 goes to temp

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

6 > 1, 6 > 1, 6 > 2, 6 > 3, 6 > 4,

6 > 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 | |

1 < 5, 1 < 5, 2 < 5, 3 < 5, 4 < 5,

5 < 6

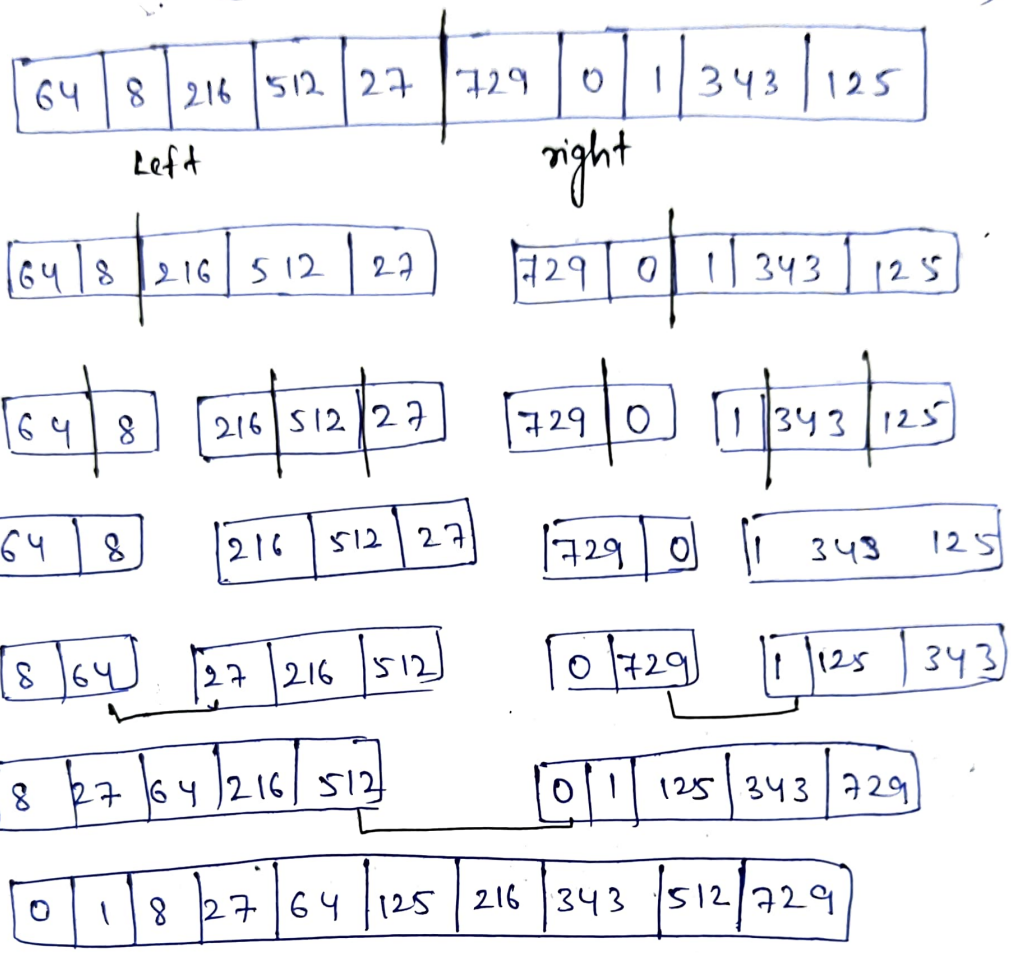
| 1 | 1 | 2 | 3 | 4 | 5 | 5 | 6 | 9 |

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]



2) 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.

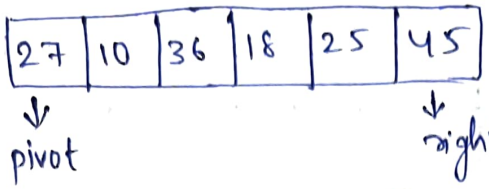
step-1: choose the highest index value has pivot

step-2: Take two variables to point left and right of the list excluding pivot.

step3: left points to the low index.

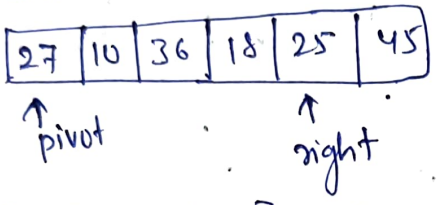
using elements your own.

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



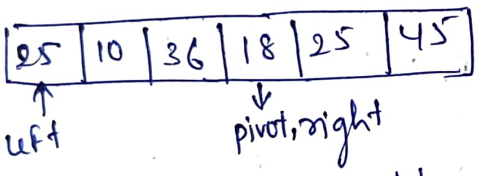
compare a[pivot] & a[right]

a[pivot] < a[right], so right moves forward one position.



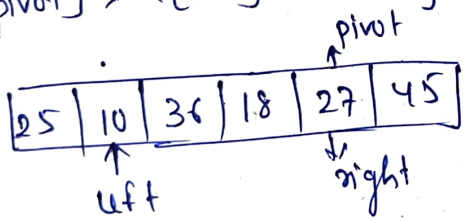
a[left] = a[pivot] = 27, a[right] = 25

a[pivot] > a[right], so sway



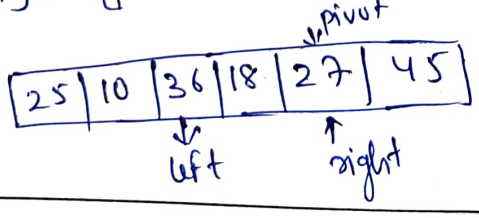
→ since, pivot is at right, so algorithm starts from left and moves to right.

a[pivot] > a[left], so algo moves one position to right.

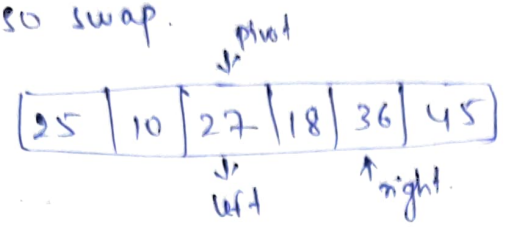


a[left] = 10, a[pivot] = a[right] = 27.

a[left] < a[pivot], so left moves forward



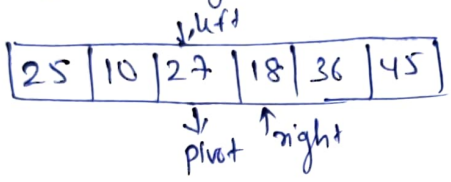
a[left] = 36, a[pivot] = a[right] = 27, a[pivot] < a[left], so swap.



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

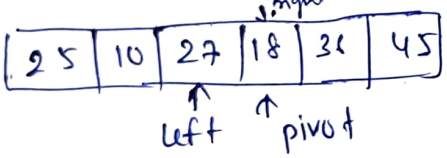
→ since pivot is at left, so algorithm starts from right and move to left.

a[pivot] < a[right], right moves one position forward.



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

a[pivot] > a[right] so swap.



→ since pivot is at right, so algorithm starts from left and moves to right

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

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

→ Now, a[pivot], a[left] and a[right] are same, so they are pointing the same element, it represents the termination of procedure.