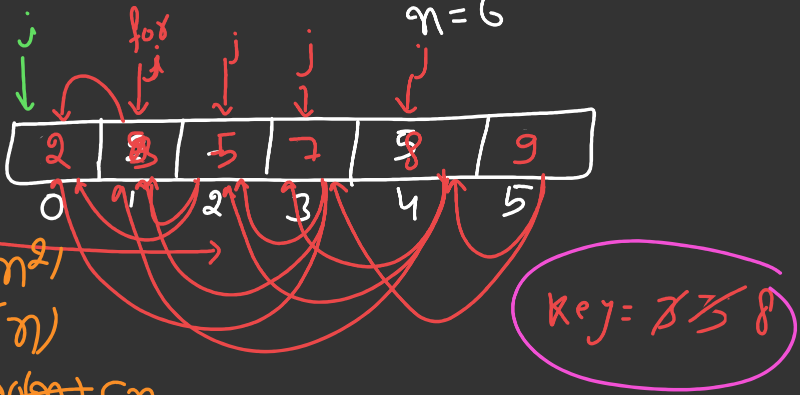
Date
17/01/23

{ Data Structures {

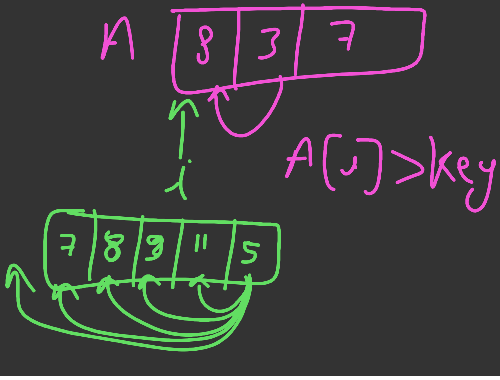
Sortings ::

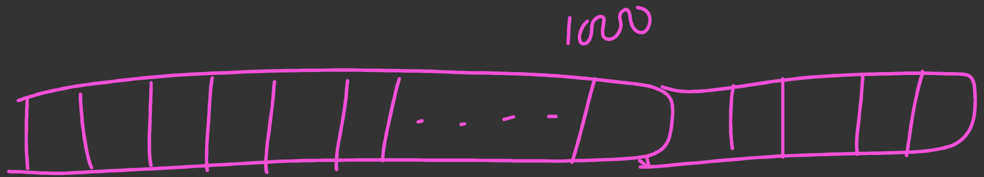
Insertion Sort ::



Worst Case = $O(n^2)$
Best Case = $O(n)$

```
$O(n^2)$  Insertion (A, n)
          {
            for ( j = 1 ;  j ≤ n-1 ;  j++ )  ⟶ ~~O(n)~~ Cn
            {     i = j-1 ;  key = A[j] ;
                  while( A[i] > key && i ≥ 0 )  ⟶ Cn
                  {
                        A[i+1] = A[i] ;
                        i = i-1 ;
                  } A[i+1] = key ;
            }
          }
```





j

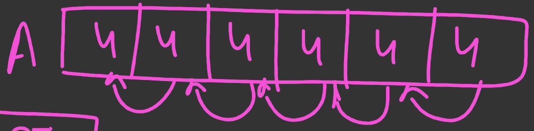
A	8	9	20	30	40	45

· Comparison based Sorting.

9	8	20	30	40	45	50

✓

Best Case $O(n)$



Stable Sortings. ✓

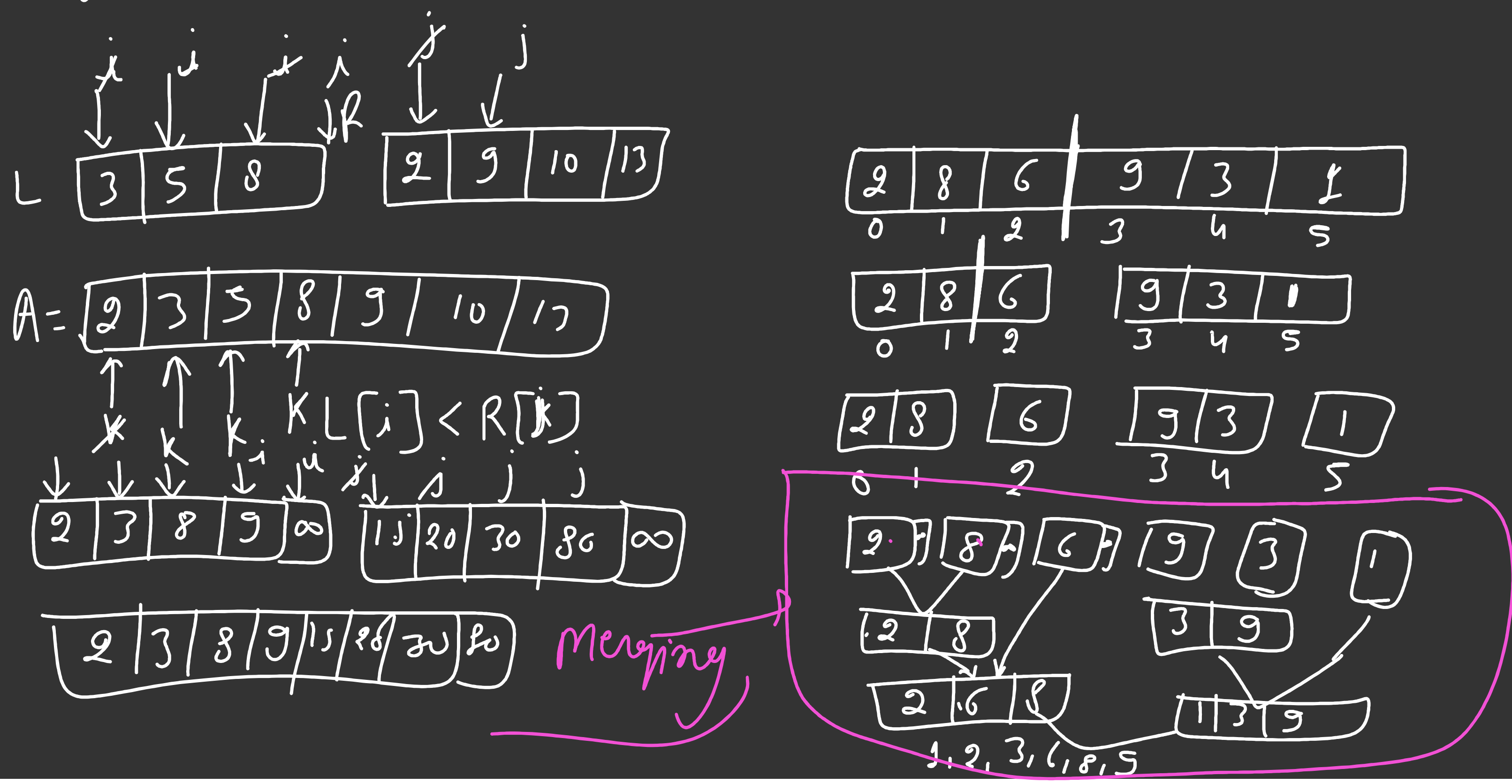
2^*	3	2^+	5	2^-

2^*	2^+	2^-	3	5

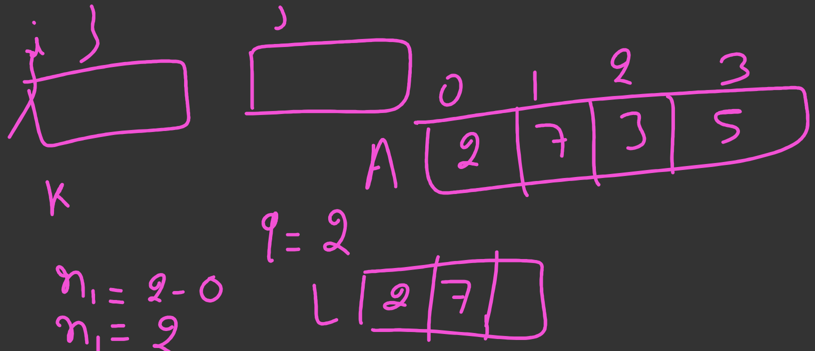
Implace Sorting ✓

Adaptive

Merge Sort ::


i i i i j j
L 3 5 8
R 2 9 10 13
A = 2 3 5 8 9 10 13
k k k k
L[i] < R[k]
2 3 8 9 ∞
i i i j j j
1.5 20 30 80 ∞
2 3 8 9 15 20 30 80
Merging
2 8 6 9 3 1
0 1 2 3 4 5
2 8 6 9 3 1
0 1 2 3 4 5
2 8 6 9 3 1
0 1 2 3 4 5
2 8 6 9 3 1
2 8 3 9
2 6 8 1 3 9
1, 2, 3, 6, 8, 9

```
merge(A, P, q, r)
{   $n_1 = q - P$
    $n_2 = r - q + 1$
  L[1.....$n_1$+1]  R[1...$n_2$+1]
  for(i←0 to $\overset{2}{n_1}$)
    {  -L[i] = A[P+i]
    }
  for(j←0 to $n_2$)
       R[j] = A[q+j]
  L[$n_1$+1] = ∞    R[$n_2$+1] = ∞      i=0; j=0;
  for(k←P to r)
  {
      if(L[i] ≤ R[j])
      {
         A[K]=L[i]
         i++;
      }
  Else{   A[K]=R[j]
             j++;
  }
```

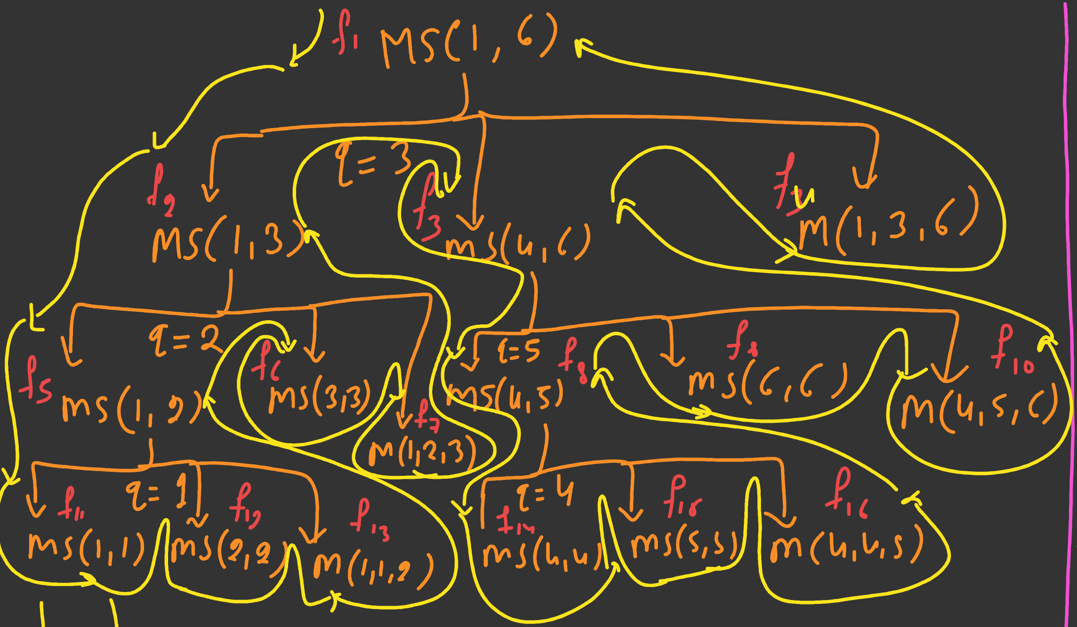


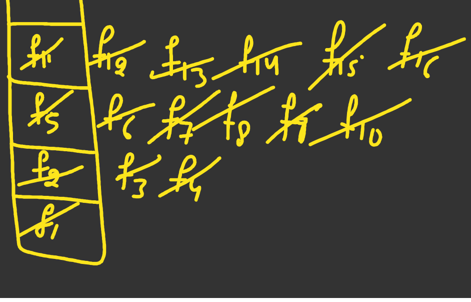
0	1	2	3
2	7	3	5

q = 2

$n_1 = 2 - 0$
$n_1 = 2$

2	7	




~~f11~~ ~~f12~~ ~~f13~~ ~~f14~~ ~~f15~~ ~~f16~~
~~f5~~ ~~f6~~ ~~f7~~ ~~f8~~ ~~f9~~ ~~f10~~
~~f2~~ ~~f3~~ ~~f4~~
~~f1~~

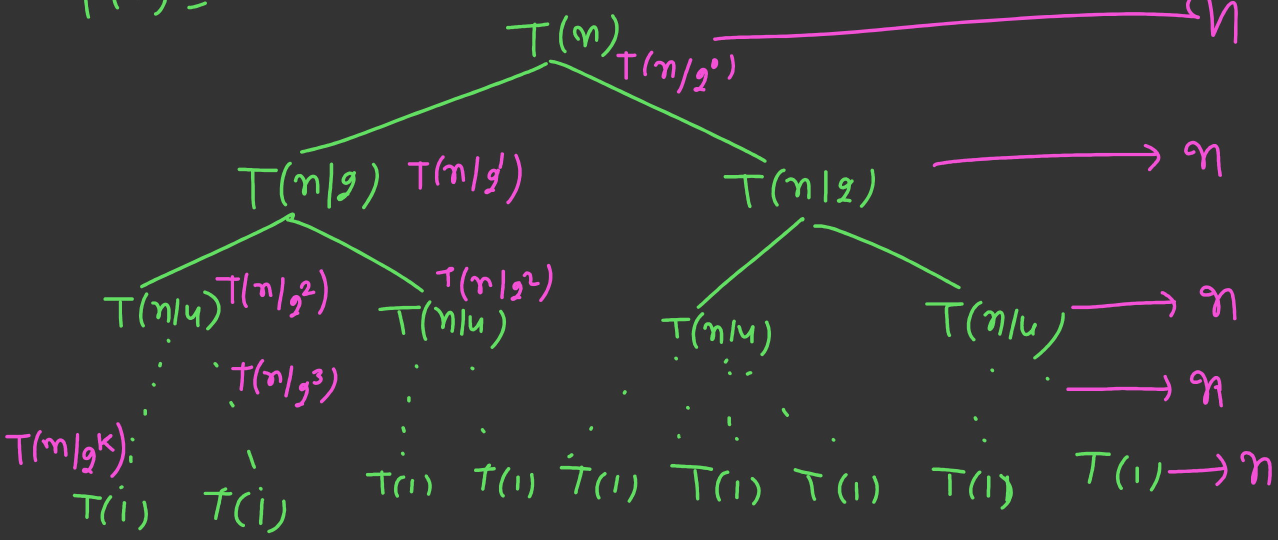
Space = O(n)
= O(
Space = n+lgn = O(n)

ms:

```
Sort
Merge(A, P, r) T(n)
{
    if (p<r) ⟶ O(1)
    {
        q ← ⌊(p+r)/2⌋ ⟶ O(1)
                      ms
        T(n/2) ← Mergesort(A, P, q)
                 ms
        T(n/2) ← Mergesort(A, q+1, r)
                M
        n ← Merge(A, P, q, r)
    }
}      T(n) = 2T(n/2) + n
```

DFS:

$T(n) = 2T(n/2) + n$



$\frac{n}{2^K} = 1 \Rightarrow n = 2^K \Rightarrow K = \lg n$

height of the tree = $K+1 = \lg n + 1$

$\Rightarrow n(\lg n + 1) \Rightarrow n\lg n + n$

$O(n\lg n)$.