Day-9 Sorting Algorithm

Quick Sort:-

51	95	66	72	42	38	39	41	15
0	1	2	3	4	5	6	7	8

Step-1) Decide left & right

left = 0 , right = 8

Step-2) Select pivot element which is the First element of array

pivot = arr[left] = 51

Step-3) Find out the exact pos^n of pivot by using no. of less than that pivot

count ⇒ no. of ele that is less than pivot

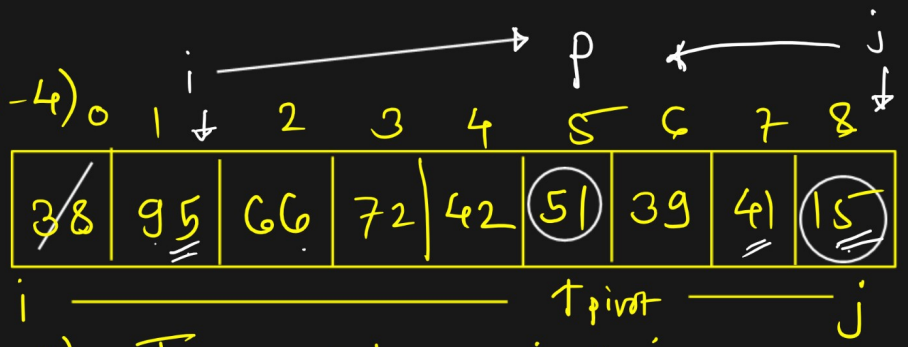
Count = 5

swap (arr[count], arr[left])
pivot

```
                                  0      8
int partition (int left, int right)
{
  int pivot = arr[left];
   int count = 0;
  for(int i = left+1; i<=right; i++)
  {
    if (arr[i] < pivot)
     {
        count++;  ⇒
     }
  }
   int p = left + count;
  swap ( arr[left] , arr[count]);

  int i = left , j = right;
```

Step-4)



Step-5) Two pointers i & j

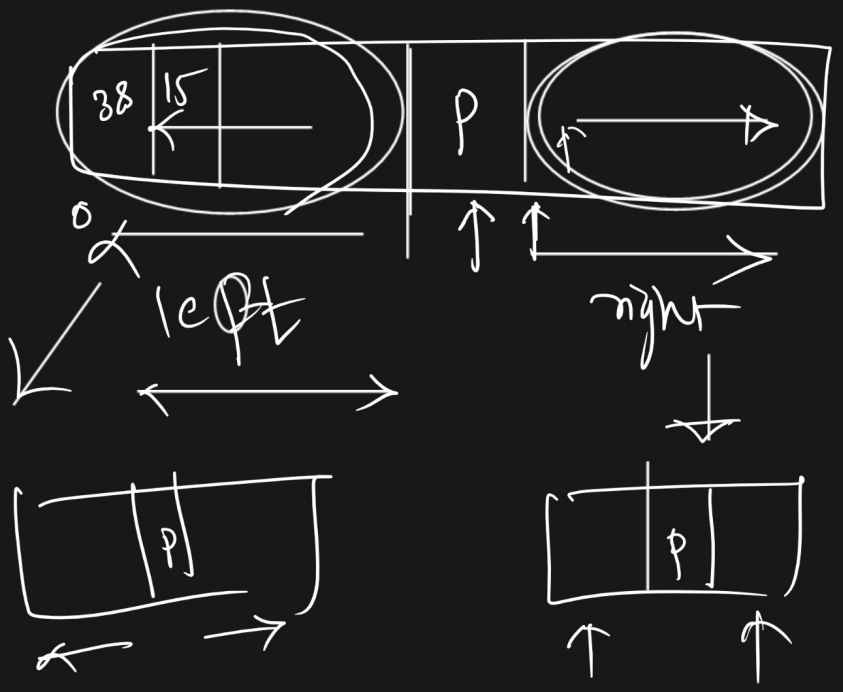
Compare arr[i] < pivot & arr[j] > pivot

i++

swap (arr[i], arr[j])

```
while ( i<p && j>p)
{
    while (arr[i] < pivot)
        i++;
    while (arr[j] > pivot)
        j--;
    swap (arr[i], arr[j])
}
return p;
}
```

QuickSort QuickSort



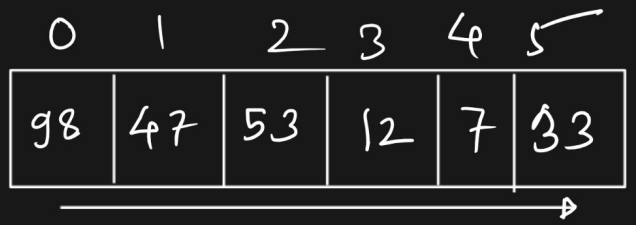

```
void quickSort (int left, int right)
{
    if (left < right)
    {
        int p = partition (left, right);
        quickSort ( left, p-1);
        quickSort (p+1, right);
    }
```

Quick Sort = $O(n \log n)$]

Worst = $O(n^2)$

* Search Algorithm :-

1) Linear Search :-

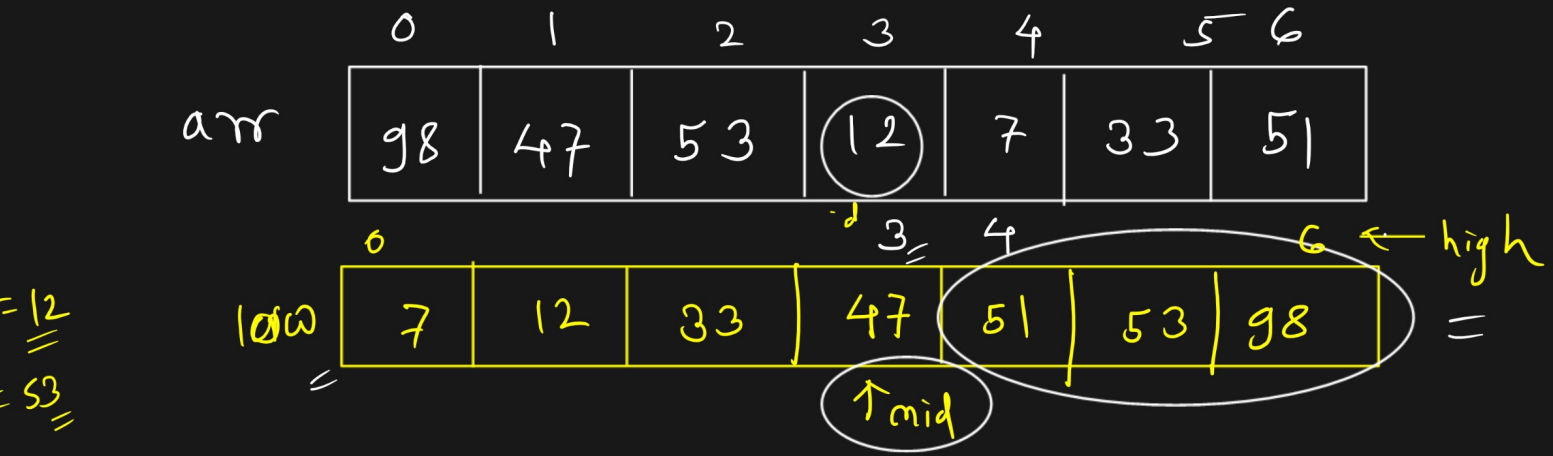


key = 72

Time Complexity = $O(n)$

2) Binary Search :-

Divide & Conquer



step-1) Find out mid index & their value

mid = low + (high-low)/2 ;

arr[mid] = arr[3] = 47

step-2) Compare Search ele with mid value

if (arr[mid] == key)
return mid;

step-3) Compare search ele > arr[mid]

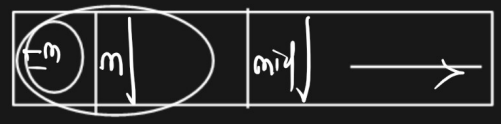
12 > 47

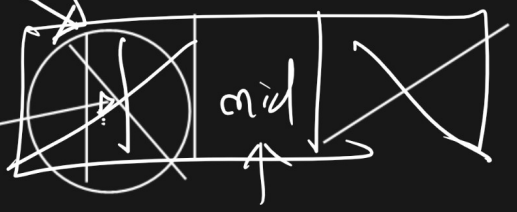
search(mid+1, right)

step-4) Compare search ele < arr[mid]

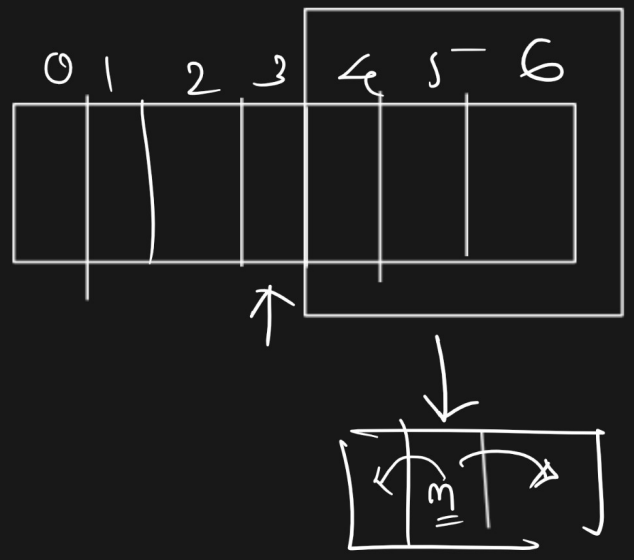
search (left, mid-1)

```
int key,
Void Bsearch ( int low, int high, int arr[])
{
   while ( low <= high)
   {
      int mid = low + (high-low)/2;
      if (arr[mid] == key)
         return mid;
```





```
        if ( arr[mid] > key)
                high = mid-1;
        if (arr[mid] < key)
    }           low = mid+1;
}   return -1;
```



H.W. ⟹ 2643 Leetcode Pbm No