Given an integer array of only 0 and 1, segregate them.
all 0 should be on left side and all 1 should be on right side

1. sort the arr
arr.sort((a,b) => a-b);//increasing
arr.sort((a,b) => b-a);//decreasing
T.C : O(nLogn)

2. countOf0++ = 6
   countOf1++ = 7

   O(n)
   for(0-6)
   arr.push(0)
   for(7-14)
   arr.push(1)

```
let arr = [0,1,1,1,0,1,0,1,0,0,0,1];
let ans = [0,0,0,0,0,0,1,0,0,0,1,1,1,1,0,1,1,1,1,1,1];
let arr = [1,1,1,1,0,0,1,0]

const segregate = (arr) => {
   let n = arr.length;
   let left = 0;
   let right = n-1;

   while(left<right){

      while(arr[left] == 0 && left< right)
       {
          left++;

       }

      while(arr[right] == 1 && left< right)
       {   right--;}

      if(left< right){
         arr[left] = 0;
         arr[right] =1;
         left++;
         right--;
      }
```

```
    }

    console.log(arr);
}

segregate(arr);

TC : O(n)
SC : O(1)


given 0,1,2 , segregate them
Dutch National Flag problem

let arr = [0,1,1,2,1,0,2,1,0];



//1. arr.sort(a,b=>a-b);
//2. freq of 0,1,2  -> TC: O(n)+o(n)
//3.


const func = (arr) => {
    let n = arr.length;
    let left = 0;
    let right = n-1;
    let mid =0;
    let temp;

    while(mid<= right){

        if(arr[mid] == 0)
        {
            temp = arr[mid];
            arr[mid] = arr[left];
            arr[left] = temp;

            left++;
            mid++;
        }
        else if(arr[mid] == 2)
        {
            temp = arr[mid];
```

```
            arr[mid] = arr[right];
            arr[right] = temp;
            right--;
        }

        else
         mid++;
    }

    console.log(arr);
}
//TC : O(n)
//SC : O(1)
func(arr);
```