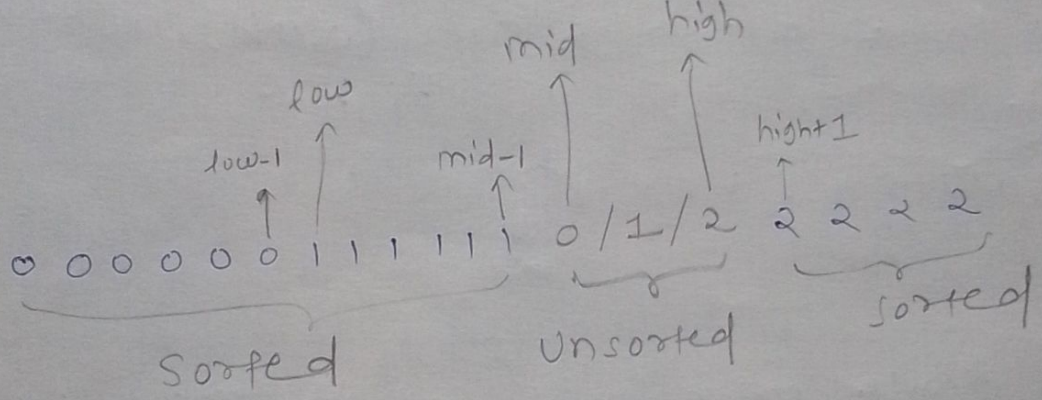
Dutch National flag Algorithm

Used to segergate three diffelent objects or numbers.

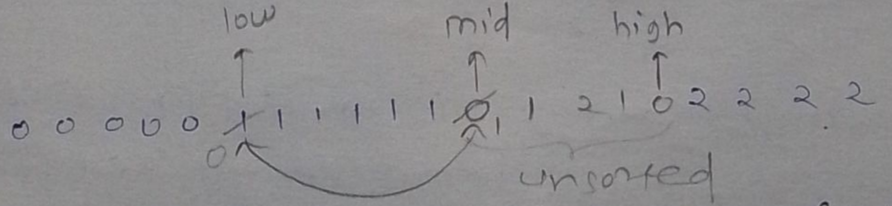
Assumption

0 to low-1 ⇒ 0
low to mid-1 ⇒ 1

high+1 to n-1 ⇒ 2

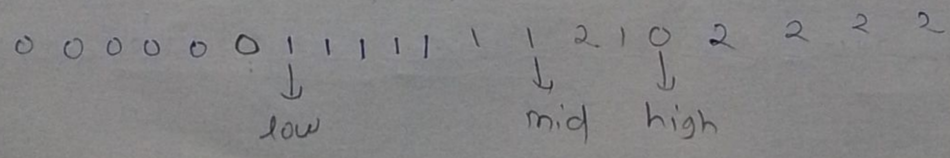


Everything betn mid to high is unsorted at first everything is unsorted in given array so initially we'll keep mid at 0th index and high at n-1th index



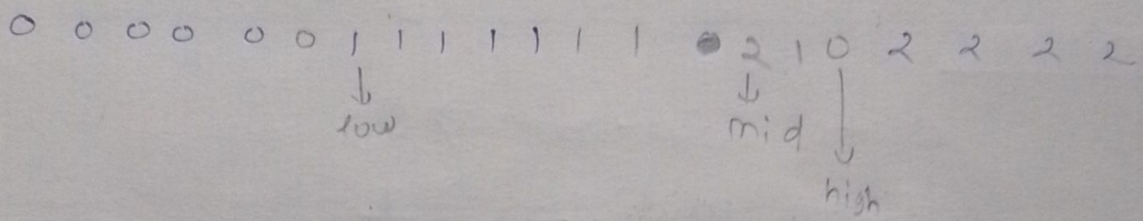
if (a[mid] == 0) swap(a[mid], a[low])

since 0 to l-1 ⇒ 0 and l to m-1 ⇒ 1 so low++, mid++



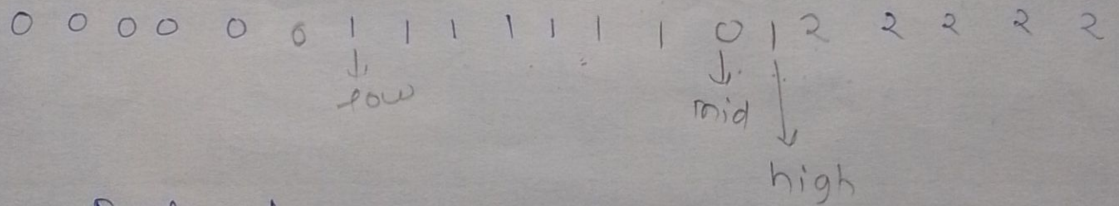
if (a[mid]==1)

Since 1 is supposed to be between
low to mid-1 simply mid++
l to m-1 ⇒ 1

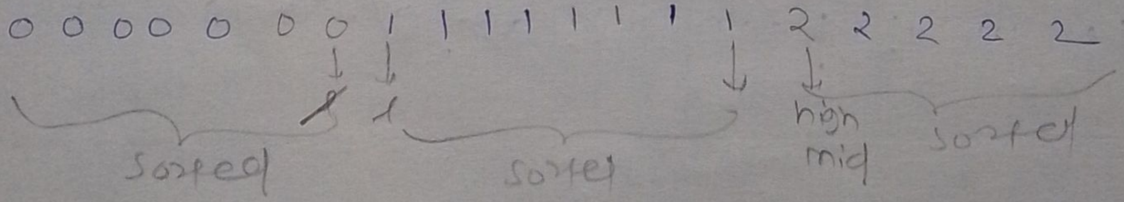


if (a[mid]==2)

at high there could be 0/1/2, No matter what is there at high swap(a[mid], a[high])
high+1 to n-1 ⇒ 2 so high--



But don't move mid, as it may not sorted index
Now by first if condition



when mid == high array is sorted so stop

→ ← on each iteration at least one
mid high el gets to its correct position

when they meet we stop as array sort

Time complexity - O(N) Space Complexity - O(1)