ARRAYS LEETCODE 33
Search in Rotated Sorted Array
/
Input: nums=[4, 5, 6, 7, 0, 1, 2]
target: 0
1) · Sortol -> We can use Binary Search
2) But it is rotated, so one direction will be sorted
3) Find out which part of mid element is sorted
int low = 0, int high = munus.length $-1 = 7-1=6$
0 1 2 3 4 5 5
0 1 23 4 5 6 A 5 67 0 1 2
loco=0 1mid high=6
mid = 0 + (6-0)/2 = 3
Car1: torget == muns[mid] -> return mid;
Chech cohich part is sorted
 15 (7 0 1 2
4,5,6,7,0,1,2
When the second is a second to the second is a second to
nums[mid] < nums[low] - stight post is sorted
 Care 2: Left Part Sooted
muns [mid] > moss[bo] - left port is soxted
Satisfies : (() +
santies if (torget > must las) & torget < must mill
sugh = mid-1;
eleminate Pugus man 1,
 eleminate high=mid-1; one part 3 else 5
one f else s low = mid+1;
classmate

DATE
Care 3: Right Part is soxted
if (torget ≥ nemotinid 7 k& torget ≤ nemo[high]) {
loco = mid + 1;
elxs
high = mid - 1;
Cool 4: If nothing is satisfied for (low < high) condition above, return -1;
condition above, return -1;