30 Days Leetcode challenge - Day 15

public class leetday15

{

class Solution1 {

public int search(int[] arr, int tar)

{

int low = 0;

int high = arr.length-1;

int ans = -1;

while(low<=high)

{

int mid = (low+high)/2;

if(arr[mid]==tar)

{

ans = mid;

}

// check left half is sorted

if(arr[low]<=arr[mid])

{

if(arr[low]<= tar && tar <= arr[mid])

{

high = mid - 1;

}

else

{

low = mid+1;

}

}

else

{

if(arr[mid]<=tar && tar <= arr[high])

{

low = mid +1;

}

else

{

high = mid -1;

}

}

}

return ans;

}

}

class Solution2 {

public boolean search(int[] arr, int tar)

{

int low = 0;

int high = arr.length-1;

int ans = -1;

while(low<=high)

{

int mid = (low+high)/2;

if(arr[mid]==tar)

{

return true;

}

if(arr[low] == arr[mid] && arr[mid] == arr[high])

{

low = low + 1;

high = high -1;

continue;

}

// check left half is sorted

if(arr[low]<=arr[mid])

{

if(arr[low]<= tar && tar <= arr[mid])

{

high = mid - 1;

}

else

{

low = mid+1;

}

}

else

{

if(arr[mid]<=tar && tar <= arr[high])

{

low = mid +1;

}

else

{

high = mid -1;

}

}

}

return false;

}

}

}