NAME: MOAZZAM FAROOQUI

ROLL NO: CT-24068

COURSE CODE: CT-159

ASSIGNMENT: DSA PRACTICE SESSION TASK#1

Q1. Find first and last position of elements in a sorted array.

SOURCE CODE:

```
class Solution {
   public:
        int firstOccurrence(vector<int>% nums,int target){
        int low=0;
        int high=nums.size()-1;
        int first=-1;

        while (low<=high){
            int mid=(low+high)/2;

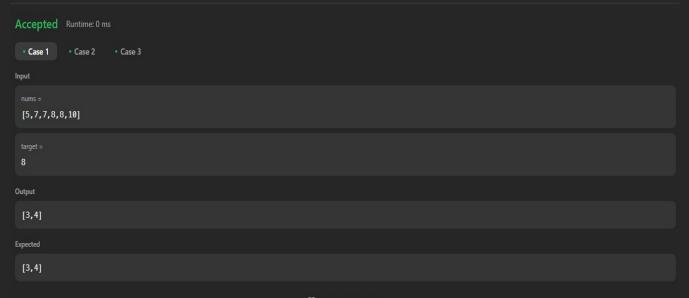
            if (nums[mid]==target){
                first=mid;
                high=mid-1;
            }
            else if(nums[mid]<target){
                low=mid+1;
            }
            else{
                high=mid-1;
            }
            else{
                high=mid-1;
            }
            return first;
        }
        return first;
}</pre>
```

```
int lastOccurrence(vector<int>& nums, int target){
    int low=0;
    int high=nums.size()-1;
    int int last=-1;

    while (low <= high){
        int mid=(low+high)/2;

        if (nums[mid]==target){
            last=mid;
            low=mid+1;
        }
        else if(nums[mid]<target){
            low=mid+1;
        }
        else {
            high=mid-1;
        }
    }
    return last;
}</pre>
```

ACCEPTANCE STATUS:



Contribute a testcase

Q2. Search a 2D Matrix

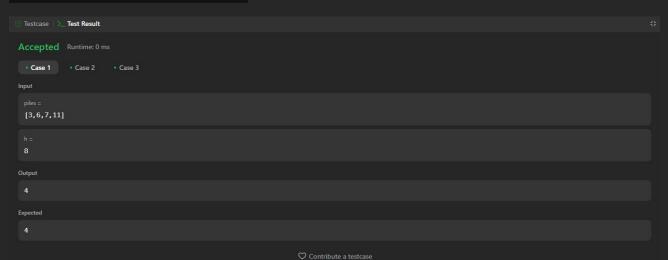
SOURCE CODE:

```
</>Code
C++ ∨ A Auto
  1 class Solution {
  2 ∨public:
int m=matrix.size();
            int n=matrix[0].size();
            int low=0;
            int high=m*n-1;
  8 V
           while(low<=high){
               int mid=(low+high)/2;
               int row=mid/n;
               int col=mid%n;
               int value=matrix[row][col];
               if(value==target){
  13 V
  16 🗸
               else if(value<target){</pre>
                  low=mid+1;
  19 🗸
                  high=mid-1;
```

estcase >_ Test Result	
ccepted Runtime: 0 ms	
Case 1 • Case 2	
natrix = [[1,3,5,7],[10,11,16,20],[23,30,34,60]]	
	O
tput	
rue	
ected	
rue	
♥ Contribute a testcase	

Q3. Koko Eating Bananas

SOURCE CODE:



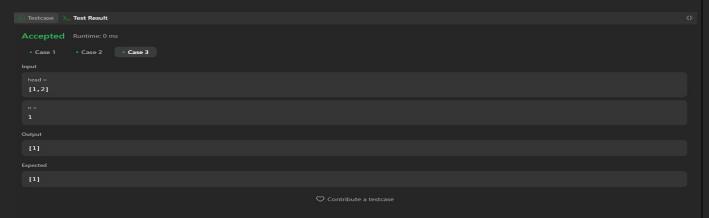
Q4. Merge Two Sorted Lists

SOURCE CODE:

Q5. Remove Nth Node From End of List

SOURCE CODE:

ACCEPTANCE STATUS:



6. Linked List Cycle II

SOURCE CODE:

```
if(slow==fast){
    ListNode*entry=head;
    while(entry!=slow){
        entry=entry->next;
        slow=slow->next;
}
return entry;
}
return nullptr;
}
```

