Find the element that appears once in sorted array

Given a sorted array arr[] of size N. Find the element that appears only once in the array. All other elements appear exactly twice.

Example 1:

```
Input:
N = 11
arr[] = {1, 1, 2, 2, 3, 3, 4, 50, 50, 65, 65}
Output: 4
Explanation: 4 is the only element that
appears exactly once.
```

Your Task:
You don't need to read input or print anything. Complete the function **findOnce()** which takes sorted array and its size as its input parameter and returns the element that appears only once.

Expected Time Complexity: O(log N)
Expected Auxiliary Space: O(1)

Constraints:

1 <= N <= 10^5

-10^5 <= arr[i] <= 10^5

```
def findOnce(self,arr : list, n : int):
        # Complete this function
        if n ==1:
            return arr[0]
        if arr[0]!=arr[1]:
            return arr[0]
        if arr[-1]!=arr[-2]:
            return arr[-1]

        lo = 0
        hi = n-1

        while lo<=hi:
            mid = (lo+hi)//2
            if arr[mid]!=arr[mid-1] and arr[mid]!=arr[mid+1]:
```

```
            return arr[mid]
        elif arr[mid]==arr[mid-1]:
            lc = mid-lo+1
            if lc%2==0:
                lo = mid+1
            else:
                hi = mid-2
        elif arr[mid]==arr[mid+1]:
            rc = hi-mid+1
            if rc%2==0:
                hi = mid-1
            else:
                lo = mid+2
```