Find the element that appears once in a sorted array

```c
#include <stdio.h>
#include <stdlib.h>

int getElementAppearsOnce(int *arr, int start, int end)
{
        if(start < end)
        {
                int middle = (start + end) / 2;
                if(middle % 2 == 0)
                        return (arr[middle] == arr[middle+1]) ? getElementAppearsOnce(arr,
middle+2, end) : getElementAppearsOnce(arr, start, middle);
                else
                        return (arr[middle] == arr[middle - 1])? getElementAppearsOnce(arr, middle
- 2, end): getElementAppearsOnce(arr, start, middle - 1);
        }
        return (start == end)? start : -1;
}

int main()
{
        int *arr, size, ele;
        printf("Enter size of an array\n");
        scanf("%d", &size);
        //allocate memory for array
        arr = (int *)malloc(size * sizeof(int));

        printf("Enter Array elements ");
        for(int index = 0; index < size; index++)
                scanf("%d", &arr[index]);
        ele = getElementAppearsOnce(arr, 0, size - 1);
        if(ele > -1)
                printf("The element appears once in a given sorted array is = %d\n",
                        arr[ele]);
        return 0;
}
```

Time Complexity: O(logn)
Space Complexity: O(logn)