Index of First Occurrence in a Sorted Array

I/P: arr[] = [1, 10, 10, 10, 20, 20, 40]
x = 20

O/P: 4

I/P: arr[] = [10, 20, 30]
x = 15

O/P: -1

I/P: arr[] = [15, 15, 15]
x = 15

O/P: 0

Naive Approach

```
def firstOccurrence(arr, n, x):
    for i in range(0,n):
        if (arr[i]==x):
            return i

    return -1
```

Recursive Binary Search

```
def firstOccurrence(arr, low, high, x):
    if low > high:
        return -1
    mid = (low + high)//2
    if x > arr[mid]:
        return firstOccurrence(arr, mid+1, high, x)
    elif x < arr[mid]:
        return firstOccurrence(arr, low, mid-1, x)

    else:
        if mid == 0 or arr[mid-1] != arr[mid]:
            return mid

        else :
            firstOccurrence(arr, low, mid-1, x)
```

Iterative Binary search

```
def firstOccurrence(arr, n, x):
    low = 0
    high = n-1
    while (low <= high):
        mid = (low + high)//2
        if x > arr[mid]:
            low = mid+1
        elif x < arr[mid]:
            high = mid-1
        else:
            if mid==0 or arr[mid-1] != arr[mid]:
                return mid
            else:
                high = mid - 1
    return -1
```