Strivers-A2Z-DSA-Sheet-main\02.Binary Search\1D Arrays\07.Number_of_occurences.cpp

```cpp
/*
QUESTION:
Given a sorted array Arr of size N and a number X, you need to find the number of occurrences of X in Arr.

Example:

Input:
N = 7, X = 2
Arr[] = {1, 1, 2, 2, 2, 2, 3}
Output: 4
Explanation: 2 occurs 4 times in the given array.
Example 2:

Input:
N = 7, X = 4
Arr[] = {1, 1, 2, 2, 2, 2, 3}
Output: 0
Explanation: 4 is not present in the given array.
*/

/*
APPROACH:
1. Use binary search to find the first occurrence of the target element.
2. Use binary search to find the last occurrence of the target element.
3. Return the difference between the indices of the first and last occurrence + 1.

CODE:
*/

int countOccurrences(int arr[], int n, int x) {
    int first = lower_bound(arr, arr + n, x) - arr;
    if (first == n || arr[first] != x)
        return 0;
    int last = upper_bound(arr, arr + n, x) - arr;
    return last - first;
}

// TIME COMPLEXITY: O(log N), where N is the size of the array.
// SPACE COMPLEXITY: O(1).

```