Prob 6.Given a sorted array arr[] of size N and a number X, you need to find the number of occurrences of X in given array.

Solve:

```
#include <stdio.h>

int findFirstOccurrence(int arr[], int n, int x) {
    int left = 0, right = n - 1;
    int result = -1;
    while (left <= right) {
        int mid = left + (right - left) / 2;

        if (arr[mid] == x) {
            result = mid;
            right = mid - 1;
        } else if (arr[mid] < x) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }
    return result;
}

int findLastOccurrence(int arr[], int n, int x) {
    int left = 0, right = n - 1;
    int result = -1;
    while (left <= right) {
        int mid = left + (right - left) / 2;
```

```
        if (arr[mid] == x) {
            result = mid;
            left = mid + 1;
        } else if (arr[mid] < x) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }
    return result;
}

int countOccurrences(int arr[], int n, int x) {
    int first = findFirstOccurrence(arr, n, x);
    if (first == -1) {
        return 0;    }

    int last = findLastOccurrence(arr, n, x);
    return last - first + 1; }

int main() {
    int arr[] = {1, 2, 2, 2, 2, 3, 4, 5, 5, 6};
    int n = sizeof(arr) / sizeof(arr[0]);
    int x = 2;

    int count = countOccurrences(arr, n, x);
    printf("Number of occurrences of %d: %d\n", x, count);

    return 0;
```

}