

**Prob 10.** Given an array `arr`. Find the majority element in the array. If no majority exists, return `-1`. A majority element in an array is an element that appears strictly more than `arr.size() / 2` times in the array.

**Solve:**

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

```
int findMajorityElement(int arr[], int n) {
```

```
    int candidate = -1;
```

```
    int count = 0;
```

```
    for (int i = 0; i < n; i++) {
```

```
        if (count == 0) {
```

```
            candidate = arr[i];
```

```
            count = 1;
```

```
        } else if (arr[i] == candidate) {
```

```
            count++;
```

```
        } else {
```

```
            count--;
```

```
        }
```

```
    }
```

```
    count = 0;
```

```
    for (int i = 0; i < n; i++) {
```

```
        if (arr[i] == candidate) {
```

```
            count++;
```

```
        }
```

```
    }
```

```
    if (count > n / 2) {
```

```
        return candidate;
```

```
    } else {
```

```
        return -1;
```

```
    }
```

```
}
```

```
int main() {
```

```
    int arr[] = {2, 3, 2, 2, 4, 2, 5, 2};
```

```
    int n = sizeof(arr) / sizeof(arr[0]);
```

```
    int majorityElement = findMajorityElement(arr, n);
```

```
    if (majorityElement != -1) {
```

```
        printf("Majority Element: %d\n", majorityElement);
```

```
    } else {
```

```
        printf("No Majority Element\n");
```

```
    }
```

```
    return 0;
```

```
}
```