HW_Counting Mismatched

```
6
1 1 4 2 1 3
```

given 1 1 4 2 1 3

0 1 2 3 4 5

heights

$[4, 1, 3] \rightarrow 3$

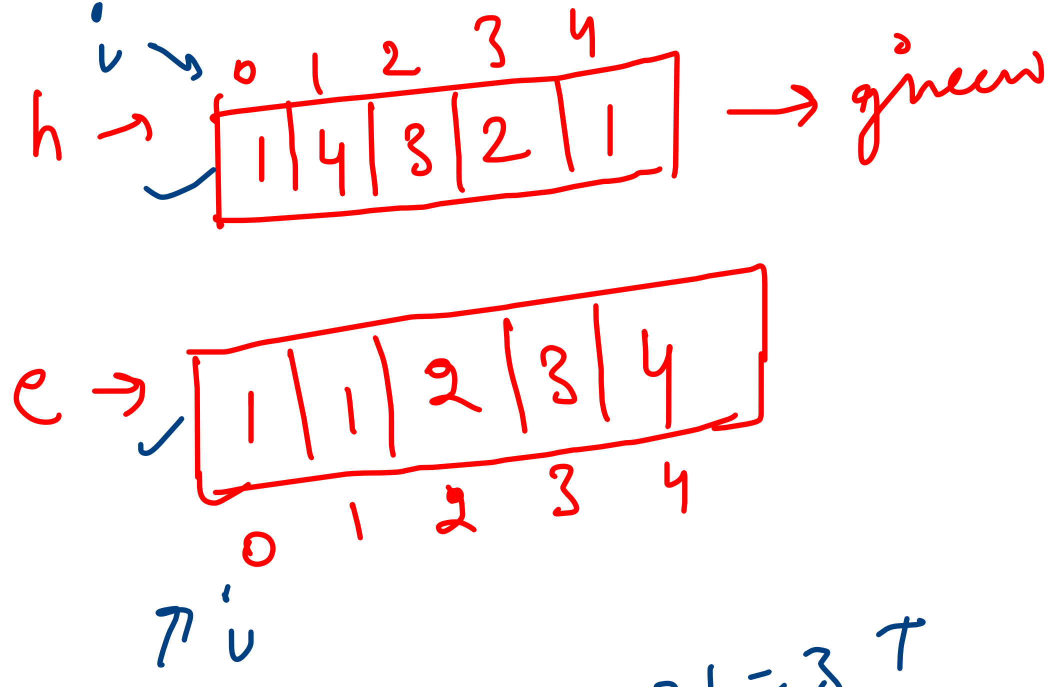
$\rightarrow \underbrace{\text{Roojaa}}$ $\rightarrow 2$

$\rightarrow \underbrace{\text{Roja}}$

exp.

0	1	2	3	4	5
1	1	1	2	3	4

```
public static int mismatch(int height[], int n){
    int expected[] = Arrays.copyOf(height, n);
    Arrays.sort(expected);

    // compare and count
    int count = 0;
    for(int i = 0; i < n; i++){
        if(height[i] != expected[i]){
            count++;
        }
    }
    return count;
}
```



count = ~~0~~ ~~1~~ ~~2~~ ~~3~~ 4

o/p → 4

i=0, 1 != 1 F

i=1, 4 != 1 T

i=2, 3 != 2 T

i=3, 2 != 3 T

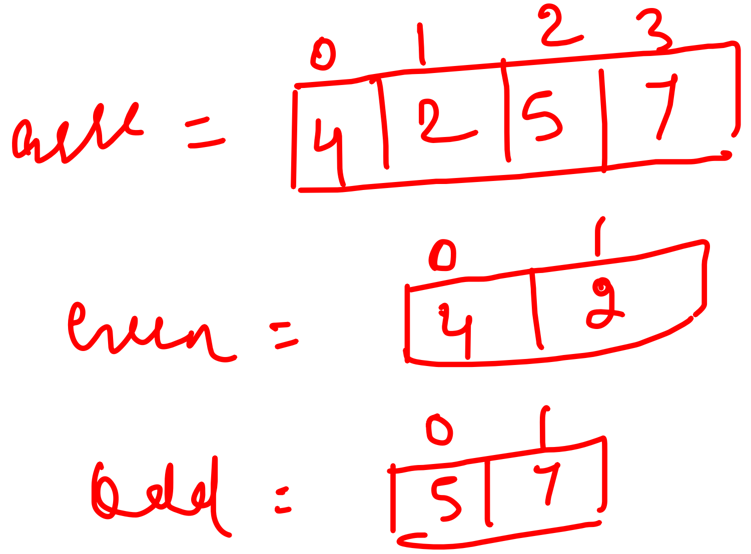
i=4, 1 != 4 T

TC → O(n log n) + n

SC → O(n)

```java
Scanner s = new Scanner(System.in);
int n = s.nextInt()

int arr[] = new int[n];
int even[] = new int[n/2];
int odd[] = new int[n/2];
int evencount = 0;
int oddcount = 0;

for(int i = 0; i < n; i++){
    arr[i] = s.nextInt();
    if(arr[i] % 2 == 0){
        even[evencount++] = arr[i];
    }
    else{
        odd[oddcount++] = arr[i];
    }
}
```

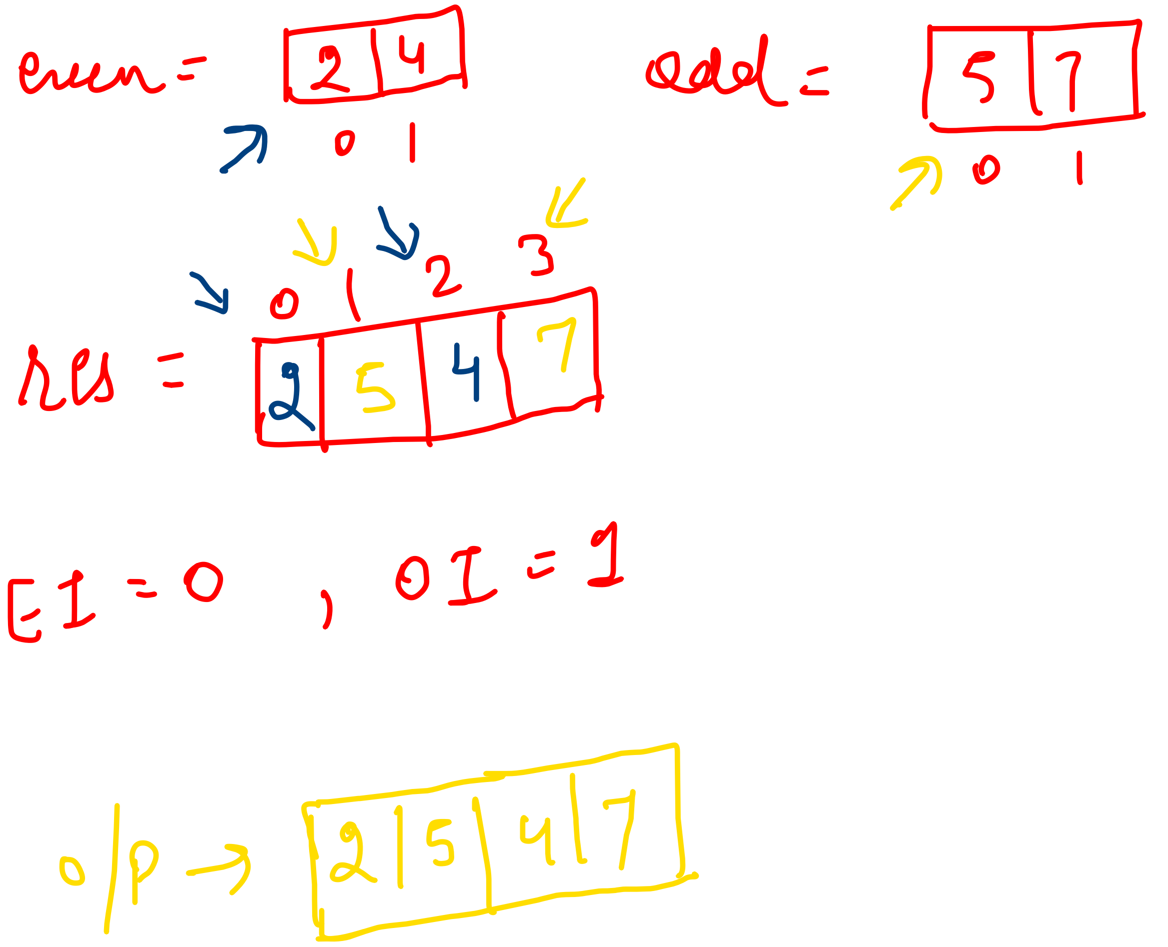
$n=4$



```java
    Arrays.sort(even);
    Arrays.sort(odd);

    // create an result array
    int result[] = new int[n];
    int evenindex = 0;
    int oddindex = 1;

    // place even numbers
    for(int i = 0; i < evencount; i++){
        result[evenindex] = even[i];
        evenindex += 2;
    }

    // place odd numbers
    for(int i = 0; i < oddcount; i++){
        result[oddindex] = odd[i];
        oddindex += 2;
    }

    // print result array
    for(int i = 0; i < n; i++){
        System.out.print(result[i] + " ");
    }
}
```

n

n



$TC \rightarrow O(n \log n) + n$

$SC \rightarrow O(n)$

$$arr = \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \\ \hline 2 & 3 & 3 & 3 & 4 & 5 \end{array}$$

curr = 2
count = 1

3 == 2 F

count > (n+1)/2
1 > — F

curr = 3
count = ~~1~~ ~~2~~
3

3 == 3
3 == 3
4 == 3 F

3 >= (n+1)/2 y
3 > 2 T

maj = 3
found = t

o/p -> 3

curr = 4
count = 1

5 == 4

curr = 5
count = 1

1 > (n+1)/2