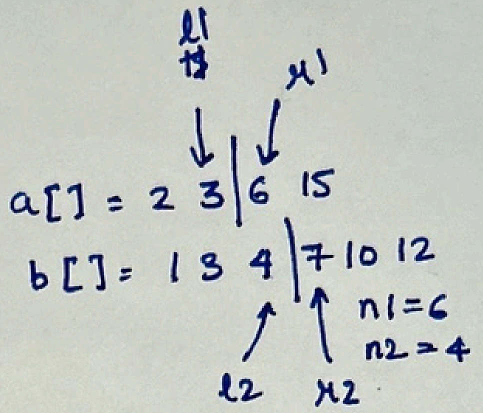
CPL - 2

Shreyans Tatiya

16010123325



4+6=10/2=5

```
Q1. a[], b[]
    if (a.length > b.length) {
        int n1 = a.length;
          n2 = b.length;
    }
    else {
        ~~int~~ n1 = b.length;
    }     n2 = a.length;

    n = n1+n2;
    left = (n1+n2+1) /2;    → left half
    low=0, high = n1;
    while (low <= high) {
        mid 1 = (low + high) /2;
        mid 2 = left - mid1;
                                                   max of a
      l1 ~~lt~~ = (mid1 >0) ? a[mid1-1] : Integer.MIN_VALUE;
                                                   max of b
      l2 = (mid2 >0) ? b[mid2-1] : Integer.MIN_VALUE;
      r1 = (mid1 <n1) ? a[mid1] : Integer.MAX_VALUE; ← min of a
      r2 = (mid2 < n2) ? b[mid2] : Integer.MAX_VALUE;
                                                   ← min of b
              <=
      if (~~lt~~l1 ~~≤~~ r2 && l2 <= r1) {
          if (n%2 ==1) return Math.max (l1, l2);
          else
              return ( (double) (Math.max (l1, l2) + Math.min (r1, r2)))/2.0;
      }
      else if (l1 > r2)
            high = mid 1 -1;
      else
          low = mid1 +1;
    }
```

Q2.
```
int cnt = 0;
    el = 0;
n = arr.length[];
for (i=0 → n) {
    if (cnt == 0) {
        cnt = 1;
        el = arr[i];
    }
    else if (el == arr[i]) {
        cnt++;
    }
    else {
        cnt--;
    }
}
```

2 2 1 1 1 2 2

el = ~~2~~ ~~1~~ 2

↳ cnt = ~~0~~ ~~1~~ ~~2~~ ~~1~~ ~~0~~ ~~1~~ ~~0~~ 1

S·complexity = O(1)

Q3.
```
cnt1 = 0; cnt2 = 0;
el1 = Integer.MIN_VALUE; el2 = Integer.MIN_VALUE; n = arr.length;
for (i=0 → n) {
    if (cnt1 == 0 && el2 != arr[i]) {
        cnt1 = 1;
        el1 = arr[i];
    }
    else if (cnt2 == 0 && el1 != arr[i]) {
        cnt2 = 1;
        el2 = arr[i];
    }
    elseif (arr[i] == el1) cnt1++;
    else if (arr[i] == el2) cnt2++;
    else {
        cnt1--;
        cnt2--;
    }
}
```

2 1 1 3 1 4 5 6

cnt1 = 1

el1 = 2

cnt2 = ~~0~~ ~~1~~ 2

el2 = ~~min~~ 1

Scomplexity = O(1)