Quiz

arr

1	2	3	4	5	1

Arrays.sort(arr);

Arrays.sort(arr); // O(nlogn)

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

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }

    boolean ans = solve(arr, n);
    System.out.println(ans);
}

public static boolean solve(int[] arr, int n) {
    Arrays.sort(arr);
    for (int i = 0; i < n - 1; i++) {
        if ( arr[i] == arr[i + 1] ) {
            return true;
        }
    }
    return false;
}
```

T.C

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

$\approx O(n\log(n))$

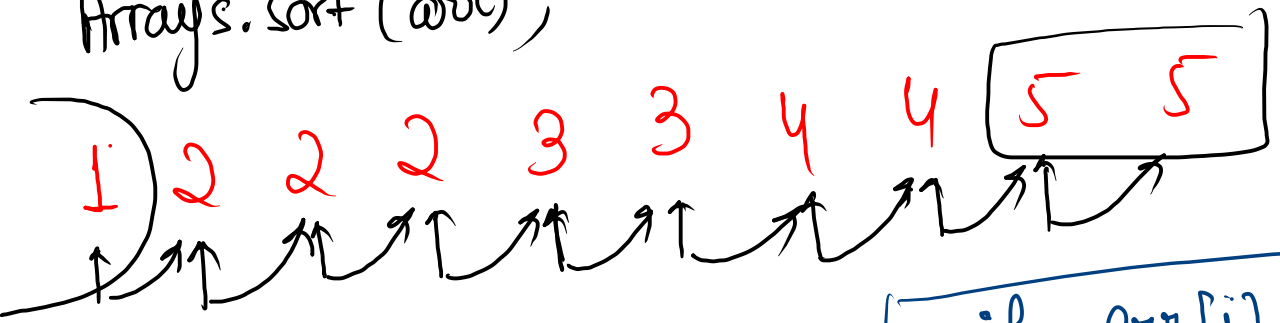
$n^3 > n^2 > n\log n > n > \log(n) > 1$

Max Count 3

5 1 2 4 2 2 3 3 4 5

max = ~~0~~ ~~1~~ (1) (2) 3

Arrays.sort(arr);



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

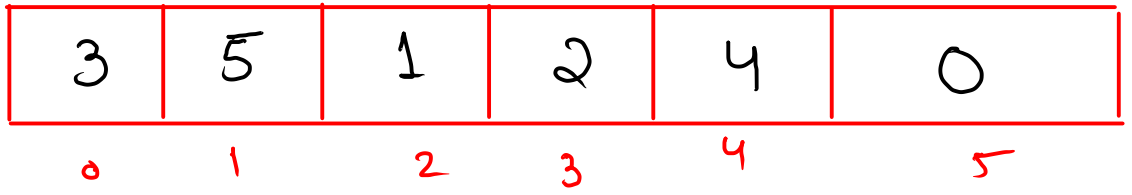
= ~~1~~ ~~2~~ ~~1~~ ~~2~~ ~~1~~ ~~2~~

1

count, ans, max

```
if arr[i] == arr[i+1]
    { count++;
else
    { if(count > max)
          max = count
    { count = 1
```

↳ Sorting // Arrays.sort(———)

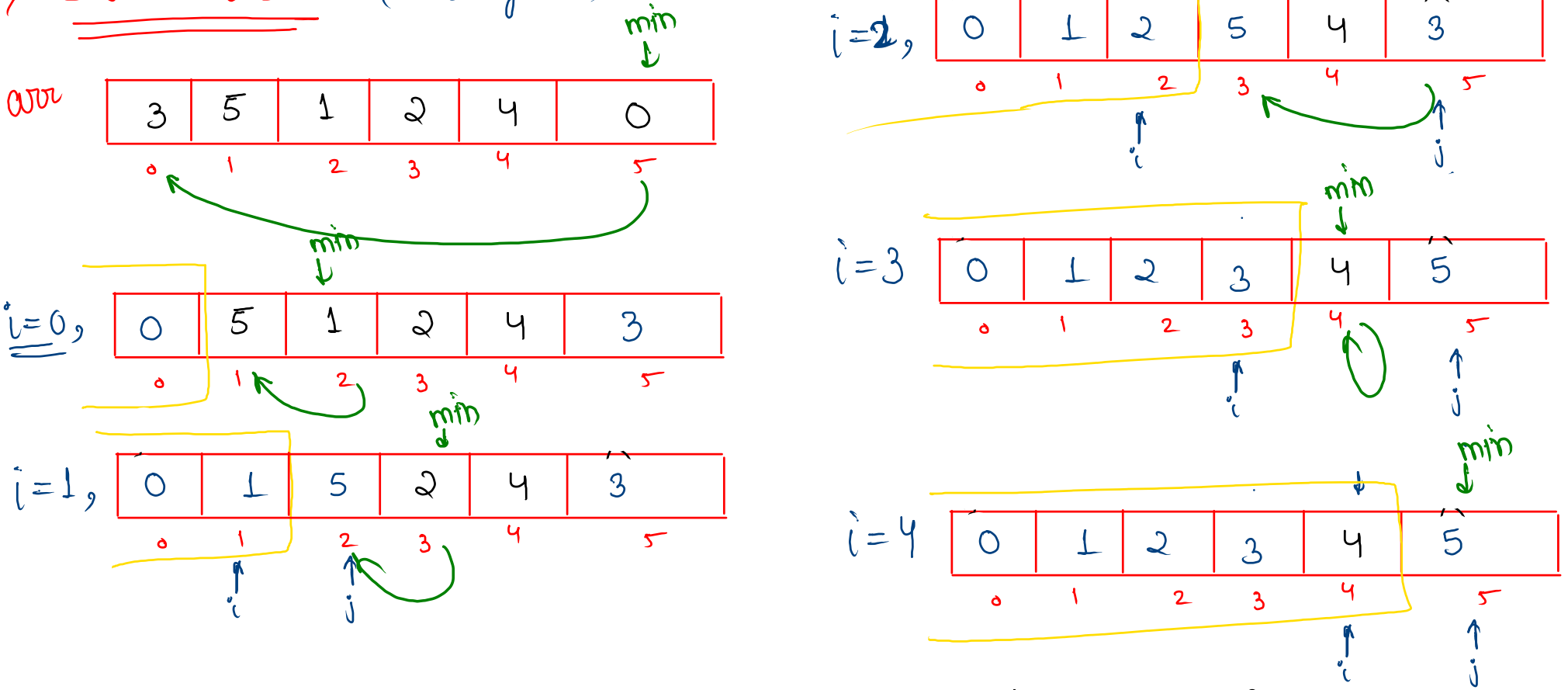


1) Bubble sort
2) Insertion sort
3) Selection sort

} used to arrange our array either in ascending or decending order

first 3

⇒ Selection sort (accending order)



selection sort :- each time find the smallest value from unsorted/remaining and swap it with first unsorted value.

selection sort

```java
public static void main(String[] args) {
    int[] arr = { 3, 5, 1, 2, 4, 0 };
    int n = arr.length;
    for (int i = 0; i < n - 1; i++) {
        int mini = i;
        for (int j = i + 1; j < n; j++) {
            if ( arr[j] < arr[mini] ) {
                mini = j;
            }
        }
        swap( arr, i, mini );
    }

    for (int i = 0; i < n; i++) {
        System.out.println(arr[i]);
    }
}

public static void swap(int[] arr, int x, int y) {
    int temp = arr[x];
    arr[x] = arr[y];
    arr[y] = temp;
}
```

$T.C = O(n^2)$

$S.C = O(1)$