**ARRAY PROBLEMS**

**LARGEST ELEMENT**

// **OPTIMAL**

1. Initialize the n = arr.length;
2. Max = arr[o]
3. Loop through I is not n
4. Check if arr[i] > max
5. Update the max
6. Else do nothing

**SECOND LARGEST ELEMENT**

**// BRUTE FORCE**

1. Initialize the smax = -1
2. Max = arr[0]
3. Loop from n-1 till >=0
4. Check the last element is not equal to the current element
5. Update the second max as the arr[i]

**// BETTER SOLUTION**

1. First find the ax element by max function
2. Store the max element using the variable
3. Loop from starting and till the I Is not equal to the n
4. If the element Is > ssamll and != largest
5. Update the second largest as the current array element

**// OPTIMAL CODE**

1. Loop from starting and till the n
2. If arr[i] > max
3. Update the smax as the max and max as the arr[i]
4. For the last check arr[i] < max and greated than the smallest
5. Update the small as the arr[i]

**SORT THE ELEMENTS**

**// OPTIMAL CODE**

1. Loop starting from the 1 and till n // 1st element is already soterd
2. If a[i] >= a[i-1]
3. Simply do nothing
4. Else return false