1. Implement all common array operations that includes traversal like finding maximum, minimum, sum, product, linear search, reverse array. (do all this solution in single class and try to reuse the loop for function).
2. Write a code that swaps every alternate index element in an array.
   * if array = [10,20,30,40,50,60] => output array => [20,10,40,30,60,50]
   * If array = [10,20,30,40,50] => output array => [20,10,40,30,50]
3. Find unique value in odd size array. Every other element is present twice in array only a single value there with 1 occurrence. Return that value.
   * Array = [ 10,20,25,20,10] then answer =>25.
   * Array = [ 20,30,40,45,40,30,20] then answer =>45.
4. Find whether the values in an array are in unique number of occurrence or not
   * If array = [1,2,3,4] => returns false because 1,2,3,4 elements are repeating one time.
   * If array = [1,2,2,3,3,3] => return true because every value has unique number of occurrence
5. Find duplicate value in array which contains 1 to n-1 values. You can also consider the same problem which doesn't contain 1 to n-1 values.
   * If array = [1,3,2,4,3] => output => 3
   * If array = [1,2,3,2] => output => 2
6. Find Intersection of 2 sorted arrays. Intersection means the same elements both arrays can have
   * If array-1 = [1,2,5,7] and array-2 = [5,6,7] => output => [5,7]
   * If array-1 = [1,2,3,4,5] and array-2 = [5,6] => output => [5]
7. Find Intersection of 3 sorted arrays
8. Find pairs of indexes that gives sum equals to target.
   * Index should be returned in sorted order
   * Example: - array => [1,2,3,4,5] and target => 6 Then answer => [ [ 0,4], [ 1,3]]
9. Find triplet of indexes that gives sum equals to target.
10. sort an array with values 0,1.
11. sort an array with values 0,1,2