Arrays - Assignment Solutions

1.

```
const twoSum = (array, target) => {
      let indexes = [];

      for(let i = 0; i < array.length; i++){
      for(let j = i + 1; j < array.length; j++){
      if (array[i] + array[j] === target) {
            indexes.push(array[i]);
            indexes.push(array[j]);
      }
      }
      }
      return indexes;
}
console.log(twoSum([1,2,3],5)); // [2, 3]
```

2.

```
function findOneFrequencyElement(A, ar_size)
{
	for (let i = 0; i < ar_size; i++) {
	let count = 0;

	for (let j = 0; j < ar_size; j++) {
		if (A[i] == A[j]) {
		count++;
		}
	}
	if (count == 1) {
		return A[i];
	}
	}

	// if no element exist at most once
	return -1;
}

let ar = [ 2, 3, 5, 4, 5, 3, 4 ];
let n = 7;

console.log("Single Frquency Element In This Array "
		+ findOneFrequencyElement(ar, n)); // Single Frquency Element In
This Array 2
```

3.

```
function medianOfArray (array) {
var length = array. length;
if (length % 2 == 1) {
return array[Math.floor(length/2)];
} else {
return (array[length/2]+array[length/2-1])/2;}
}
function medianOfTwoSortedArray (arr1, arr2, pos){
if (pos <= 0) {
return -1; }
if (pos == 1) {
return (arr1[0] + arr2[0]) / 2 ;
}
if (pos == 2) {
return (Math.max(arr1[0], arr2[0]) + Math.min(arr1[1],
arr2[1])) / 2 ;
}
var median1 = medianOfArray(arr1),
median2 = medianOfArray(arr2);
if (median1 == median2) {
return median1;
}
var evenOffset = pos % 2 == 0? 1: 0,
offsetMinus = Math.floor (pos/2) - evenOffset,
```

```
offsetPlus = Math.floor (pos/2) + evenOffset;

if (median1 < median2) {

return medianOfTwoSortedArray(arr1.slice(offsetMinus), arr2.

slice(offsetMinus), offsetPlus);

} else {

return medianOfTwoSortedArray(arr2.slice(offsetMinus), arr1.

slice(offsetMinus), offsetPlus);

}

}

medianOfTwoSortedArray([7,8,9],[1,2,3],3); // 5
```

4.

```
function commonElements (kArray) {

var hashmap = {},

last, answer = [];

for (var i = 0 , kArrayLength = kArray.length; i< kArrayLength;i++) {

var currentArray = kArray[i];

last = null

for (var j = 0 , currentArrayLen = currentArray.length; j<currentArrayLen;
j++) {

var currentElement = currentArray[j];

if (last != currentElement) {

if (! hashmap[currentElement]) {

hashmap[currentElement] = 1

} else {

hashmap[currentElement]++

}

}

last = currentElement;

}

}

for (var prop in hashmap) {

if (hashmap[prop] == kArray.length){

answer.push (parseInt (prop)) ;

}

}

return answer;
```

```
}

commonElements([[1 ,2 ,3 ],[1 ,2 ,3 ,4 ],[1 ,2 ]]); // [1, 2]
```

5.

```
var maxArea = function(H) {

      let ans = 0, i = 0, j = H.length-1

      while (i < j) {

      ans = Math.max(ans, Math.min(H[i], H[j]) * (j - i))

      H[i] <= H[j] ? i++ : j--

      }

      return ans

};

console.log("Maximum Area In The Container Filled With Water Is",maxArea([1,8,6,2,5,4,8,3,7])); // Maximum Area In The Container Filled With Water Is 49
```