1.

a.Print odd numbers in an array.

=> for (let i=1; i<=10; i++){

if(i%2!=0){

console.log(i);

}};

b. Convert all the strings in title caps in a string array.

=>

let stringArray = ["hello world", "javaScript is fun", "openai GPT-4"];

let titleCasedArray = convertArrayToTitleCaps(stringArray);

console.log(titleCasedArray);

c. Sum of all numbers in an array.

=> function sumOfArray(arr) {

let sum = 0; for (let i = 0; i < arr.length; i++) {

sum += arr[i];

}

console.log("Sum of all numbers in the array:", sum);

}

d. Return all the prime numbers in an array.

=>

function isPrime(num) { if (num <= 1) return false;

if (num <= 3) return true;

if (num % 2 === 0 || num % 3 === 0) return false;

for (let i = 5; i \* i <= num; i += 6) {

if (num % i === 0 || num % (i + 2) === 0) return false;

}

return true; }

e. Return all the palindromes in an array.

=>

function isPalindrome(str) {

const cleanedStr = str.replace(/[^a-zA-Z0-9]/g, '').toLowerCase();

return cleanedStr === cleanedStr.split('').reverse().join('');

}

function getPalindromes(arr) {

return arr.filter(isPalindrome);

}

f. Return median of two sorted array of the same size.

=>

function findMedianSortedArrays(arr1, arr2) {

let n = arr1.length;

function median(array) {

let len = array.length;

if (len % 2 === 0) {

return (array[len / 2 - 1] + array[len / 2]) / 2;

} else {

return array[Math.floor(len / 2)];

}

}

if (n === 1) {

return (arr1[0] + arr2[0]) / 2;

}

let low = 0, high = n;

while (low <= high) {

let partitionX = Math.floor((low + high) / 2);

let partitionY = n - partitionX;

let maxX = (partitionX === 0) ? -Infinity : arr1[partitionX - 1];

let minX = (partitionX === n) ? Infinity : arr1[partitionX];

let maxY = (partitionY === 0) ? -Infinity : arr2[partitionY - 1];

let minY = (partitionY === n) ? Infinity : arr2[partitionY];

if (maxX <= minY && maxY <= minX) {

if ((n \* 2) % 2 === 0) {

return (Math.max(maxX, maxY) + Math.min(minX, minY)) / 2;

} else {

return Math.max(maxX, maxY);

}

} else if (maxX > minY) {

high = partitionX - 1;

} else {

low = partitionX + 1;

}

}

throw new Error("Input arrays are not sorted or have different sizes.");

}

g. Remove duplicates from an array.

=>

function removeDuplicates(arr) {

duplicates let uniqueSet = new Set(arr);

let uniqueArray = [...uniqueSet];

return uniqueArray; }

h. Rotate an array by k times.

=>

function rotateArray(arr, k) {

const n = arr.length;

k = k % n;

return arr.slice(-k).concat(arr.slice(0, n - k));

}

2.

a. print odd numbers in an array.

=>

const printOddNumbers = (arr) => {

arr.forEach(num => {

if (num % 2 !== 0) {

console.log(num);

}

});

};

b. Convert all the strings in title caps in a string array

=>

const toTitleCase = str => str.replace(/\w\S\*/g, word => word.charAt(0).toUpperCase() + word.slice(1).toLowerCase());

const convertArrayToTitleCase = arr => arr.map(toTitleCase);

c. Sum of all numbers in an array.

=>

const sumArray = (arr) => arr.reduce((accumulator, currentValue) => accumulator + currentValue, 0);

d. Return all the prime numbers in an array.

=>

const getPrimes = (arr) => {

const isPrime = (num) => {

if (num <= 1) return false;

for (let i = 2; i <= Math.sqrt(num); i++) {

if (num % i === 0) return false;

}

return true;

return arr.filter(isPrime);

};

e. Return all the palindromes in an array.

=>

const getPalindromes = (arr) => arr.filter(str => str === str.split('').reverse().join(''));