1..Given an unsorted array of integers nums, return the length of the longest consecutive elements sequence.

nums = [100,4,200,1,3,2]

Output: 4

var nums=[100,4,200,1,3,2];

function longestConsecutive(nums){

let count=1;

for(i=0;i<nums.length;i++){

for(j=i;j<nums.length;j++){

if(nums[i]>nums[j]){

let temp=nums[i];

nums[i]=nums[j];

nums[j]=temp;

}

}

}

// //console.log(nums)

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

if(nums[i+1]===nums[i]+1){

count=count+1;

}

}

return count;

}

longestConsecutive([100,4,200,1,3,2]);

//2 Input: nums = [2,0,2,1,1,0]

//Output: [0,0,1,1,2,2]

function sort(nums){

for(i=0;i<nums.length;i++){

for(j=i;j<nums.length;j++){

if(nums[i]>nums[j]){

let temp=nums[i];

nums[i]=nums[j];

nums[j]=temp;

}

}

}

return nums;

}

sort( [2,0,1])

// Explanation: Buy on day 2 (price = 1) and sell on day 3 (price = 5), profit = 5-1 = 4.

// Then buy on day 4 (price = 3) and sell on day 5 (price = 6), profit = 6-3 = 3.

// Total profit is 4 + 3 = 7.

var Maxprofit = function(prices) {

let totalProfit = 0;

let dayProfit;

for (i = 0; i < prices.length; i++) {

dayProfit = prices[i + 1] - prices[i]

if (dayProfit > 0) {

totalProfit = totalProfit + dayProfit

}

}

return totalProfit;

}

Maxprofit([7, 1, 5, 3, 6, 4])

/. Find whether an array is subset of another array

// Input: arr1[] = {11, 1, 13, 21, 3, 7}, arr2[] = {11, 3, 7, 1}

//add all sum of all array

let array = [1, 2, 3, 4, 5]

function sumArray(arr) {

let sum = 0;

for (i = 0; i < array.length; i++) {

sum = sum + array[i]

// console.log(sum)

}

return sum

}

sumArray()

//Program to find out prime numbers in a given array.

// //Find whether an array is subset of another array

const arr1 = [10, 5, 2, 23, 19], arr2 = [19, 5, 2]

function isSubset(arr1, arr2) {

let count = 0;

for (let i = 0; i < arr1.length; i++) {

for (let j = 0; j < arr2.length; j++) {

if (arr1[i] === arr2[j]) {

// console.log(arr1[i])

count = count + 1;

}

}

}

if (count === arr2.length) {

console.log("happy")

} else {

console.log("sad")

}

}

isSubset([10, 5, 2, 23, 19], [19, 5, 2]);

//Rotate the given array by n times toward right

function rightRotate(arr,n){

for(let i=0 ;i<n ;i++){

let last;

last=arr[arr.length-1]; //length alwase count 1 and index count for 0

for(let j=arr.length-1;j>0;j--){

arr[j]=arr[j-1];

}

arr[0]=last;

}

console.log(arr)

}

rightRotate([1, 2, 3, 4, 5], 2);

//delete ripited letter ABBAA

function Deleted(str){

let del=0;

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

if(str[i]===str[i+1]){

del=del+1

}else{

console.log("Different")

}

} console.log(del)

}

Deleted("AAABBB")