# Solutions-Exercises: Functions and Logic Flow

## 1.Leap Year

function leapYear() {

let button = document.querySelector('#exercise>button');

button.addEventListener('click', checkYear);

function checkYear(){

let input = document.querySelector('#exercise>input');

let outputText = '';

if (checkLeapYear(Number(input.value))){

outputText = 'Leap Year';

} else {

outputText = 'Not Leap Year'

}

console.log(checkLeapYear(Number(input.value)));

let yearDiv = document.querySelector('#year');

let yearDivH2 = yearDiv.querySelector('h2');

let yearDivtDiv = yearDiv.querySelector('div');

yearDivH2.textContent = outputText;

yearDivtDiv.textContent = input.value;

input.value = '';

}

function checkLeapYear(year){

return ((year % 4 == 0) && (year % 100 != 0)) || (year % 400 == 0);

}

}

|  |
| --- |
| function leapYear() { |
|  | let input = document.querySelector('#exercise>input'); |
|  | let checkButton = document.querySelector('#exercise>button'); |
|  |  |
|  | checkButton.addEventListener('click', checkYear); |
|  |  |
|  | function checkYear() { |
|  | let outputText = ''; |
|  | if(Number(input.value) % 4 === 0) { |
|  | outputText = "Leap Year"; |
|  | } else { |
|  | outputText = "Not Leap Year"; |
|  | } |
|  | let yearDivElement = document.querySelector('#year'); |
|  | let h2Element = yearDivElement.querySelector('h2'); |
|  | let divElement = yearDivElement.querySelector('div') |
|  |  |
|  | h2Element.textContent = outputText; |
|  | divElement.textContent = input.value; |
|  | input.value = ''; |
|  | } |
|  | } |

## 2.Simple Number Validator

function validate() {

document.querySelector('#exercise>fieldset>div>button').addEventListener('click', checkValidNumber);

function checkValidNumber(){

let input = document.querySelector('#exercise>fieldset>div>input').value;

let lastDigit = Number(input[input.length - 1]);

let weightArray = [2, 4, 8, 5, 10, 9, 7, 3, 6];

let sum = 0;

for (let i = 0; i < 9; i++) {

let inputDigit = input[i];

let weightDigit = weightArray[i];

sum += inputDigit \* weightDigit;

}

let reminder = sum % 11;

if (reminder === 10){

reminder = 0;

}

let response = document.getElementById('response');

if (lastDigit === reminder) {

response.textContent = 'This number is Valid!';

} else {

response.textContent = 'This number is NOT Valid!';

}

}

}

|  |
| --- |
| function validate() { |
|  | const weights = [2, 4, 8, 5, 10, 9, 7, 3, 6]; |
|  | let sum = 0; |
|  | let response = document.querySelector('#response'); |
|  | document.querySelector('#exercise>fieldset>div>button').addEventListener('click', checkIfSimpleNumber); |
|  |  |
|  | function checkIfSimpleNumber() { |
|  | let inputField = document.querySelector('#exercise>fieldset>div>input').value; |
|  | const lastDigit = inputField[inputField.length - 1]; |
|  |  |
|  | for (let i = 0; i < inputField.length - 1; i++) { |
|  | const currentNumber = inputField[i]; |
|  | const currentWeight = weights[i]; |
|  |  |
|  | sum += currentNumber \* currentWeight; |
|  | } |
|  |  |
|  | let remainder = sum % 11; |
|  | if(remainder === 10) { |
|  | remainder = 0; |
|  | } |
|  |  |
|  | appendNumber(Number(lastDigit), remainder); |
|  |  |
|  | function appendNumber(lastDigit, remainder) { |
|  | if(lastDigit === remainder) { |
|  | response.textContent = "This number is Valid!"; |
|  | } else { |
|  | response.textContent = "This number is NOT Valid!"; |
|  | } |
|  | } |
|  | } |
|  | } |

## 3.EGN Generator

function validate() {

let button = document.querySelector('#exercise>div>button');

button.addEventListener('click', generateEGN);

function generateEGN() {

let weights = [2, 4, 8, 5, 10, 9, 7, 3, 6];

let year = document.getElementById('year').value;

let yearEnding = year.slice(2);

let months = { January: '01', February: '02', March: '03', April: '04', May: '05', June: '06', July: '07', August: '08', September: '09', October: '10', November: '11', December: '12' };

let month = months[document.getElementById('month').value];

let day = document.getElementById('date').value;

if (day.length === 1) {

day = '0' + day;

}

let region = document.getElementById('region').value;

let regionBeginning = region.slice(0, 2);

let gender = document.getElementById('male').checked ? 2 : 1;

if (year && month && day && gender && region && year >= 1900 && year <= 2100 && region >= 43 && region <= 999) {

let egn = yearEnding + month + day + regionBeginning + gender;

let egnSum = 0;

for (let i = 0; i < 9; i++) {

egnSum += Number(egn[i]) \* Number(weights[i]);

}

let validCheckNumber = egnSum % 11;

if (validCheckNumber > 9) {

validCheckNumber = 0;

}

egn += validCheckNumber;

document.getElementById('egn').textContent = `Your EGN is: ${egn}`;

document.getElementById('year').value = '';

document.getElementById('month').value = '';

document.getElementById('date').value = '';

document.getElementById('male').checked = false;

document.getElementById('female').checked = false;

document.getElementById('region').value = '';

}

}

}

function validate() {

let monthObj = {

"January": '01', "February": '02', "March": '03',

"April": '04', "May": '05', "June": '06',

"July": '07', "August": '08', "September": '09',

"October": '10', "November": '11', "December": '12'

}

document.getElementsByTagName('button')[0]

.addEventListener('click', (e) => {

let year = document.getElementById('year');

let month = document.getElementById('month');

let day = document.getElementById('date');

let male = document.getElementById('male');

let female = document.getElementById('female');

let regCode = document.getElementById('region');

let egn = '';

egn += year.value.slice(-2);

egn += monthObj[month.value];

egn += ('0' + day.value).slice(-2);

egn += regCode.value.slice(0, 2);

let gender = male.checked ? '2' : (female.checked ? '1' : '');

egn += gender;

let egnCheck = egn.split('').map(Number);

let weightPosition = [2, 4, 8, 5, 10, 9, 7, 3, 6];

let weightSumArr = [];

for (let i = 0; i < 9; i++) {

weightSumArr.push(egnCheck[i] \* weightPosition[i]);

}

let weightSumNum = weightSumArr.reduce((a, b) => a + b);

let weightSum = weightSumNum % 11;

egn += weightSum != 10 ? weightSum : '0';

if ((+year.value >= 1900 && +year.value <= 2100)

&& (+regCode.value >= 43 && +regCode.value <= 999)

&& gender && month.value != 'Select a month') {

document.getElementById('egn').textContent = `Your EGN is: ${egn}`;

year.value = '';

month.options[month.selectedIndex].selected = false;

day.value = '';

male.checked = false;

female.checked = false;

regCode.value = '';

}

})

}

|  |
| --- |
| 100/100  function solve() { |
|  | let button = document.querySelector('#exercise button'); |
|  | button.addEventListener('click', generateEGN); |
|  |  |
|  | function generateEGN() { |
|  | let egn; |
|  | let weights = [2, 4, 8, 5, 10, 9, 7, 3, 6]; |
|  | var getYear = () => document.getElementById('year').value; |
|  | var getMonth = () => { |
|  | months = { |
|  | January: '01', |
|  | February: '02', |
|  | March: '03', |
|  | April: '04', |
|  | May: '05', |
|  | June: '06', |
|  | July: '07', |
|  | August: '08', |
|  | September: '09', |
|  | October: '10', |
|  | November: '11', |
|  | December: '12', |
|  | }; |
|  | return months[document.getElementById('month').value]; |
|  | }; |
|  | var getDay = () => ('0' + (document.getElementById('date').value)).slice(-2); |
|  | var getGender = () => document.getElementById('male').checked ? 2 : 1; |
|  | var getRegion = () => document.getElementById('region').value; |
|  |  |
|  |  |
|  | if (getYear() && getMonth() && getDay() && getGender() && getRegion() && |
|  | getYear() >= 1900 && getYear() <= 2100 && getRegion() >= 43 && getRegion() <= 999) { |
|  |  |
|  | egn = (getYear()).slice(2) + getMonth() + getDay() + (getRegion()).slice(0, 2) + getGender(); |
|  |  |
|  | let egnSum = 0; |
|  | for (let i = 0; i < 9; i++) { |
|  | if (egn[i] !== 0) { |
|  | egnSum += (Number(egn[i]) \* Number(weights[i])); |
|  | } |
|  | } |
|  | let validChecksum = egnSum % 11 > 9 ? 0 : egnSum % 11; |
|  |  |
|  | egn += validChecksum; |
|  |  |
|  | document.getElementById('egn').textContent = `Your EGN is: ${egn}`; |
|  |  |
|  | document.getElementById('year').value = ''; |
|  | document.getElementById('date').value = ''; |
|  | document.getElementById('region').value = ''; |
|  | document.getElementById('month').value = ''; |
|  | document.getElementById('male').checked = false; |
|  | document.getElementById('female').checked = false; |
|  | } |
|  | } |
|  | } |

|  |
| --- |
| function validate() { |
|  |  |
|  | // 50 out of 100 :( |
|  |  |
|  | let button = document.querySelector('button'); |
|  | button.addEventListener('click', (e) => { |
|  | const year = document.querySelector('#year'); |
|  |  |
|  | const month = document.querySelector('#month'); |
|  | const day = document.querySelector('#date'); |
|  | const gender = document.querySelector('input[name="gender"]:checked'); |
|  | const regionCode = document.querySelector('#region'); |
|  |  |
|  | if(Number(year.value) < 1900 || Number(year.value) > 2100 || |
|  | Number(regionCode.value) < 43 || Number(regionCode.value > 999) || |
|  | gender.value === null || |
|  | day.value === '' || |
|  | month.value === '' |
|  | ) { |
|  | clearAllFields(); |
|  | return; |
|  | } |
|  |  |
|  | const yearToCode = year.value.slice(2); |
|  | const monthToObj = { |
|  | January: '01', |
|  | February: '02', |
|  | March: '03', |
|  | April: '04', |
|  | May: '05', |
|  | June: '06', |
|  | July: '07', |
|  | August: '08', |
|  | September: '09', |
|  | October: ' 10', |
|  | November: '11', |
|  | December: '12', |
|  | } |
|  | const monthToCode = monthToObj[month.value]; |
|  | const dayToCode = day.value > 10 ? day.value : '0' + day.value; |
|  | const genderToObj = { |
|  | Male: 2, |
|  | Female: 1, |
|  | } |
|  | const genderToCode = genderToObj[gender.value]; |
|  | let regionCodeToCode; |
|  | if(regionCode.value >= 43 && regionCode.value <= 99) { |
|  | regionCodeToCode = regionCode.value; |
|  | } else { |
|  | regionCodeToCode = regionCode.value.slice(0, -1); |
|  | } |
|  | const numberToFindLastDigit = yearToCode + |
|  | monthToCode + |
|  | dayToCode + |
|  | regionCodeToCode + |
|  | genderToCode; |
|  |  |
|  | const EGN = findLastNumber(numberToFindLastDigit); |
|  |  |
|  | let egnPElement = document.querySelector('#egn'); |
|  | egnPElement.textContent = `Your EGN is: ${EGN}`; |
|  |  |
|  | clearAllFields(); |
|  |  |
|  | function findLastNumber(numberToStr) { |
|  | let sum = 0; |
|  | const weights = [2, 4, 8, 5, 10, 9, 7, 3, 6]; |
|  | for (let i = 0; i < numberToStr.length; i++) { |
|  | const currentNumber = numberToStr[i]; |
|  | const currentWeight = weights[i]; |
|  |  |
|  | sum += currentNumber \* currentWeight; |
|  | } |
|  |  |
|  | let remainder = sum % 11; |
|  | if(remainder === 10) { |
|  | remainder = 0; |
|  | } |
|  | return numberToStr + remainder; |
|  | } |
|  |  |
|  | function clearAllFields() { |
|  | year.value = ''; |
|  | month.value = ''; |
|  | date.value = ''; |
|  | document.getElementById('male').checked = false; |
|  | document.getElementById('female').checked = false; |
|  |  |
|  | regionCode.value = ''; |
|  |  |
|  | } |
|  | }) |
|  | } |

4.Cooking Numbers

function solve() {

let inputElement = document.querySelector('#exercise>input');

let buttons = document.querySelectorAll('button');

let buttonNames = {Chop: chop, Dice: dice, Spice: spice, Bake: bake, Fillet: fillet};

Array.from(buttons).forEach(button => button.addEventListener('click', buttonNames[button.textContent]));

//let inputNumber = Number(inputElement.value);

//console.log(document.querySelector('#exercise>input').value);

let outputElement = document.getElementById('output');

let currentNumber;

function updateNumber(){

let inputNumber = Number(inputElement.value);

return Number(currentNumber) || inputNumber;

//return Number(currentNumber) || Number(inputElement.value);

}

function chop(){

currentNumber = updateNumber() / 2;

outputElement.textContent = currentNumber;

}

function dice(){

currentNumber = Math.sqrt(updateNumber());

outputElement.textContent = currentNumber;

}

function spice(){

currentNumber = updateNumber() + 1;

outputElement.textContent = currentNumber;

}

function bake(){

currentNumber = updateNumber() \* 3;

outputElement.textContent = currentNumber;

}

function fillet(){

currentNumber = updateNumber() \* 0.8;

outputElement.textContent = currentNumber;

}

}

|  |
| --- |
| function solve() { |
|  | let chopBtn = document.querySelectorAll('#operations button')[0]; |
|  | let diceBtn = document.querySelectorAll('#operations button')[1]; |
|  | let spiceBtn = document.querySelectorAll('#operations button')[2]; |
|  | let bakeBtn = document.querySelectorAll('#operations button')[3]; |
|  | let filletBtn = document.querySelectorAll('#operations button')[4]; |
|  | let resultElement = document.querySelector('#output'); |
|  |  |
|  | function chop() { |
|  | let number = Number(document.querySelector('#exercise input').value); |
|  | if (resultElement.textContent) { |
|  | resultElement.textContent /= 2; |
|  | } else { |
|  | resultElement.textContent = number / 2; |
|  | } |
|  | } |
|  |  |
|  | function dice() { |
|  | let number = Number(document.querySelector('#exercise input').value); |
|  | if (resultElement.textContent) { |
|  | resultElement.textContent = Math.sqrt(resultElement.textContent); |
|  | } else { |
|  | resultElement.textContent = Math.sqrt(number); |
|  | } |
|  | } |
|  |  |
|  | function spice() { |
|  | let number = Number(document.querySelector('#exercise input').value); |
|  | if (resultElement.textContent) { |
|  | resultElement.textContent = Number(resultElement.textContent) + 1; |
|  | } else { |
|  | resultElement.textContent = Number(number + 1); |
|  | } |
|  | } |
|  |  |
|  | function bake() { |
|  | let number = Number(document.querySelector('#exercise input').value); |
|  | if (resultElement.textContent) { |
|  | resultElement.textContent \*= 3; |
|  | } else { |
|  | resultElement.textContent = number \* 3; |
|  | } |
|  | } |
|  |  |
|  | function fillet() { |
|  | let number = Number(document.querySelector('#exercise input').value); |
|  | if (resultElement.textContent) { |
|  | resultElement.textContent = Number(resultElement.textContent) \* 0.8; |
|  | } else { |
|  | resultElement.textContent = number \* 0.8; |
|  | } |
|  | } |
|  | chopBtn.addEventListener('click', chop); |
|  | diceBtn.addEventListener('click', dice); |
|  | spiceBtn.addEventListener('click', spice); |
|  | bakeBtn.addEventListener('click', bake); |
|  | filletBtn.addEventListener('click', fillet); |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | } |

|  |
| --- |
| function solve() { |
|  | let number = document.querySelector('#exercise>input'); |
|  | const buttons = document.querySelectorAll('button'); |
|  | let currentNumber; |
|  | let outputElement = document.querySelector('#output'); |
|  |  |
|  | const buttonsNames = { |
|  | Chop: chop, |
|  | Dice: dice, |
|  | Spice: spice, |
|  | Bake: bake, |
|  | Fillet: fillet, |
|  | } |
|  |  |
|  | Array.from(buttons).forEach(btn => btn.addEventListener('click', buttonsNames[btn.textContent])); |
|  |  |
|  | function chop() { |
|  | currentNumber = updateNumber() / 2; |
|  | outputElement.textContent = currentNumber; |
|  | } |
|  |  |
|  | function dice() { |
|  | currentNumber = Math.sqrt(updateNumber()); |
|  | outputElement.textContent = currentNumber; |
|  | } |
|  |  |
|  | function spice() { |
|  | currentNumber = updateNumber() + 1; |
|  | outputElement.textContent = currentNumber; |
|  | } |
|  |  |
|  | function bake() { |
|  | currentNumber = updateNumber() \* 3; |
|  | outputElement.textContent = currentNumber; |
|  | } |
|  |  |
|  | function fillet() { |
|  | currentNumber = updateNumber() \* 0.8; |
|  | outputElement.textContent = currentNumber; |
|  | } |
|  |  |
|  | function updateNumber() { |
|  | return Number(currentNumber) || Number(number.value); |
|  | } |
|  | } |

## 5.Cards Generator

function solve() {

let button = document.querySelector('button');

button.addEventListener('click', addCards);

function addCards(){

let cardValues = ['2', '3', '4', '5', '6', '7', '8', '9', '10', 'J', 'Q', 'K', 'A'];

let fromCardInputElement = document.getElementById('from');

let toCardInputElement = document.getElementById('to');

let cardSuitSelectElement = document.querySelector('select');

let cardsSectionElement = document.getElementById('cards');

let fromCardIndex = cardValues.indexOf(fromCardInputElement.value);

let toCardIndex = cardValues.indexOf(toCardInputElement.value);

let cardSuit = '';

if(cardSuitSelectElement.value.includes('Hearts')){

cardSuit = '&hearts;';

} else if(cardSuitSelectElement.value.includes('Spades')){

cardSuit = '&spades;';

} else if(cardSuitSelectElement.value.includes('Diamonds')){

cardSuit = '&diamond;';

} else if(cardSuitSelectElement.value.includes('Clubs')){

cardSuit = '&clubs;';

}

for (let i = fromCardIndex; i <= toCardIndex; i++) {

let cardValue = cardValues[i];

let cardDiv = document.createElement('div');

cardDiv.classList.add('card');

let upSuitP = document.createElement('p');

upSuitP.innerHTML = cardSuit;

cardDiv.appendChild(upSuitP);

let cardValueP = document.createElement('p');

cardValueP.innerHTML = cardValue;

cardDiv.appendChild(cardValueP);

let downSuitP = document.createElement('p');

downSuitP.innerHTML = cardSuit;

cardDiv.appendChild(downSuitP);

cardsSectionElement.appendChild(cardDiv);

}

fromCardInputElement.value = '';

toCardInputElement.value = '';

}

}

|  |
| --- |
| function solve() { |
|  | let button = document.querySelector('#exercise button'); |
|  | button.addEventListener('click', generateCards); |
|  |  |
|  | function generateCards() { |
|  | let from = document.getElementById('from').value; |
|  | let to = document.getElementById('to').value; |
|  | let suitChoosen = document.querySelector('#exercise select').selectedIndex; |
|  | let unicode = getSuit(suitChoosen); |
|  | let resultElement = document.getElementById('cards'); |
|  |  |
|  | let numbers = ['2', '3', '4', '5', '6', '7', '8', '9', '10', 'J', 'Q', 'K', 'A']; |
|  | let cards = []; |
|  |  |
|  | if (numbers.indexOf(from) !== -1 && numbers.indexOf(to) !== -1) { |
|  | for (let i = numbers.indexOf(from); i <= numbers.indexOf(to); i++) { |
|  | let card = { |
|  | suit: unicode, |
|  | value: numbers[i] |
|  | }; |
|  | cards.push(card); |
|  | } |
|  | for (let card of cards) { |
|  | let div = document.createElement('div'); |
|  | let leftP = document.createElement('p'); |
|  | let middleP = document.createElement('p'); |
|  | let rightP = document.createElement('p'); |
|  | div.setAttribute('class', 'card'); |
|  | div.appendChild(leftP); |
|  | div.appendChild(middleP); |
|  | div.appendChild(rightP); |
|  | leftP.innerHTML = card.suit; |
|  | middleP.innerHTML = card.value; |
|  | rightP.innerHTML = card.suit; |
|  | resultElement.appendChild(div); |
|  | } |
|  |  |
|  | } |
|  | } |
|  |  |
|  | function getSuit(suitChoosen) { |
|  | switch (suitChoosen) { |
|  | case 0: |
|  | unicode = '&hearts;'; |
|  | break; |
|  | case 1: |
|  | unicode = '&spades;'; |
|  | break; |
|  | case 2: |
|  | unicode = '&diamond;'; |
|  | break; |
|  | case 3: |
|  | unicode = '&clubs;'; |
|  | break; |
|  | } |
|  |  |
|  | return unicode; |
|  | } |
|  |  |
|  | } |

|  |
| --- |
| function solve() { |
|  | const allCards = ['2', '3', '4', '5', '6', '7', '8', '9', '10', 'J', 'Q', 'K', 'A']; |
|  |  |
|  | const button = document.querySelector('button'); |
|  | button.addEventListener('click', addCards); |
|  |  |
|  | function addCards() { |
|  | let fromCard = document.querySelector('#from'); |
|  | let toCard = document.querySelector('#to'); |
|  | let cardSuit = document.querySelector('select'); |
|  | let sectionElement = document.querySelector('section#cards'); |
|  |  |
|  | const fromCardIndex = allCards.indexOf(fromCard.value); |
|  | const toCardIndex = allCards.indexOf(toCard.value); |
|  | let suit = ''; |
|  | if(cardSuit.value.includes('Hearts')) { |
|  | suit = '&hearts;'; |
|  | } else if(cardSuit.value.includes('Spades')) { |
|  | suit = '&spades;'; |
|  | } else if(cardSuit.value.includes('Diamonds')) { |
|  | suit = '&diamond;'; |
|  | } else if(cardSuit.value.includes('Clubs')) { |
|  | suit = '&clubs;'; |
|  | } |
|  |  |
|  | for (let index = fromCardIndex; index <= toCardIndex; index++) { |
|  | const element = allCards[index]; |
|  | let divElement = document.createElement('div'); |
|  | divElement.classList.add('card'); |
|  | let firstSuitP = document.createElement('p'); |
|  | let secondSuitP = document.createElement('p'); |
|  | firstSuitP.innerHTML = suit; |
|  | secondSuitP.innerHTML = suit |
|  |  |
|  | let cardElement = document.createElement('p'); |
|  | cardElement.textContent = element; |
|  |  |
|  | divElement.appendChild(firstSuitP); |
|  | divElement.appendChild(cardElement); |
|  | divElement.appendChild(secondSuitP); |
|  |  |
|  | sectionElement.appendChild(divElement); |
|  | } |
|  |  |
|  | fromCard.value = ''; |
|  | toCard.value = ''; |
|  | } |
|  | } |

## 6.Greatest Common Divisor

function greatestCD() { //75/100 in Judge

let firstNumber = document.getElementById('num1').value;

let secondNumber = document.getElementById('num2').value;

let result = findGreatestCommonDivisor(firstNumber, secondNumber);

let resultElement = document.getElementById('result');

resultElement.innerHTML = result;

function findGreatestCommonDivisor(firstNumber, secondNumber){

let lowerNumber = Math.min(firstNumber, secondNumber);

for(let i = lowerNumber; i > 0; i--){

if (firstNumber % i === 0 && secondNumber % i === 0){

return i;

}

}

}

}

|  |
| --- |
| function greatestCD() { |
|  | let num1 = Number(document.getElementById('num1').value); |
|  | let num2 = Number(document.getElementById('num2').value); |
|  | let result = document.getElementById('result'); |
|  |  |
|  | function gcd(a, b) { |
|  | if ( ! b) { |
|  | return a; |
|  | } |
|  |  |
|  | return gcd(b, a % b); |
|  | } |
|  | result.innerHTML = gcd(num1, num2); |
|  | } |

## 7.Binary Search \*

function binarySearch() {

let inputIntegerArrayElement = document.getElementById('arr');

let inputrString = inputIntegerArrayElement.value;

//inputrString = inputrString.substring(1, inputrString.length - 1);

inputIntegerArray = inputrString.split(', ').map(Number);

//console.log(inputIntegerArray);

let numberToSearchElement = document.getElementById('num');

let numberToSearch = Number(numberToSearchElement.value);

let resultElement = document.getElementById('result');

if (inputIntegerArray.indexOf(numberToSearch) === -1){

resultElement.innerHTML = numberToSearch + ' is not in the array';

} else {

resultElement.innerHTML = `Found ${numberToSearch} at index ${inputIntegerArray.indexOf(numberToSearch)}`;

}

}

|  |
| --- |
| function binarySearch() { |
|  | let arr = document.getElementById('arr').value; |
|  | let num = Number(document.getElementById('num').value); |
|  | let result = document.getElementById('result'); |
|  | let val = false; |
|  | let index; |
|  | arr = arr.split(', '); |
|  | //console.log(arr); |
|  | for(let i = 0; i < arr.length;i++){ |
|  | if(+arr[i] === num){ |
|  | val = true; |
|  | index = i; |
|  | } |
|  | } |
|  | if(val === true){ |
|  | result.innerHTML = `Found ${num} at index ${index}`; |
|  | }else{ |
|  | result.innerHTML = `${num} is not in the array`; |
|  | } |
|  | } |

## 8.Hailstone sequence

function getNext() {

let inputNumber = Number(document.getElementById('num').value);

let resultElement = document.getElementById('result');

resultElement.textContent = doHailstoneSequence(inputNumber) + ' ';

function doHailstoneSequence(n){

let sequence = [n];

while(n !== 1){

if (n % 2 === 0){

n /= 2;

} else {

n = n \* 3 + 1;

}

sequence.push(n);

}

return sequence.join(' ');

}

}

|  |
| --- |
| function getNext() { |
|  | let num = Number(document.getElementById('num').value); |
|  | let result = document.getElementById('result'); |
|  |  |
|  | function hailStoneSeq(n){ |
|  | var seq=[n]; |
|  |  |
|  | while(n!==1){ |
|  | if(n%2===0){ |
|  | n/=2; |
|  | } else{ |
|  | n=(n\*3)+1; |
|  | } |
|  |  |
|  | seq.push(n); |
|  | } |
|  | return seq.join(' '); |
|  | } |
|  |  |
|  | result.innerHTML = hailStoneSeq(num) + ' '; |
|  |  |
|  | } |

## 9.Dot Product \*\*

function solve() {

let firstMatrix = JSON.parse(document.getElementById('mat1').value);

let secondMatrix = JSON.parse(document.getElementById('mat2').value);

let secondMatrixTransposed = transpose(secondMatrix);

let resultElement = document.getElementById('result');

let result = multiplyMatrices(firstMatrix, secondMatrixTransposed);

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

let p = document.createElement('p');

resultElement.appendChild(p);

p.textContent = result[i].join(', ');

}

function transpose(matrix){

return matrix[0].map((col, i) => matrix.map(row => row[i]));

}

function multiplyMatrices(m1, m2){

let resultMatrix = [];

for (let i = 0; i < m1.length; i++) {

resultMatrix[i] = [];

for (let j = 0; j < m2[0].length; j++) {

let sum = 0;

for (let k = 0; k < m1[0].length; k++) {

sum += m1[i][k] \* m2[k][j];

}

resultMatrix[i][j] = sum;

}

}

return resultMatrix;

}

}

|  |
| --- |
| function solve() { |
|  | let matrixOne = JSON.parse(document.getElementById('mat1').value); |
|  | let matrixTwo = JSON.parse(document.getElementById('mat2').value); |
|  | let resultElement = document.getElementById('result'); |
|  |  |
|  | let result = multiplyMatrices(matrixOne, transpose(matrixTwo)); |
|  |  |
|  | for (let i = 0; i < result.length; i++) { |
|  | let p = document.createElement('p'); |
|  | resultElement.appendChild(p); |
|  | p.innerHTML = result[i].join(', '); |
|  | } |
|  |  |
|  | function multiplyMatrices(m1, m2) { |
|  | var result = []; |
|  | for (var i = 0; i < m1.length; i++) { |
|  | result[i] = []; |
|  | for (var j = 0; j < m2[0].length; j++) { |
|  | var sum = 0; |
|  | for (var k = 0; k < m1[0].length; k++) { |
|  | sum += m1[i][k] \* m2[k][j]; |
|  | } |
|  | result[i][j] = sum; |
|  | } |
|  | } |
|  | return result; |
|  | } |
|  |  |
|  | function transpose(matrix) { |
|  | return matrix[0].map((col, i) => matrix.map(row => row[i])) |
|  | } |
|  |  |
|  | } |

## 10.Factors

function solve() {

let number = Number(document.getElementById('num').value);

let resultElement = document.getElementById('result');

resultElement.textContent = getFactors(number);

function getFactors(number) {

let factors = [];

for (let i = 1; i <= number; i++) {

if (number % i === 0){

factors.push(i);

}

}

return factors.join(' ');

}

}

|  |
| --- |
| function solve() { |
|  | let num = Number(document.getElementById('num').value); |
|  | let result = []; |
|  | for(let i = 1;i<= num;i++){ |
|  | if(num % i === 0){ |
|  | result.push(i); |
|  | } |
|  | } |
|  | let finalRes = document.getElementById('result'); |
|  | finalRes.innerHTML = result.join(' '); |
|  | } |