Министерство образования Республики Беларусь

Белорусско-Российский университет

Кафедра «Программное обеспечение информационных технологий»

Отчёт по лабораторной работе № 6

По дисциплине “Технологии Интернет-программирования”

“Обработка массивов”

Выполнил ст. гр. АСОИ-191 Недюхин В. А.

Проверила Выговская Н. В.

Могилев 2022

**Цель:** Изучение методов обработки массивов.

**Вариант 3 (12)**

Задача 1

Создайте функцию find(arr, value), которая ищет в массиве arr значение

value и возвращает его номер, если найдено, или -1, если не найдено.

Например:

arr = ["test", 2, 1.5, false];

find(arr, "test"); // 0

find(arr, 2); // 1

find(arr, 1.5); // 2

find(arr, 0); // -1

Задача 2

Создайте массив. Заполните его случайными значениями от А до В.

Значения А, В и размер массива вводятся на форме.

Напишите функцию map(fn, array), которая принимает на вход функцию

и массив, и обрабатывает каждый элемент массива этой функцией, возвращая

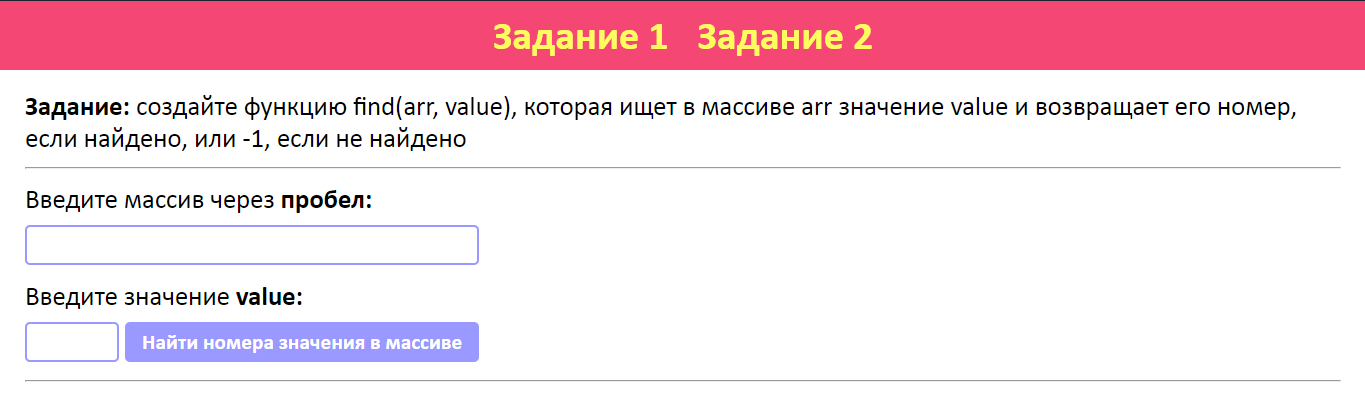
новый массив. Пример:

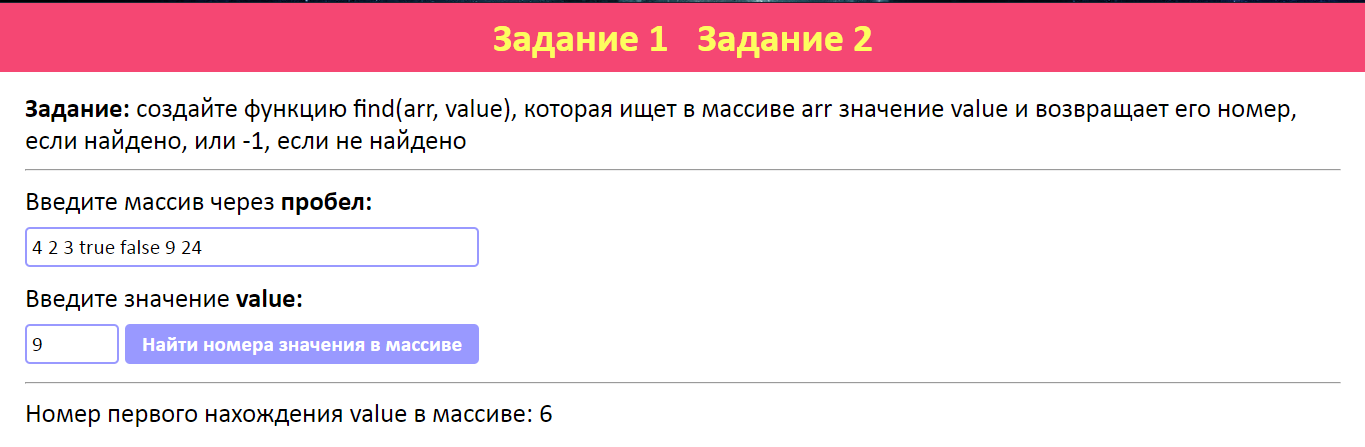
function square(x) { return x \* x; } // возведение в квадрат

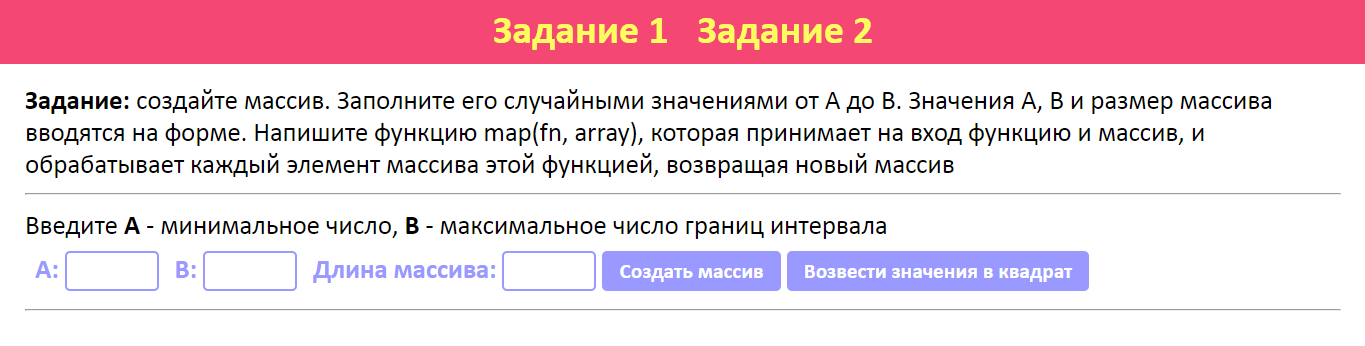
console.log(map(square, [1, 2, 3, 4])); // [1, 4, 9, 16]

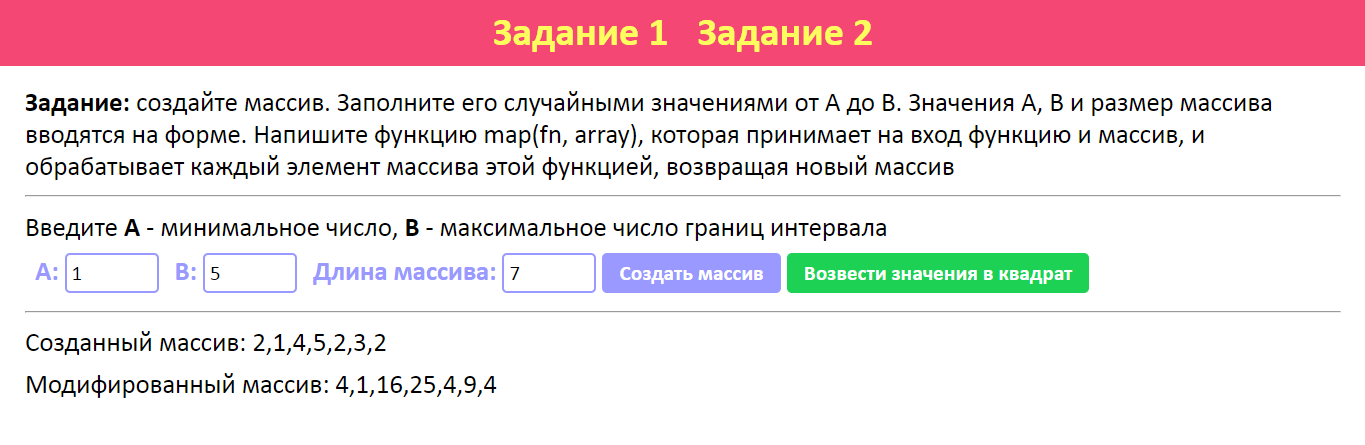
console.log(map(square, [])); // []

Функция не должна изменять переданный ей массив









Код:

Lab6\_Page1.html

<!DOCTYPE *html*>

<html *lang*="ru">

<head>

  <meta *charset*="UTF-8">

  <meta *http-equiv*="X-UA-Compatible" *content*="IE=edge">

  <meta *name*="viewport" *content*="width=device-width, initial-scale=1.0">

  <title>Lab 6</title>

  <link *rel*="stylesheet" *href*="./Lab6\_Style.css">

</head>

<body>

  <header>

    <div>

      <a *href*="./Lab6\_Page1.html">Задание 1</a>

      <a *href*="./Lab6\_Page2.html">Задание 2</a>

    </div>

  </header>

  <main>

    <p><strong>Задание: </strong>создайте функцию find(arr, value), которая ищет в массиве arr значение

      value и возвращает его номер, если найдено, или -1, если не найдено</p>

      <hr>

      <p>Введите массив через <strong>пробел:</strong></p>

      <input *class*="long" *id*="addArrayInput" *type*="text" *onchange*="addArray()"></input>

      <p>Введите значение <strong>value:</strong></p>

      <input *class*="short" *id*="addValueInput" *type*="text" *onchange*="addValue()"></input>

      <button *id*="findButton" *type*="button" *value*="Find index">Найти номера значения в массиве</button>

      <hr>

      <p *id*="array"></p>

      <p *id*="index"></p>

    </main>

  <script *src*="./Lab6\_Tusk1.js"></script>

</body>

</html>

Lab6\_Page2.html

<!DOCTYPE *html*>

<html *lang*="ru">

<head>

  <meta *charset*="UTF-8">

  <meta *http-equiv*="X-UA-Compatible" *content*="IE=edge">

  <meta *name*="viewport" *content*="width=device-width, initial-scale=1.0">

  <title>Lab 6</title>

  <link *rel*="stylesheet" *href*="./Lab6\_Style.css">

</head>

<body>

  <header>

    <div>

      <a *href*="./Lab6\_Page1.html">Задание 1</a>

      <a *href*="./Lab6\_Page2.html">Задание 2</a>

    </div>

  </header>

  <main>

    <p><strong>Задание: </strong>создайте массив. Заполните его случайными значениями от А до В.

      Значения А, В и размер массива вводятся на форме.

      Напишите функцию map(fn, array), которая принимает на вход функцию

      и массив, и обрабатывает каждый элемент массива этой функцией, возвращая

      новый массив</p>

    <form>

      <hr>

      <p>Введите <strong>A</strong> - минимальное число, <strong>B</strong> - максимальное число границ интервала</p>

      <label>A:</label>

      <input *class*="short" *id*="min" *type*="number" />

      <label>B:</label>

      <input *class*="short" *id*="max" *type*="number" />

      <label>Длина массива:</label>

      <input *class*="short" *id*="length" *type*="number" />

      <button *id*="generateButton" *type*="button" *value*="Generate array">Создать массив</button>

      <button *id*="modButton" *type*="button" *value*="Square each element">Возвести значения в квадрат</button>

      <hr>

    </form>

    <p *id*="array"></p>

    <p *id*="modArray"></p>

  </main>

  <script *src*="./Lab6\_Tusk2.js"></script>

</body>

</html>

Lab6\_Tusk1.js

let array = [];

let value;

let findButton = document.querySelector("#findButton");

let indexContainer = document.querySelector("#index");

findButton.addEventListener("click", () => showArrayAndIndex());

function addArray() {

    array = document.querySelector("#addArrayInput").value.split(" ");

}

function addValue() {

    value = document.querySelector("#addValueInput").value;

}

function find(*arr*, *value*) {

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

        if (*arr*[i] == *value*) return i + 1;

    }

    return -1;

*//return arr.indexOf(value);*

}

function showArrayAndIndex() {

    addArray();

    addValue();

    indexContainer.textContent = "Номер первого нахождения value в массиве: ";

    indexContainer.textContent += (find(array, value));

}

Lab6\_Tusk2.js

let array = new Array();

let modArray = new Array();

let generateButton = document.querySelector("#generateButton");

let modButton = document.querySelector("#modButton");

let arrayContainer = document.querySelector("#array");

let modArrayContainer = document.querySelector("#modArray");

generateButton.addEventListener("click", () => showGeneratedArray());

modButton.addEventListener("click", () => showModArray());

function map(*fn*, *array*) {

    const modArray = [];

*array*.forEach((*el*, *index*) => {

        modArray[*index*] = fn(*el*);

  });

  return modArray;

}

function generateRandomArray(*min*, *max*, *length*) {

    const array = [];

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

      array[i] = Math.floor(Math.random() \* (*max* - *min* + 1)) + *min*;

    }

    return array;

}

function showGeneratedArray() {

    arrayContainer.textContent = "Созданный массив: ";

    modArrayContainer.textContent = " ";

    const min = +document.querySelector("#min").value;

    const max = +document.querySelector("#max").value;

    const length = +document.querySelector("#length").value;

    array = generateRandomArray(min, max, length);

    arrayContainer.textContent += array;

}

function showModArray() {

    modArrayContainer.textContent = "Модифированный массив: ";

    const modArray = map((*el*) => *el* \*\* 2, array);

    modArrayContainer.textContent += modArray

}

Lab6\_Style.css

:root {

  --dark:#0f0f0f;

  --blackblue:#2d355b;

  --rock:#666666;

  --grey:#c9c9c9;

  --lightgrey:#e6e6e6;

  --white:#ffffff;

  --pink:#f5b2f5;

  --lily:#ff8ebd;

  --bubble:#f54773;

  --red:#ff0000;

  --peach:#ff4545;

  --lightred:#c4383e;

  --violet:#7030a0;

  --magenta:#cf10ff;

  --water:#0070c0;

  --blue:#2f54fc;

  --moon:#6666ff;

  --lavender:#9999ff;

  --steel:#eeeeff;

  --orange:#ffa42d;

  --yellow:#fdfd5e;

  --lightyellow:#ffff83;

  --darkgreen:#0e9135;

  --green:#1cd153;

}

body {

  width: 100vw;

  min-height: 100vh;

  margin: 0;

  display: flex;

  flex-direction: column;

}

header {

  margin: 0;

  padding: 10px 15px;

  display: flex;

  justify-content:space-around;

  background-color: var(--bubble);

}

header a {

  margin: 0;

  font: bold 30pt Calibri;

  color: var(--yellow);

}

header a {

  text-decoration: none;

}

header a:first-child {

  margin-left: 0px;

}

header a {

  margin-left: 25px;

}

header a:hover{

  color: var(--blue);

}

main{

  margin: 0 15px;

  padding: 10px;

  font: 20pt Calibri;

}

main p{

  margin: 10px 0;

}

main div{

  padding: 10px 0px;

}

main label:first-child{

  margin-left: 0px;

}

main label{

  margin-left: 10px;

  font-weight: bold;

  color:var(--lavender);

}

main input, button{

  margin-bottom: 5px;

  padding: 5px;

  font: 16pt Calibri;

  border: 2px solid var(--lavender);

  border-radius: 5px;

}

main input.short{

  width: 80px;

}

main input.long{

  width: 440px;

}

main input:hover{

  border: 2px solid var(--green);

}

main button, input[type="submit"]{

  padding: 5px 15px;

  font-weight: bold;

  color: white;

  border-radius: 5px;

  border-color: var(--lavender);

  background-color: var(--lavender);

}

main button:hover, input[type="submit"]:hover{

  cursor: pointer;

  border-color: var(--green);

  background-color: var(--green);

}