Министерство образования и науки Российской Федерации

ФГБОУ ВО «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

Лабораторная работа №1

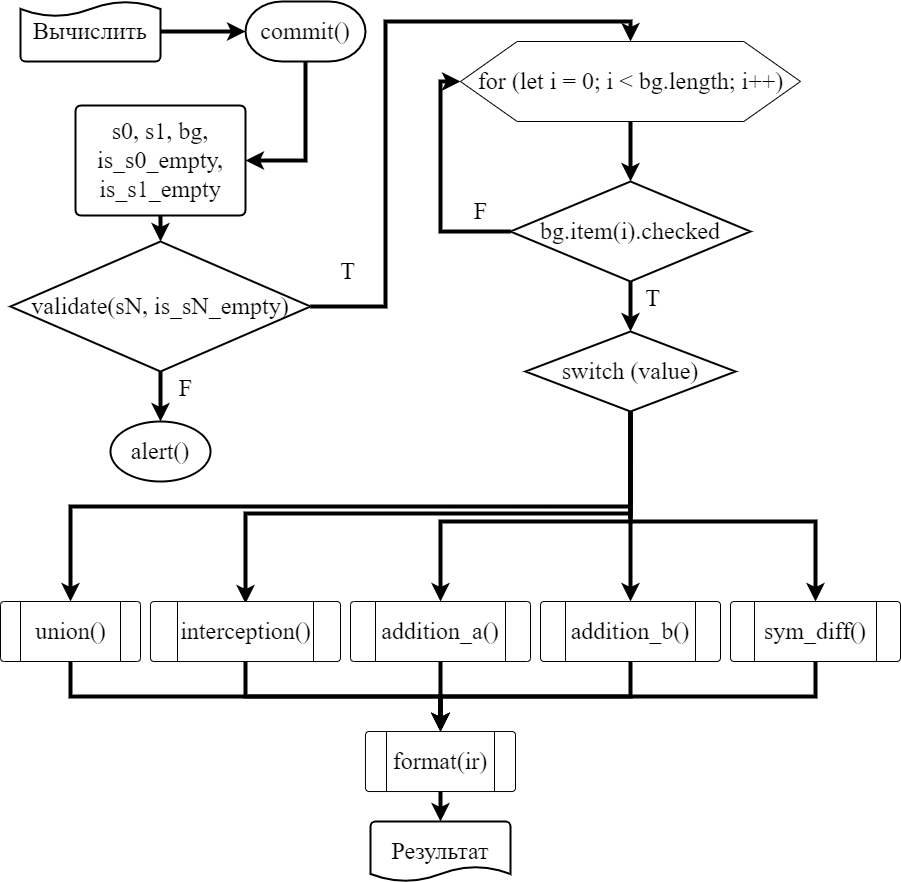
«Операции с множествами»

Выполнил студент

группы ИВТАСбд-11

Нуждов Д. А.

Ульяновск, 2022

**Постановка задачи**Разработать программу на JavaScript, выполняющую операции с двумя множествами (объединение, пересечение, дополнение, симметрическую разность). Элементы множеств должны выглядеть так: ibij, где i – четная цифра, b – буква, j – нечетная цифра. Сделать валидацию вводимых данных.  
  
**Особенности реализации**Работа выполнена в виде сайта. По нажатии на кнопку запускается функция, заданная в параметре onсlick, которая определяет тип операции и передает управление алгоритму, выполняющую ее. В конце результат выводится на экран.  
В случае ошибки ввода появляется оповещение.  
  
**Блок-схема**

**Интерфейс**Интерфейс сайта состоит из заголовка, двух полей ввода и двух флажков (показывают программе, что это множество пустое), группы переключателей  
(для выбора операции), поля результата и кнопки.

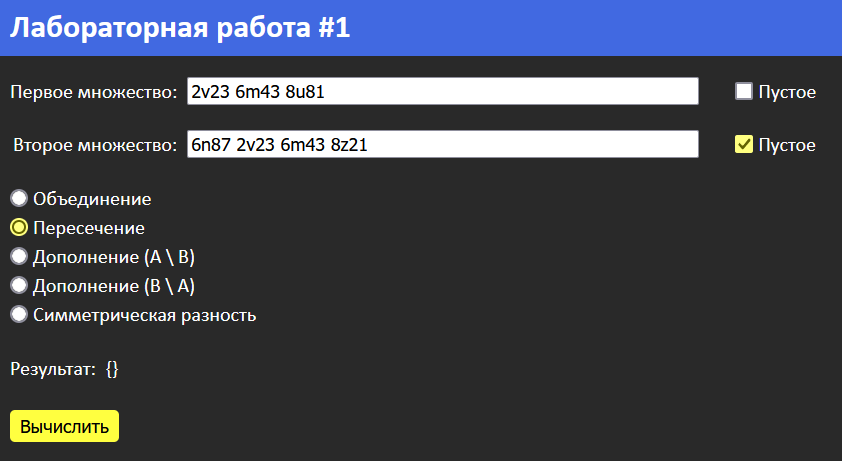
Рис. 1 – внешний вид

Рис. 2 – работа с пустым множеством

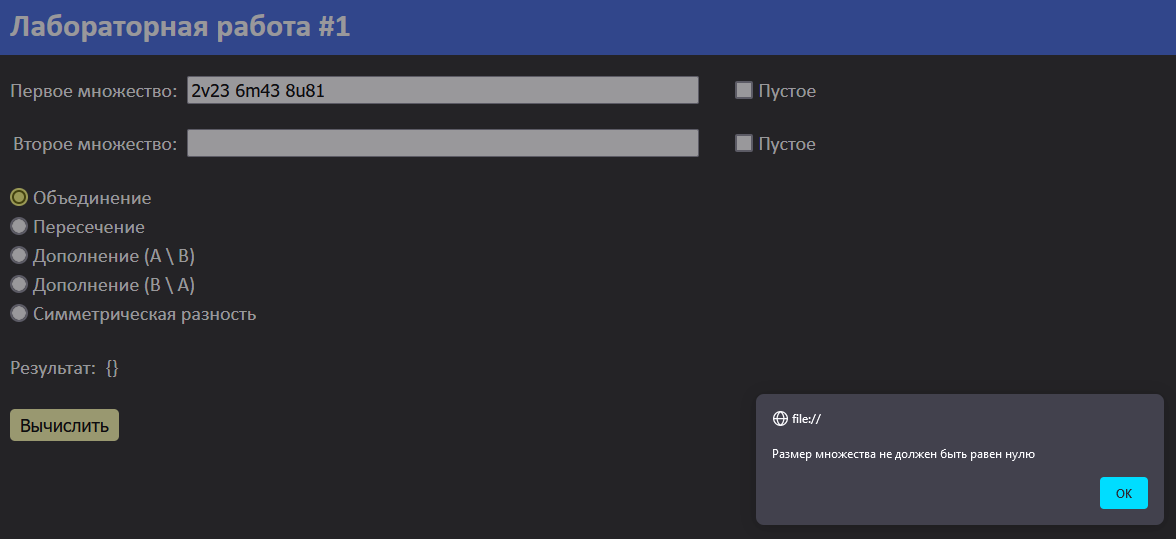
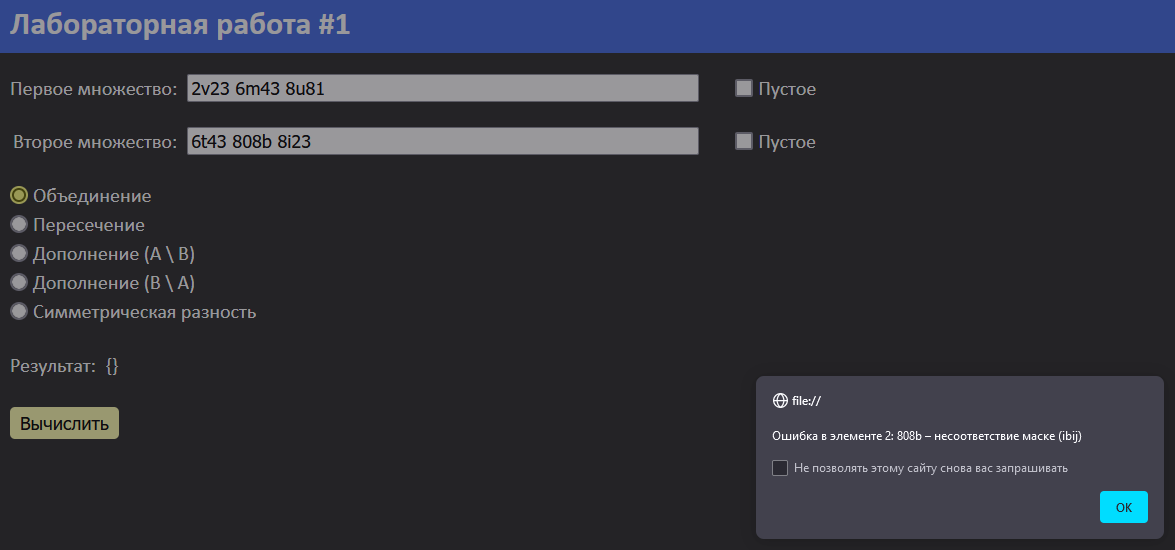
****Рис. 3 – ошибка неопределенного множества

Рис. 4 – ошибка несоответствия маски

**Выводы о проделанной работе**Программа справляется с поставленной задачей. Мной были усвоены знания по веб-разработке (верстка страниц, работа со стилями и разработка скриптов).  
  
**Список литературы**1) Веб-технологии для разработчиков (https://developer.mozilla.org/ru/docs/Web)

**Приложение***index.html*<html lang="ru">

<head>

<meta charset="utf-8"/>

<title>Лабораторная работа #1</title>

<link href="style.css" rel="stylesheet"/>

<script src="mod.js"></script>

</head>

<body>

<h1>Лабораторная работа #1</h1>

<table>

<tr>

<div class="f">

<div class="fif">

<label for="first">Первое множество: </label><input id="first" type="text">

</div>

<div class="fcb">

<input id="cb0" name="empty0" type="checkbox"><label for="cb0">Пустое</label>

</div>

</div>

</tr>

<tr>

<td>

<div class="s">

<div class="sif">

<label for="second">Второе множество: </label><input id="second" type="text">

</div>

<div class="scb">

<input id="cb1" name="empty1" type="checkbox"><label for="cb1">Пустое</label>

</div>

</div>

</td>

</tr>

</table>

<div id="rb" class="radio">

<p><input id="v0" name="operation" type="radio" value="union"><label for="v0">Объединение</label></p>

<p><input id="v1" name="operation" type="radio" value="intersection"><label for="v1">Пересечение</label></p>

<p><input id="v2" name="operation" type="radio" value="addition\_a"><label for="v2">Дополнение (A \ B)</label></p>

<p><input id="v3" name="operation" type="radio" value="addition\_b"><label for="v3">Дополнение (B \ A)</label></p>

<p><input id="v4" name="operation" type="radio" value="sym\_diff"><label for="v4">Симметрическая разность</label></p>

</div>

<div class="r">

<label>Результат: </label><label id="result"></label>

</div>

<button id="commit" onclick="commit();">Вычислить</button>

</body>

</html>  
  
*style.css*div {

margin-bottom: 10px;

}

h1 {

font-family: "Calibri", serif;

background-color: #4169e1;

padding: 10px;

color: white;

}

body {

margin: 0;

background-color: #292929;

height: 100%;

}

label {

margin-right: 5px;

font-size: 20px;

font-family: "Calibri", serif;

color: white;

}

#commit {

font-size: 18px;

margin-left: 10px;

border-width: 0;

border-radius: 5px;

background: #ffff80;

color: black;

padding: 5px 10px 5px 10px;

}

#commit:hover {

background: #ffffb0;

}

#commit:focus {

background: #ffff40;

}

#first {

width: 512px;

font-size: 18px;

margin-right: 32px;

margin-left: 5px;

}

#second {

width: 512px;

font-size: 18px;

margin-right: 32px;

margin-left: 5px;

}

.f {

padding-left: 10px;

display: flex;

align-items: center;

}

.f .fif {

display: flex;

align-items: baseline;

}

.f .fcb {

display: flex;

align-items: center;

}

.s {

margin-bottom: 10px;

padding-left: 10px;

display: flex;

align-items: center;

}

.s .sif {

display: flex;

align-items: baseline;

}

.s .scb {

display: flex;

align-items: center;

}

.r {

margin-bottom: 30px;

padding-left: 10px;

}

.radio {

margin-bottom: 30px;

font-size: 18px;

padding-left: 10px;

}

.radio input {

margin: 0 5px 0 0;

width: 18px;

height: 18px;

accent-color: #ffff80;

}

.radio p {

margin: 5px 0 5px 0;

display: flex;

align-items: center;

}

.f [type=checkbox] {

margin-right: 5px;

width: 18px;

height: 18px;

accent-color: #ffff80;

}

.s [type=checkbox] {

margin-right: 5px;

width: 18px;

height: 18px;

accent-color: #ffff80;

}  
 *mod.js*let s0, s1;

let is\_s0\_empty, is\_s1\_empty;

// форматирование в вид {2v45, 4b67, ibij, ...}

function format(v) {

return `{${v.join(", ")}}`;

}

// конвертация строки в массив

function split(v) {

let r = [];

let idx = 0;

if (v.length !== 0) {

while (idx-1 !== v.length) {

let s = "";

while (v[idx] !== " ") {

s += v[idx];

idx++;

if (idx === v.length) {

break;

}

}

if (s !== "") {

r.push(s);

}

idx++;

if (idx === v.length) {

break;

}

}

}

return r;

}

// проверка на пустоту и соответствие маске

function validate(v, s) {

let flag = true;

if (v.length === 0 && s === false) {

alert("Размер множества не должен быть равен нулю");

flag = false;

} else if (s) {

return true;

} else {

v.forEach((item, idx) => {

if (!(((parseInt(item[0]) % 2 === 0) && (parseInt(item[2]) % 2 === 0)) && (parseInt(item[3]) % 2 !== 0) && (item[1] >= "a" && item[1] <= "z"))) {

alert(`Ошибка в элементе ${idx+1}: ${item} – несоответствие маске (ibij)`);

flag = false;

}

});

}

return true === flag;

}

// объединение множеств

function union() {

if (is\_s1\_empty && !is\_s0\_empty) {

return s0;

} else if (is\_s0\_empty && !is\_s1\_empty) {

return s1;

} else if (is\_s0\_empty && is\_s1\_empty) {

return [];

} else {

let v = [];

s0.forEach(item => {

if (!(v.includes(item))) {

v.push(item);

}

});

s1.forEach(item => {

if (!(v.includes(item))) {

v.push(item);

}

});

return v;

}

}

// пересечение множеств

function intersection() {

if (is\_s0\_empty || is\_s1\_empty) {

return [];

} else {

let v = [];

s0.forEach(item => {

if (s1.includes(item) && !(v.includes(item))) {

v.push(item);

}

});

return v;

}

}

// дополнение A к B

function addition\_a() {

if (is\_s1\_empty && !is\_s0\_empty) {

return s0;

} else if (is\_s0\_empty && !is\_s1\_empty) {

return [];

} else if (is\_s0\_empty && is\_s1\_empty) {

return [];

} else {

let v = [];

s0.forEach(item => {

if (!(v.includes(item)) && !(s1.includes(item))) {

v.push(item);

}

});

return v;

}

}

// дополнение B к A

function addition\_b() {

if (is\_s0\_empty && !is\_s1\_empty) {

return s1;

} else if (is\_s1\_empty && !is\_s0\_empty) {

return [];

} else if (is\_s0\_empty && is\_s1\_empty) {

return [];

} else {

let v = [];

s1.forEach(item => {

if (!(v.includes(item)) && !(s0.includes(item))) {

v.push(item);

}

});

return v;

}

}

// симметрическая разность

function sym\_diff() {

if (is\_s0\_empty && !is\_s1\_empty) {

return s1;

} else if (is\_s1\_empty && !is\_s0\_empty) {

return s0;

} else if (is\_s0\_empty && is\_s1\_empty) {

return [];

} else {

let v = [];

s0.forEach(item => {

if (!(v.includes(item))) {

v.push(item);

}

});

s1.forEach(item => {

if (v.includes(item)) {

v.splice(v.indexOf(item), 1);

} else {

v.push(item);

}

});

return v;

}

}

function commit() {

s0 = split(document.getElementById("first").value);

s1 = split(document.getElementById("second").value);

is\_s0\_empty = document.getElementById("cb0").checked;

is\_s1\_empty = document.getElementById("cb1").checked;

let bg = document.getElementById("rb").getElementsByTagName("input");

if (validate(s0, is\_s0\_empty) && validate(s1, is\_s1\_empty)) {

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

if (bg.item(i).checked) {

let ir = undefined;

switch (bg.item(i).value) {

case "union":

ir = union(s0, s1);

break;

case "intersection":

ir = intersection(s0, s1);

break;

case "addition\_a":

ir = addition\_a(s0, s1);

break;

case "addition\_b":

ir = addition\_b(s0, s1);

break;

case "sym\_diff":

ir = sym\_diff(s0, s1);

break;

}

document.getElementById("result").innerHTML = format(ir);

break;

}

}

}

}