Министерство образования и науки Российской Федерации

ФГБОУ ВО Ульяновский государственный технический университет

Кафедра «Вычислительная техника»

Лабораторная работа №1

«Операции над множествами»

Выполнил студент

группы ИВТАСбд-11

Зимин Е.С.

Ульяновск, 2022

1. **Цель работы**

Написать программу, которая будет выполнять операции над множествами с проверкой на корректность ввода данных. А именно такие операции как пересечение, объединение, симметрическая разность, дополнение A/B и B/A. Первый символ массива должен быть – четной цифрой, второй – нечетной цифрой, третий и четвертый – цифрой.

1. **Краткое описание метода выполнения**

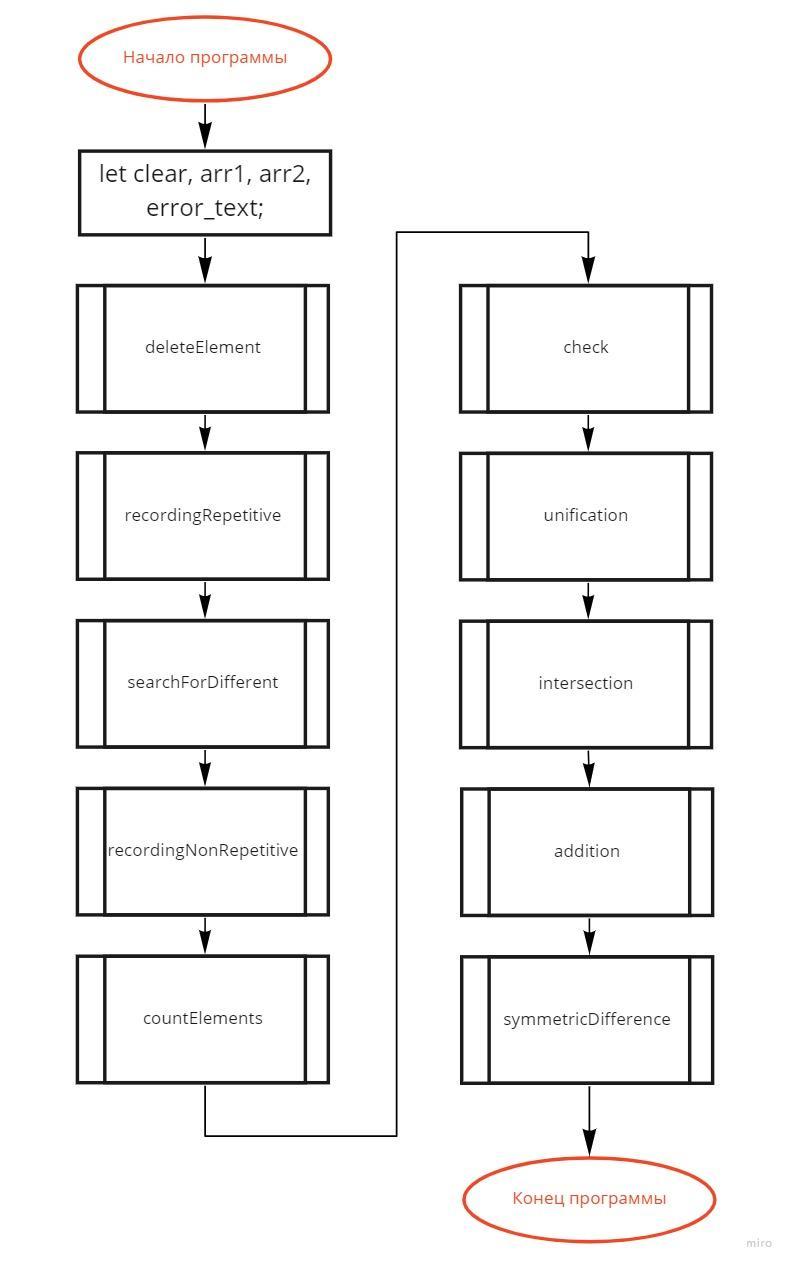
Программа реализована с помощью вспомогательных функций, которые отвечают за сравнение элементов двух разных массивов или сравнение элементов одного массива. Благодаря этому сравнению выполняются определённые действия (удаление ненужных, запись нужных элементов) по работе с массивами для достижения конечного результата.

В работе нужно сделать сайт на JavaScript и реализовать следующие операции над множествами: Объединение – множество, содержащее в себе все элементы других множеств без повторения. Пересечение – множество, содержащее в себе только элементы первого и второго множеств. Дополнением первого до второго называется множество, содержащее все элементы первого множества, которые не принадлежат второму множеству и наоборот. Симметрическая разность – множества, содержащее в себе элементы первого и второго множеств, которые не совпадают между собой.

Для ввода данных я создал два поля ввода и кнопки, отвечающие за вывод нужных множеств.

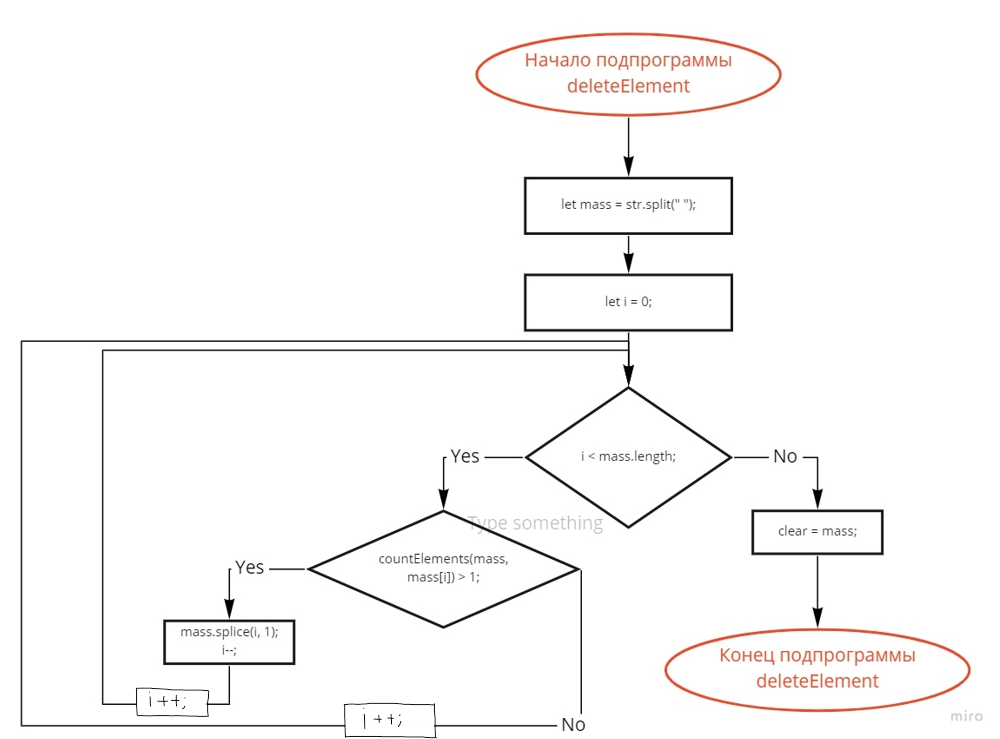
1. **Блок-схема алгоритма решения поставленной задачи.**

Основная часть:

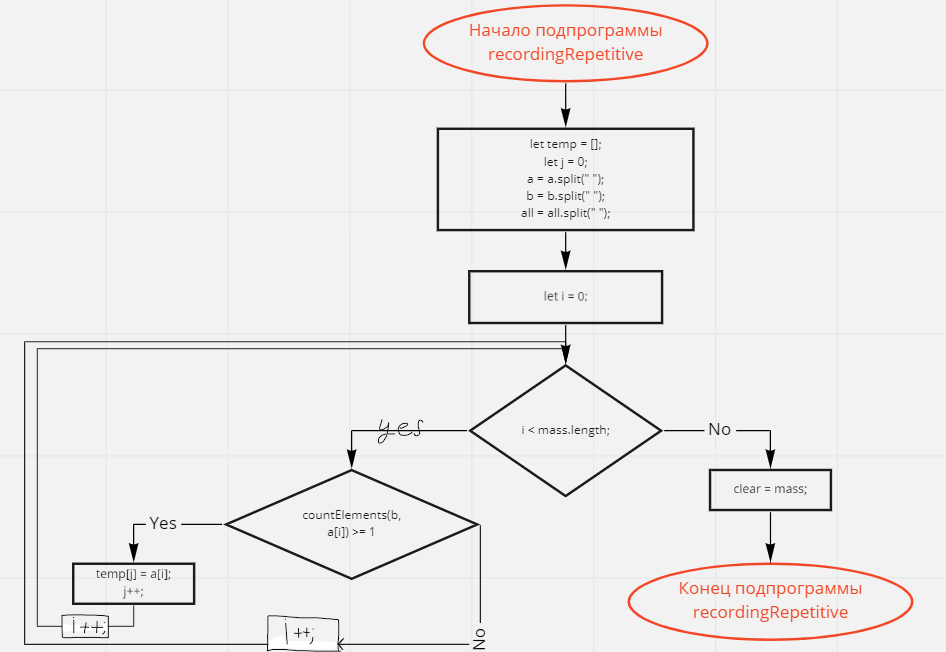


Подпрограммы (функции):

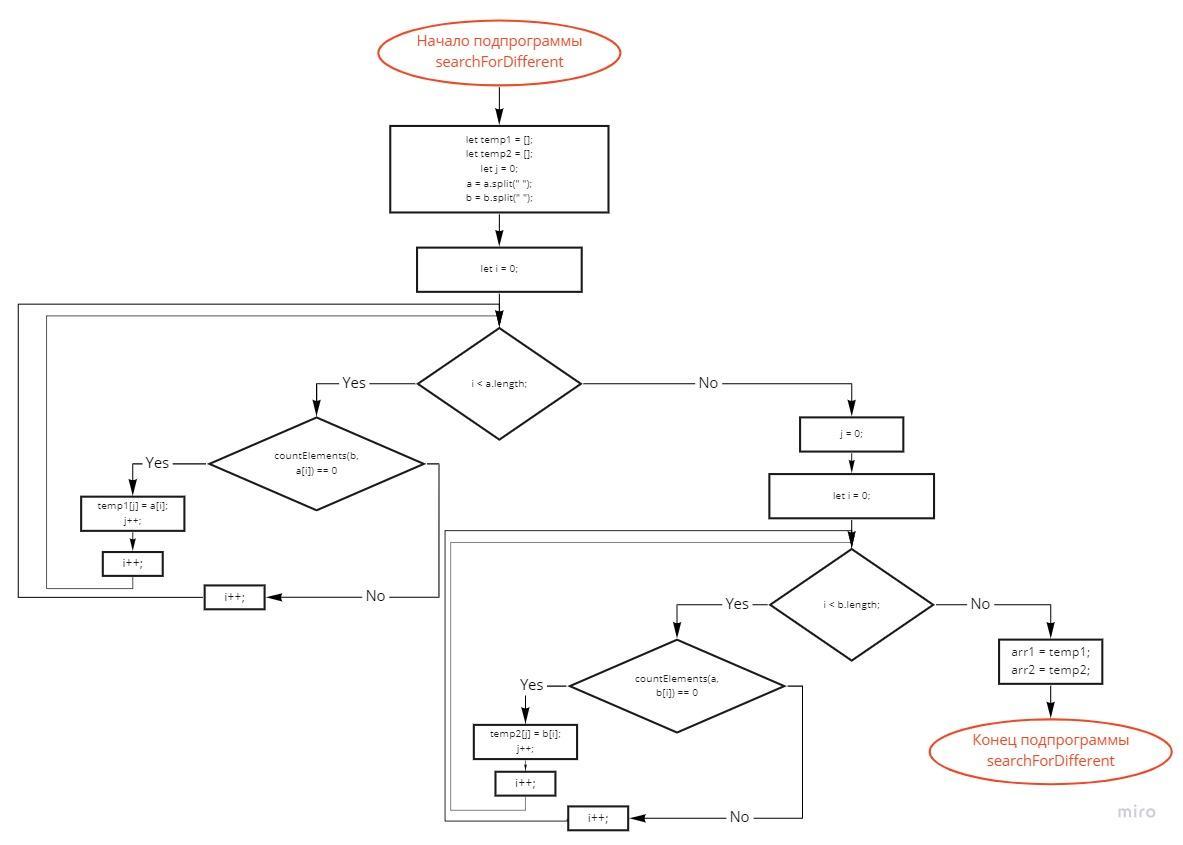
Функция deleteElements удаляет одинаковые элементы в массиве



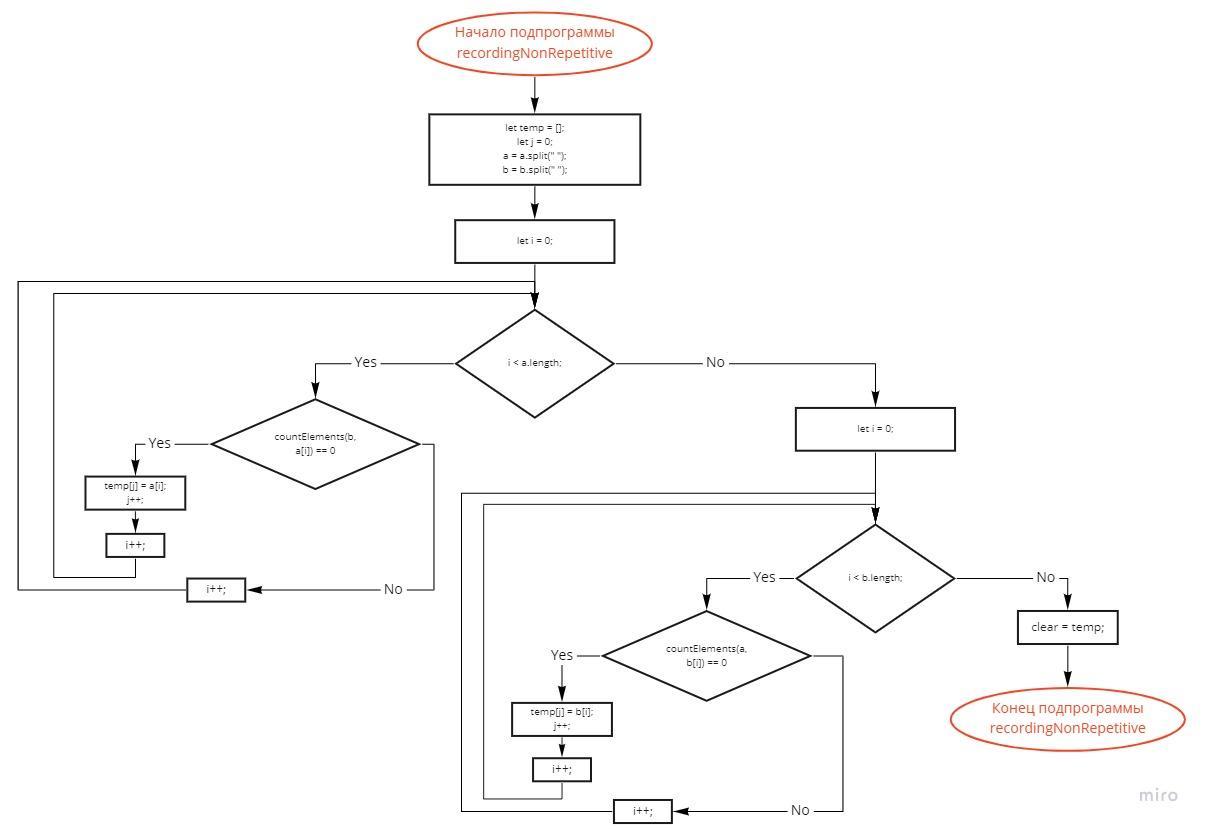
Функция recordingRepetitive ищет элементы, которые повторяются во 2 массиве из элементов 1 массива и записывает повторяющиеся элементы в новый массив



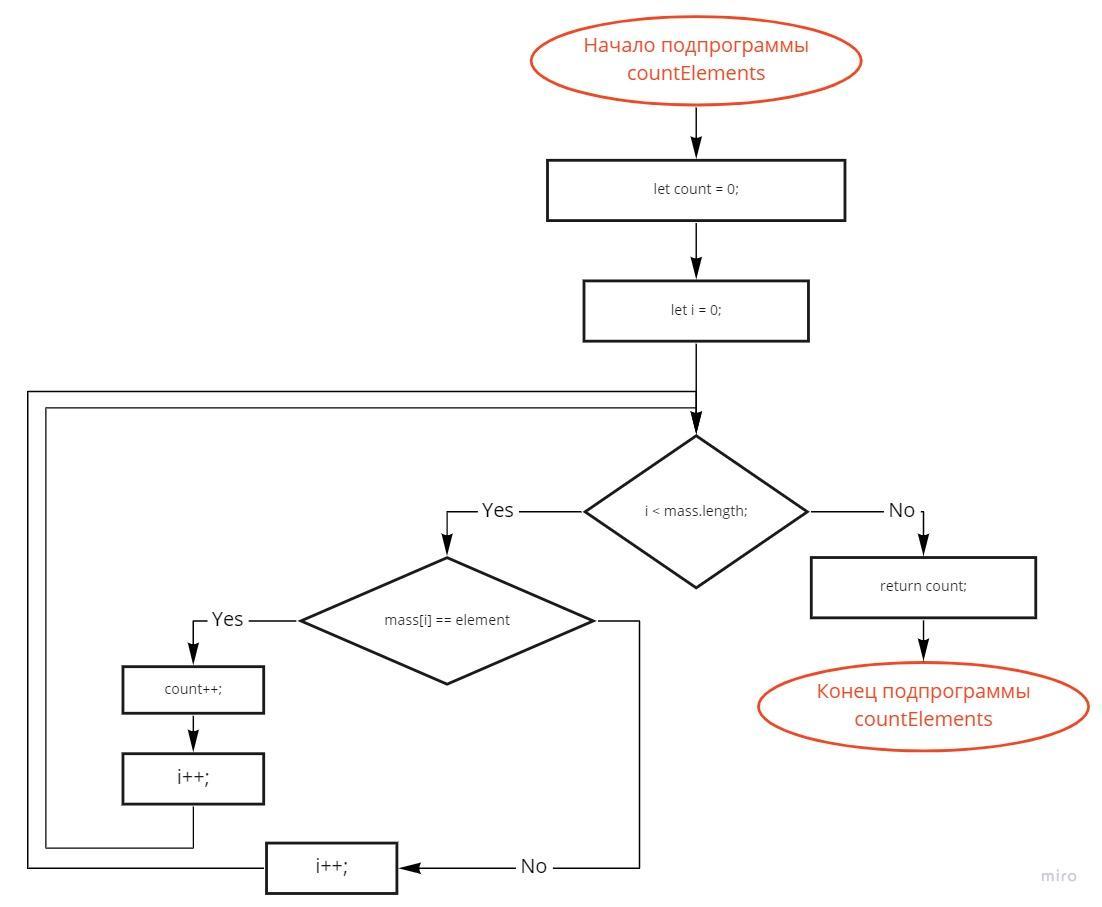
Функция searchForDifferent ищет элементы, которые не повторяются во 2 массиве из элементов 1 массива (и наоборот) и записывает повторяющиеся элементы в новые 2 массива



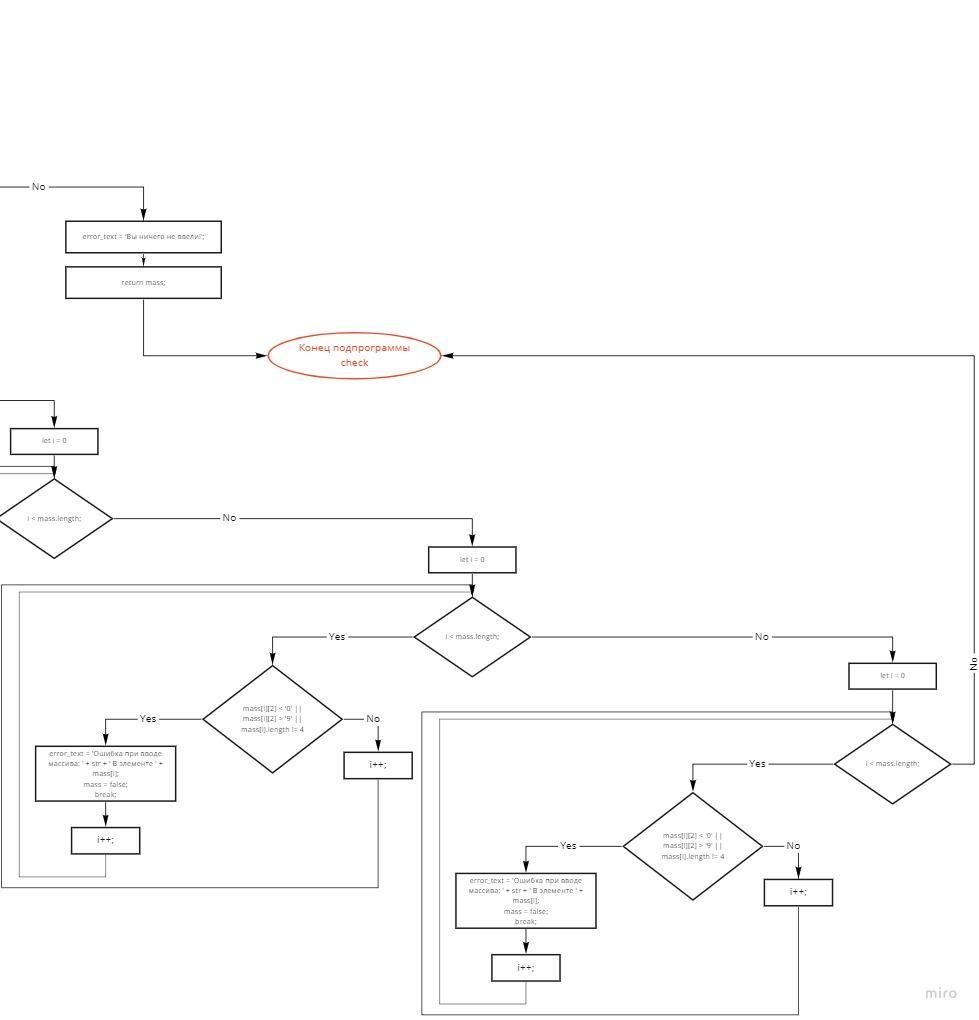
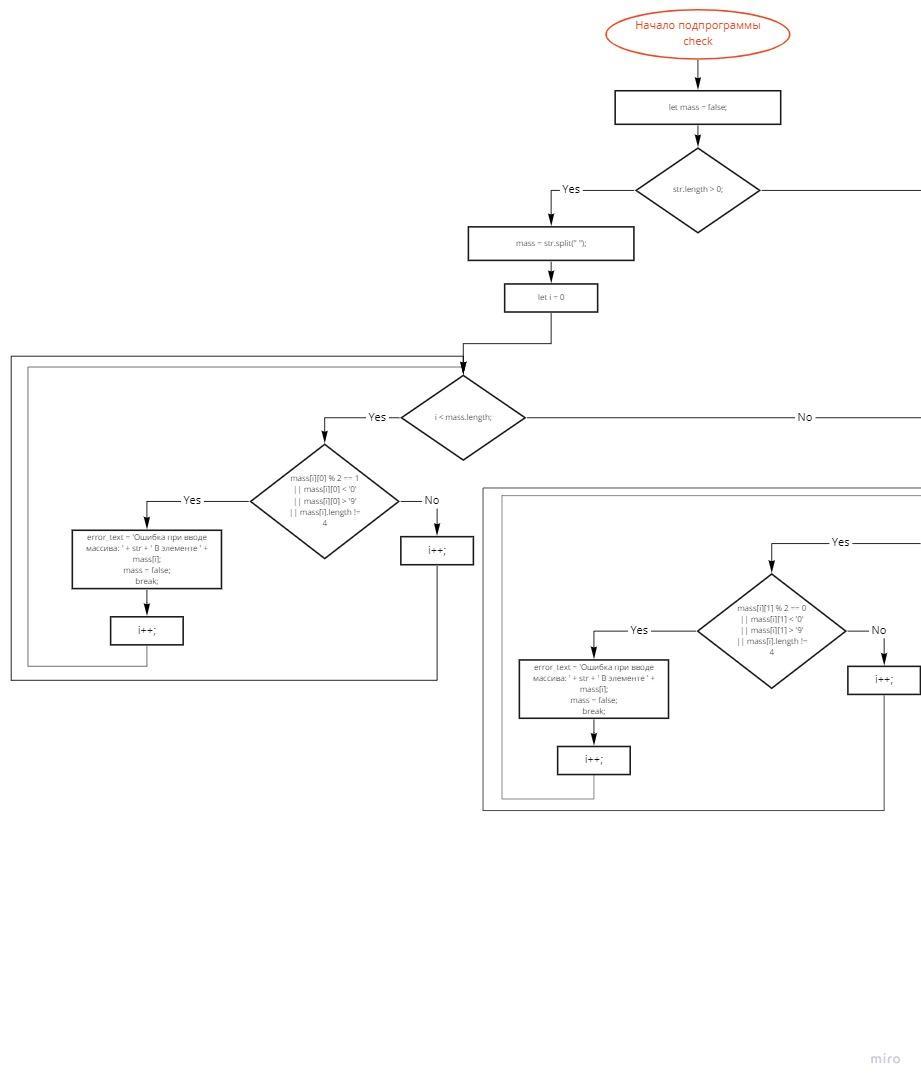
Функция recordingNonRepetitive ищет элементы, которые не повторяются во 2 массиве из элементов 1 массива (и наоборот) и записывает повторяющиеся элементы новый массив



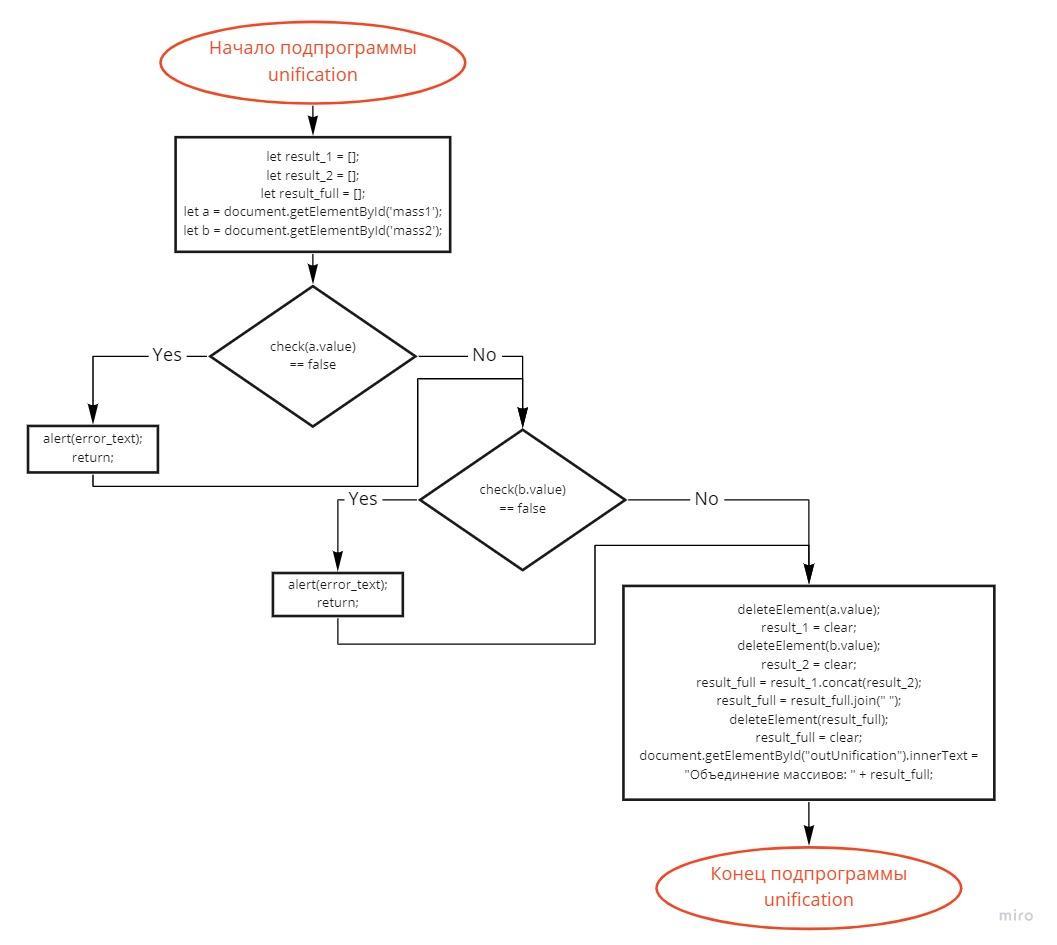
Функция countElements сравнивает элементы массив с заданным элементом и возвращает количество совпадений.



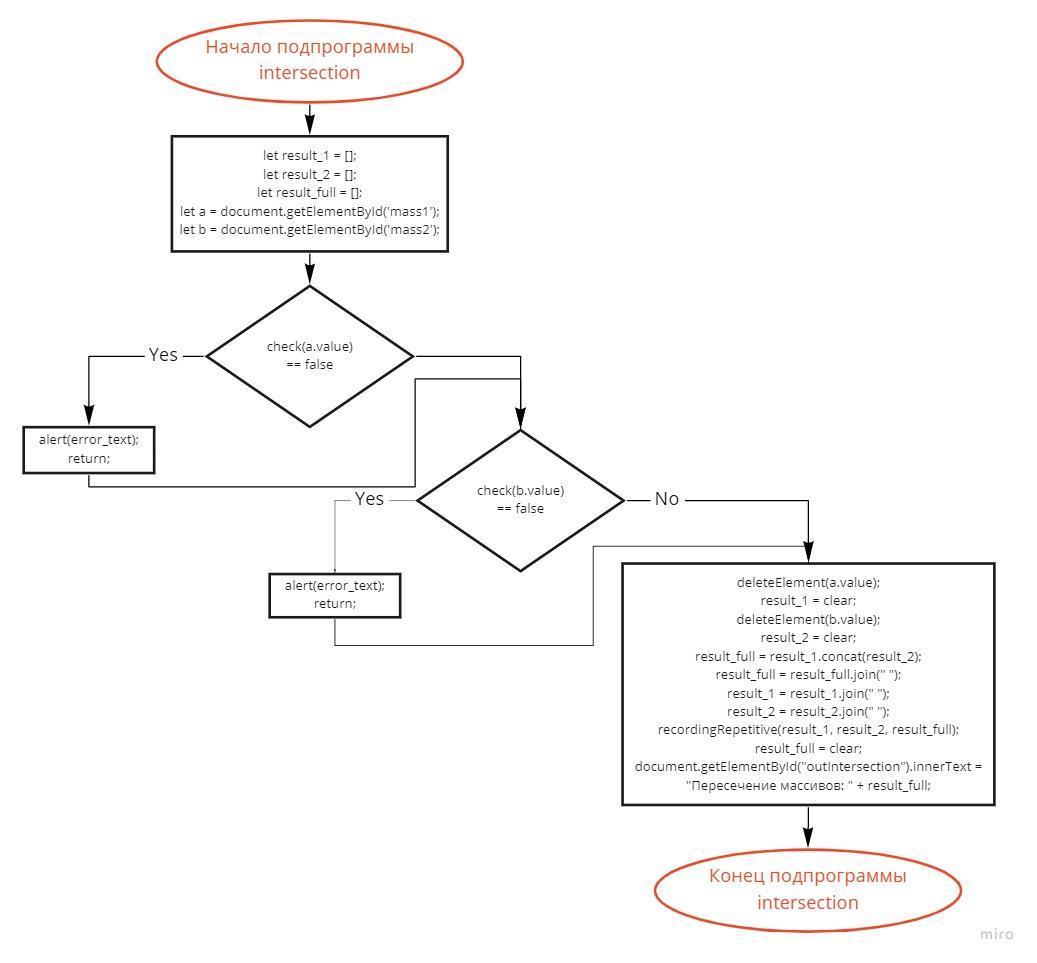
Функция check проверяет 1,2,3,4 символы массива на правильность ввода. А именно, что первый символ - четный, второй - нечётный, третий и четвёртый - цифры.



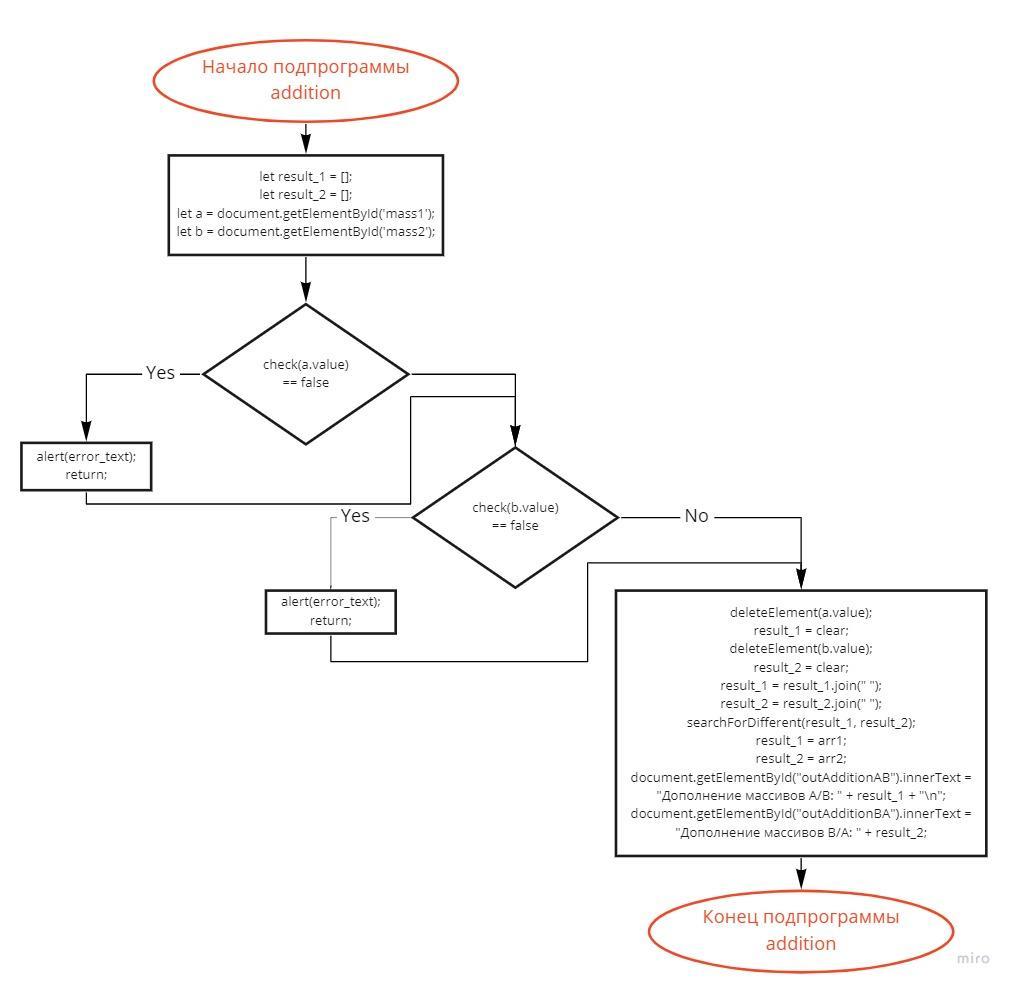
Функция unification выполняет проверку строк на правильность ввода, а также вызывает функции deleteElement, concat, join, alert и выводит результат.



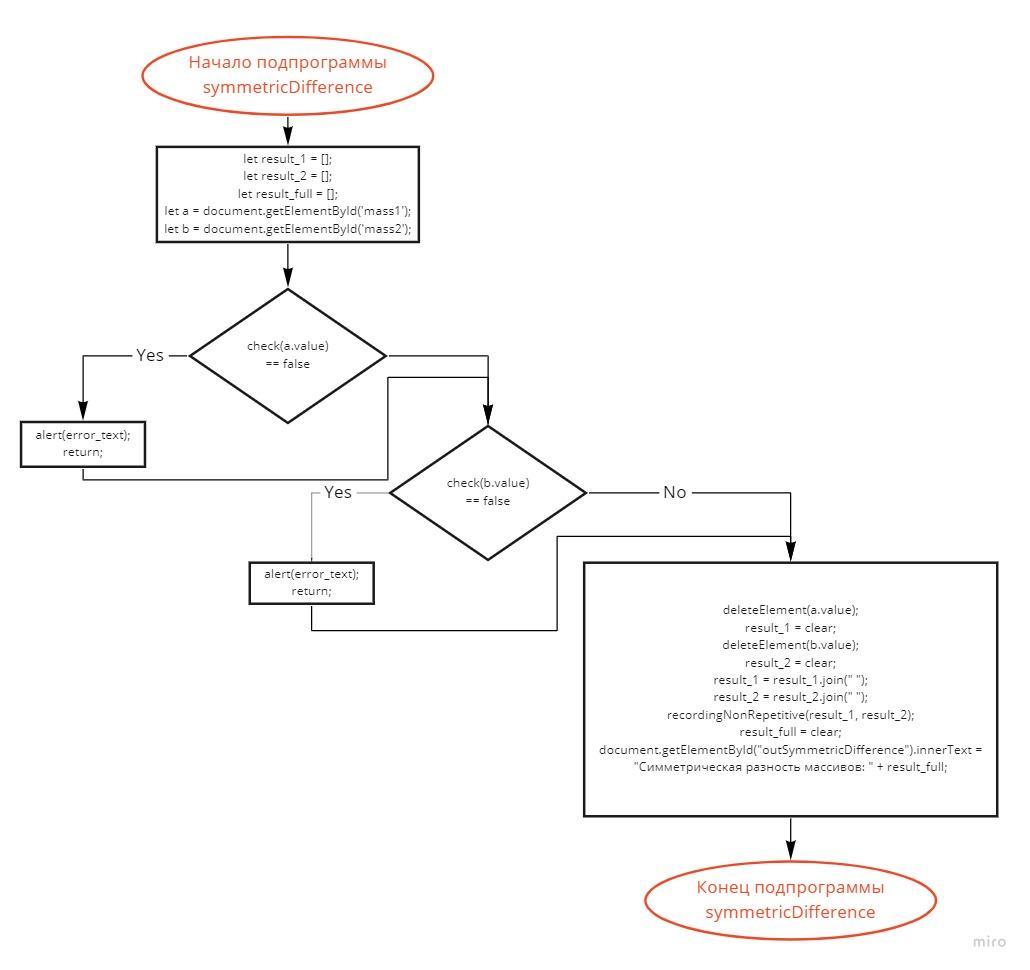
Функция intersection выполняет проверку строк на правильность ввода, а также вызывает функции deleteElement, concat, join, alert, split, recordingRepetitive и выводит результат.



Функция addition выполняет проверку строк на правильность ввода, а также вызывает функции deleteElement, join, alert, searchForDifferent и выводит результат.



Функция symmetricDifference выполняет проверку строк на правильность ввода, а также вызывает функции deleteElement, join, alert, recordingNonRepetitive и выводит результат.



1. **Листинг кода программы**

let clear, arr1, arr2, error\_text;

/\* 1 Функция объединение \*/

function deleteElement(str)

{

let mass = str.split(" ");

//Удаление повторяющихся элементов

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

if (countElements(mass, mass[i]) > 1)

{

mass.splice(i, 1);

i--;

}

clear = mass;

}

/\* 2 Функция пересечение \*/

function recordingRepetitive(a, b, all)

{

let temp = [];

let j = 0;

a = a.split(" ");

b = b.split(" ");

all = all.split(" ");

//Сохранение одинаковых элементов

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

if (countElements(b, a[i]) >= 1)

{

temp[j] = a[i];

j++;

}

clear = temp;

}

/\* 3 Функция дополнение \*/

function searchForDifferent(a, b)

{

let temp1 = [];

let temp2 = [];

let j = 0;

a = a.split(" ");

b = b.split(" ");

//Сохранение одинаковых элементов

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

if (countElements(b, a[i]) == 0)

{

temp1[j] = a[i];

j++;

}

j = 0;

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

if (countElements(a, b[i]) == 0)

{

temp2[j] = b[i];

j++;

}

arr1 = temp1;

arr2 = temp2;

}

/\* 4 Функция симметричесая разность \*/

function recordingNonRepetitive(a, b)

{

let temp = [];

let j = 0;

a = a.split(" ");

b = b.split(" ");

//Сохранение одинаковых элементов

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

if (countElements(b, a[i]) == 0)

{

temp[j] = a[i];

j++;

}

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

if (countElements(a, b[i]) == 0)

{

temp[j] = b[i];

j++;

}

clear = temp;

}

/\* Функция определяет количества вхождений элемента \*/

function countElements(mass, element)

{

let count = 0;

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

if (mass[i] == element)

count++;

return count;

}

/\* Функция проверки элементов массива \*/

function check(str)

{

let mass = false;

if(str.length > 0)

{

mass = str.split(" ");

//Проврка на ввод 1 символа слова

for (let i = 0; i < mass.length; i++) //проверка на четную, если НЕЧЁТНАЯ то ошибка

{

if (mass[i][0] % 2 == 1 || mass[i][0] < '0' || mass[i][0] > '9' || mass[i].length != 4)

{

error\_text = 'Ошибка при вводе массива: ' + str + ' В элементе ' + mass[i];

mass = false;

break;

}

}

for (let i = 0; i < mass.length; i++) //проверка на нечётную, если нечетная то ошибка

{

if (mass[i][1] % 2 == 0 || mass[i][1] < '0' || mass[i][1] > '9' || mass[i].length != 4)

{

error\_text = 'Ошибка при вводе массива: ' + str + ' В элементе ' + mass[i];

mass = false;

break;

}

}

for (let i = 0; i < mass.length; i++) //проверка на цифру

{

if (mass[i][2] < '0' || mass[i][2] > '9' || mass[i].length != 4)

{

error\_text = 'Ошибка при вводе массива: ' + str + ' В элементе ' + mass[i];

mass = false;

break;

}

}

for (let i = 0; i < mass.length; i++) //проверка на цифру

{

if (mass[i][3] < '0' || mass[i][3] > '9' || mass[i].length != 4)

{

error\_text = 'Ошибка при вводе массива: ' + str + ' В элементе ' + mass[i];

mass = false;

break;

}

}

}

else

error\_text = 'Вы ничего не ввели!';

return mass;

}

/\* КЛЮЧЕВЫЕ ФУНКЦИИ\*/

/\* 1 Функция объединения \*/

function unification()

{

let result\_1 = [];

let result\_2 = [];

let result\_full = [];

let a = document.getElementById('mass1');

let b = document.getElementById('mass2');

if (check(a.value) == false)

{

alert(error\_text);

return;

}

if (check(b.value) == false)

{

alert(error\_text);

return;

}

deleteElement(a.value);

result\_1 = clear;

deleteElement(b.value);

result\_2 = clear;

result\_full = result\_1.concat(result\_2);

result\_full = result\_full.join(" ");

deleteElement(result\_full);

result\_full = clear;

document.getElementById("outUnification").innerText = "Объединение массивов: " + result\_full;

}

/\* 2 Функция пересечения \*/

function intersection()

{

let result\_1 = [];

let result\_2 = [];

let result\_full = [];

let a = document.getElementById('mass1');

let b = document.getElementById('mass2');

if (check(a.value) == false)

{

alert(error\_text);

return;

}

if (check(b.value) == false)

{

alert(error\_text);

return;

}

//result\_1 = a.value.split(" ");

deleteElement(a.value);

result\_1 = clear;

deleteElement(b.value);

result\_2 = clear;

result\_full = result\_1.concat(result\_2);

result\_full = result\_full.join(" ");

result\_1 = result\_1.join(" ");

result\_2 = result\_2.join(" ");

recordingRepetitive(result\_1, result\_2, result\_full);

result\_full = clear;

document.getElementById("outIntersection").innerText = "Пересечение массивов: " + result\_full;

}

/\* 3 Функция дополнения \*/

function addition()

{

let result\_1 = [];

let result\_2 = [];

let a = document.getElementById('mass1');

let b = document.getElementById('mass2');

if (check(a.value) == false)

{

alert(error\_text);

return;

}

if (check(b.value) == false)

{

alert(error\_text);

return;

}

deleteElement(a.value);

result\_1 = clear;

deleteElement(b.value);

result\_2 = clear;

result\_1 = result\_1.join(" ");

result\_2 = result\_2.join(" ");

searchForDifferent(result\_1, result\_2);

result\_1 = arr1;

result\_2 = arr2;

document.getElementById("outAdditionAB").innerText = "Дополнение массивов A/B: " + result\_1 + "\n";

document.getElementById("outAdditionBA").innerText = "Дополнение массивов B/A: " + result\_2;

}

/\* 4 Функция симметрической разности \*/

function symmetricDifference()

{

let result\_1 = [];

let result\_2 = [];

let result\_full = [];

let a = document.getElementById('mass1');

let b = document.getElementById('mass2');

if (check(a.value) == false)

{

alert(error\_text);

return;

}

if (check(b.value) == false)

{

alert(error\_text);

return;

}

deleteElement(a.value);

result\_1 = clear;

deleteElement(b.value);

result\_2 = clear;

result\_1 = result\_1.join(" ");

result\_2 = result\_2.join(" ");

recordingNonRepetitive(result\_1, result\_2);

result\_full = clear;

document.getElementById("outSymmetricDifference").innerText = "Симметрическая разность массивов: " + result\_full;

}

1. **Выводы**

Программа работает корректно. Верно выполняет операции над множествами (ак пересечение, объединение, симметрическая разность, дополнение A/B и B/A) с проверкой на корректность ввода данных.

1. **Список литературы**
2. <https://youtu.be/Bluxbh9CaQ0> – JavaScript Основы для Начинающих
3. <https://learn.javascript.ru/> – Современный учебник JavaScript