Министерство образования и науки Российской Федерации ФБГОУ ВО Ульяновский государственный технически университет

Кафедра «вычислительная техника»

**Лабораторная работа №1**

«Операции над множествами»

Выполнила студентка

группы ИВТАСбд-11

Денисова Д.В.

Ульяновск, 2022

**Оглавление.**

1. Цель работы……………………………………………………………………2
2. Описание метода выполнения………………………………………………..2
3. Блок-схема алгоритма…………………………………………………….......2
4. Вывод…………………………………………………………………………11
5. Список использованной литературы………………………………………..11
6. **Цель работы.**

Необходимо написать программу, которая будет проводить операции над множествами, а именно нахождение: их пересечения, объединения, симметрической разницы, дополнение первого до второго и наоборот. Также должна присутствовать проверка на корректность введённых данных. Первый символ элемента – любая цифра, второй – чётная цифра, третий – нечётная цифра, четвертый – буква. Пользователь имеет возможность ввода данных.

1. **Описание метода выполнения.**

Множество – совокупность определенных объектов. Операции над ними – манипуляция над этими объектами. Операции над множествами: объединение, пересечение, симметрическая разница, дополнение первого до второго и наоборот. Объединение – множество, содержащее в себе все элементы других множеств без повторения. Пересечение – множество, содержащее в себе только элементы первого и второго множеств. Симметрическая разница – множества, содержащее в себе элементы первого и второго множеств, которые не совпадают между собой. Дополнением первого до второго называется множество, содержащее все элементы первого множества, которые не принадлежат второму множеству и наоборот. Для того, чтобы пользователь мог вводить данные необходимо написать небольшую html-структуру, состоящую из двух полей для ввода, полей для вывода и кнопку, по которой произойдут операции над множествами. Код проводимых операций будет написан на JavaScript. Мы получаем наши массивы в html-структуру на PHP и по id передаём в JavaScript.

1. **Блок-схема алгоритма**.

Для проверки на корректность ввода данных пользователем, я пробегаюсь по нашим массивам с определенными условиями и если ввод элементов массива некорректен, то у пользователя высвечивается предупреждение, в котором написано в каком массиве и в каком элементе был допущена ошибка ввода.

Вот так выглядит функция проверки на корректность введенных элементов:

mass=false;

if(str.length){

mass=str.split(" ");

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

if(mass[i][0]>'9' || mass[i][0]<'0' || mass[i][1]%2!=0 || mass[i][2]%2==0 || mass[i][3]<'a' || mass[i][3]>'z' || 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++){

if(uniqueness(mass, mass[i])>1){

mass.splice(i, 1);

i--;

}

}

}

else {

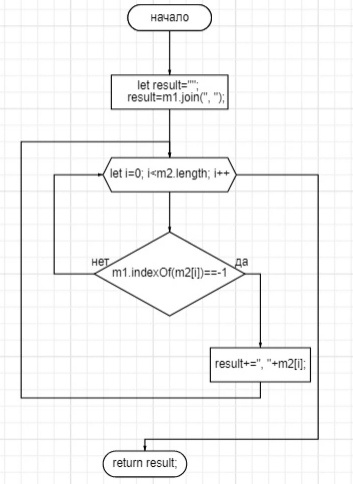
error\_text="Вы не ввели массив"

}

В первом цикле for мы проверяем является ли первый символ элемента – любой цифрой, второй – чётной цифрой, третий – нечётно цифрой, четвертый – буквой. Если условие не выполняется, то мы указываем пользователю на ошибку (если массив не введен или если не соблюден формат ввода)

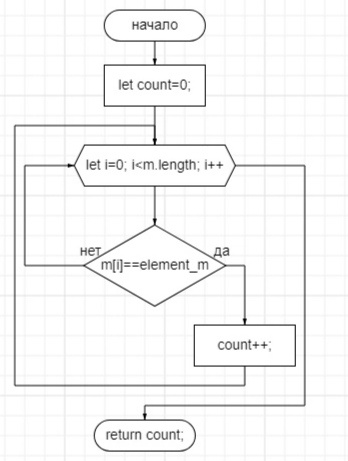
Во втором цикле for мы удаляем повторяющиеся элементы с помощью функции splice

Блок-схема функции для объединения множеств:



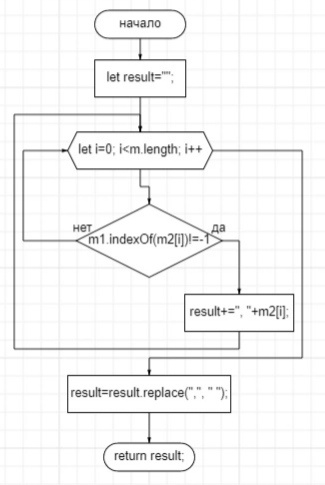
В этой функции мы берем каждый элемент второго массива и проверяем, есть ли такой же элемент в первом массиве, если нет, то добавляем этот элемент к строке, в которую предварительно положили первый массив.

Также для удобства написали функцию для поиска одинаковых элементов массива, блок-схема этой функции:



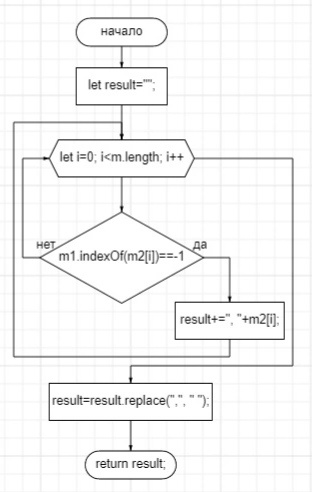
Мы передаём в функцию массив и элемент, подсчитываем сколько раз данный элемент встречается в массиве и возвращаем полученное значение. Эта функция понадобилась нам для написания всех остальных функций для операций над множествами.

Функция для нахождения пересечения множеств и ее блок-схема:



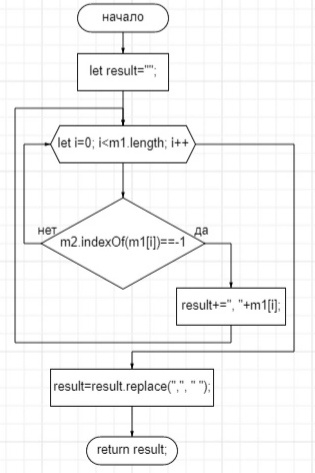
В этой функции мы ищем одинаковые элементы у первого и второго массива, собираем эти элементы в строку и возвращаем ее, предварительно удалив первую запятую для красивого вывода.

Функция для нахождения дополнения первого множества до второго и ее блок схема:



В этой функции мы находим те элементы второго массива, которых нет у первого, также добавляем их к строке, удаляем первую запятую и возвращаем эту строку.

Функция для нахождения дополнения второго множества до первого и ее блок схема:



В этой функции мы находим те элементы первого массива, которых нет у второго, также добавляем их к строке, удаляем первую запятую и возвращаем эту строку.

Далее о функции calculation, в которой мы собираем конечный результат:

function calculation() {

let result="";

var m1=document.getElementById('mass1')

var m2=document.getElementById('mass2')

if((mass\_1=checkmass(m1.value))==false){

alert(error\_text);

}

if((mass\_2=checkmass(m2.value))==false){

alert(error\_text);

}

if((mass\_1=checkmass(m1.value))!=false && (mass\_2=checkmass(m2.value))!=false){

result='Объединение массивов: ' + unification(mass\_1, mass\_2)+'\n';

result+='Пересечение массивов: ' + intersection(mass\_1, mass\_2)+'\n';

result+='Дополнение массива A до массива B: ' + addition\_AB(mass\_1, mass\_2)+'\n';

result+='Дополнение массива B до массива A: ' + addition\_BA(mass\_1, mass\_2)+'\n';

result+='Симметрическая разность масивов A и B: ' + addition\_BA(mass\_1, mass\_2)+','+addition\_AB(mass\_1, mass\_2)+'\n';

}

document.getElementById('outresult').innerText='Результат выполнения операций над массивами:' + '\n' + result;

}

В этой функции мы сначала вызываем функцию для проверки на корректность ввода массивов, если они выведены некорректно, то выводим на экран пользователю нашу переменную, в которой хранится информация об ошибке. Затем вызываем наши функции, собираем весь результат операций в одну строку и выводим ее. А симметрическая разность получилась в результате сложения результата от дополнения первого до второго и наоборот.

1. **Вывод.**

Программа протестирована, работает успешно. Она, работая с множествами, находит их пересечение, объединение, симметрическую разность, дополнение первого до второго и наоборот.

1. **Список использованной литературы.**

<https://studme.org/119335/matematika_himiya_fizik/operatsii_mnozhestvami> - операции над множествами.

<https://learn.javascript.ru/> - Современный учебник JavaScript