Министерство образования и науки Российской Федерации

ФГБОУ ВО Ульяновский государственный технический университет

Кафедра «Вычислительная техника»

Лабораторная работа №5

«Матрица достижимости»

Выполнил студент

группы ИВТАСбд-11

Зюзин Г.А

Проверил Игонин А.Г.

Ульяновск, 2022 г.

# Оглавление.

[Оглавление. 2](#_Toc1)

[Цель работы. 3](#_Toc2)

[Метод выполнения.](#_Toc3)

[Листинг кода. 8](#_Toc4)

[Выводы. 10](#_Toc5)

[Список литературы.](#_Toc6)

# Цель работы.

Разработать программу на языке программирования PHP нахождения матрицы достижимости с использованием AJAX запросами на языке программирования Java Script.

# Метод выполнения.

Матрица достижимости - это квадратная матрица W из n строк и n столбцов, где n - количество вершин графа, в которой Wij=1, если в графе существует путь от вершины i до вершины j, и Wij = 0 в противном случае. То есть матрица показывается существует ли путь между вершинами.

Для того, чтобы вычислить эту матрицу на вход нужна матрица смежности. Матрица смежности (Рис. 1) – это квадратная логическая матрица отношения на множестве вершин графа, которое задаётся его рёбрами, является одним из способов представить граф. Таблица имеет два значение 1 и 0 (или И и Л, соответственно), которые обозначают отношения между вершинами графа.

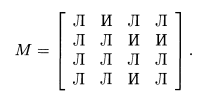


Рис. 1

Для нахождения матрицы достижимости существует несколько способов. Например, возведение матрицы в степень n (M^n) (Рис. 2).

Есть смежная матрица, которая возводится в степень определённое количество раз.

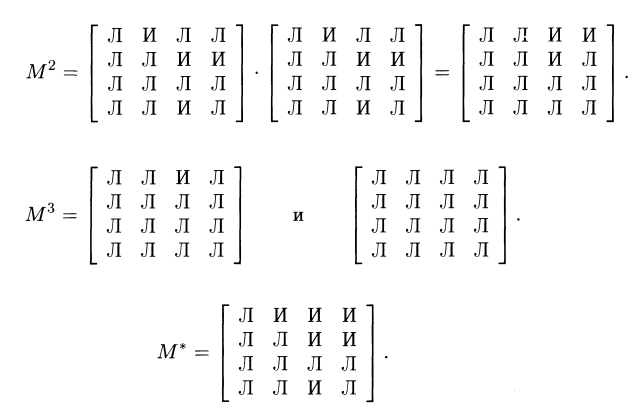


Рис. 2

М\* – матрица достижимости

Но данный способ утомительный и неэффективный. Поэтому данная программа использует алгоритм Уоршелла (Рис. 3, Рис. 4, Листинг 1).

Этот алгоритм генерирует последовательность матриц W0 = M, W1, W2, ..., Wn.

Рис. 3

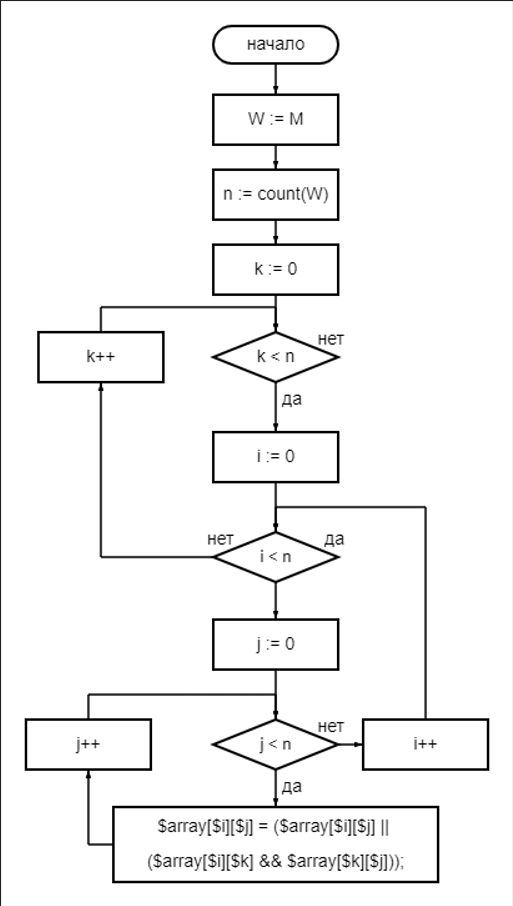
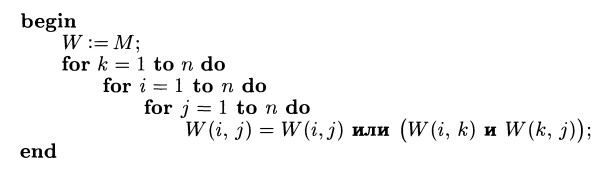


Рис. 4 – Блок-схема алгоритма Уоршелла

Wn – это искомая матрица достижимости. За каждый проход цикла алгоритм Уоршелла создаёт новую матрицу Wk, используя и заменяя элементы матрицы Wk-1.

Листинг 1

for ($k = 0; $k < count($column\_row\_matrix); $k++)

for ($i = 0; $i < count($column\_row\_matrix); $i++)

for ($j = 0; $j < count($column\_row\_matrix); $j++)

$array[$i][$j] = ($array[$i][$j] || ($array[$i][$k] && $array[$k][$j]));

for ($i = 0; $i < count($column\_row\_matrix); $i++)

$array[$i][$i] = 1;

for ($i = 0; $i < count($column\_row\_matrix); $i++) {

for ($j = 0; $j < count($column\_row\_matrix); $j++) {

if ($array[$i][$j])

echo "1";

else

echo "0";

echo " ";

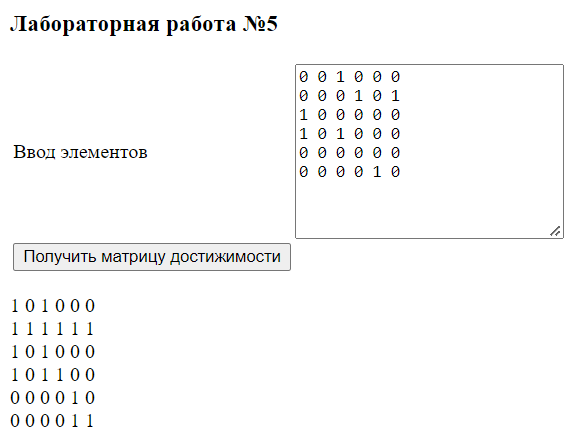
}

echo "<br>";

}

Программа выглядит следующим образом: одностарничный сайт с полем ввода для матрицы и кнопка расчёта матрицы достижимости (Рис. 4).

Рис. 5



С помощью AJAX запроса (Листинг 2) матрица инициализируется в коде языка программирования PHP (data: `matrix=${matrix}`). В программе реализованы проверки на корректность ввода, на наличие вершины.

Листинг 2

$.ajax({

type: "POST",

url: '/scripts/script4.php',

data: `matrix=${matrix}`,

success: function (data){

document.getElementById('output').innerHTML = data;

},

error: function (){

alert("Ошибка передачи данных");

}

});

На вход программа получает текст, поэтому необходимо инициализировать матрицу из этого текста. Текст разбить на строки, а строки разбить на элементы (Листинг 3).

Листинг 3

for($i = 0; $i < count($row\_matrix);$i++){

$column\_row\_matrix = explode(' ', trim($row\_matrix[$i]));

$array\_row = [];

for($j = 0; $j < count($column\_row\_matrix); $j++){

if(count($column\_row\_matrix) == 1 && $column\_row\_matrix[$j] == '')

die("Ошибка: лишние переносы!");

array\_push($array\_row, $column\_row\_matrix[$j]);

}

array\_push($array, $array\_row);

}

# Листинг кода.

AJAX запрос (ajax\_script5.js):

function sendData(){

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

$.ajax({

type: "POST",

url: '/scripts/script5.php',

data: `matrix=${matrix}`,

success: function (data){

document.getElementById('output').innerHTML = data;

},

error: function (){

alert("Ошибка передачи данных");

}

});

}

Алгоритм нахождения матрицы достижимости(script4.php):

<?php

$matrix = $\_POST['matrix'];

$row\_matrix = explode("\n", $matrix);

$array = []; //Матрица

for($i = 0; $i < count($row\_matrix);$i++){

$column\_row\_matrix = explode(' ', trim($row\_matrix[$i]));

$array\_row = [];

for($j = 0; $j < count($column\_row\_matrix); $j++){

if(count($column\_row\_matrix) == 1 && $column\_row\_matrix[$j] == '')

die("Ошибка: лишние переносы!");

array\_push($array\_row, $column\_row\_matrix[$j]);

}

array\_push($array, $array\_row);

}

for($i = 0; $i < count($array); $i++){

for($j = 0; $j < count($array[$i]); $j++){

if(count($array) != count($array[$i])){

die("Ошибка: матрица должна быть квадратной!");

}

}

}

for($i = 0; $i < count($array); $i++){

for($j = 0; $j < count($array[$i]); $j++){

if($array[$i][$j] != 0 && $array[$i][$j] != 1){

die("Ошибка: матрица должна состоять из 0 и 1!");

}

}

}

for ($k = 0; $k < count($column\_row\_matrix); $k++)

for ($i = 0; $i < count($column\_row\_matrix); $i++)

for ($j = 0; $j < count($column\_row\_matrix); $j++)

$array[$i][$j] = ($array[$i][$j] || ($array[$i][$k] && $array[$k][$j]));

for ($i = 0; $i < count($column\_row\_matrix); $i++)

$array[$i][$i] = 1;

for ($i = 0; $i < count($column\_row\_matrix); $i++) {

for ($j = 0; $j < count($column\_row\_matrix); $j++) {

if ($array[$i][$j])

echo "1";

else

echo "0";

echo " ";

}

echo "<br>";

}

?>

# Выводы.

Во время выполнения лабораторной работы, были изучены матрица достижимости, её свойства и способы нахождения. Разработана программа для нахождения матрицы достижимости. Для разработки был использован язык программирования PHP и JS, а именно AJAX запросы.

# Список литературы.

1. Матрица достижимости. Алгоритм Уоршела. - <https://studfile.net/preview/3638408/page:39/>
2. Р. Хаггарти, Дискретная математика для программистов / Техносфера, г. Москва, 2005 г. - 400с.
3. Справочник по JavaScript - <https://javascript.ru/manual>
4. Справочник по PHP - <https://www.php.net/manual/ru/langref.php>