Лабораторная работа

**Тема:** Компьютерное моделирование на основе решений систем линейных уравнений

**Постановка задачи:** реализовать Метод выбора

**Язык программирования:** JavaScript

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Имя переменной | Тип | Значение |
| matrix, a | Array | Исходная матрица |
| gaus | Function | Функция, реализующая метод |
| maximum | Function | Функция, находящая максимальный элемент в строке |
| swapper | Function | Функция для того чтобы поменять строки местами по максимальному элементу |
| n | Number | Количество строк |
| i, k, j | Number | Счетчики для обращения к элементам матрицы |
| aki | Number | Вспомогательная перменная |
| x | Array | Массив с результатом |
| sol | Array | Результат работы функции |

**Код программы:**

//Метод выбора

const matrix = [[5, 7, 6, 5, 23],

                [7, 10, 8, 7, 32],

                [6, 8, 10, 9, 33],

                [5, 7, 9, 10, 31]];

// Максимальный элемент в строке. k - номер строки

const maximum = (matrix, k) => {

    const n = matrix.length;

    let max\_el = 0;

    for (let i = k; i < n; ++i){

        if (matrix[k][i] > max\_el) {

            max\_el = matrix[k][i];

            max\_indx = i;

        }

    }

    res = [max\_el, max\_indx];

    return res;

}

// Функция для того чтобы поменять строки местами по максимвальному элементу.

const swapper = (matrix, k, max\_indx) => {

    const n = matrix.length-1;

    for (let i = 0; i < n; ++i) {

        tmp\_el = matrix[i][k];

        matrix[i][k] = matrix[i][max\_indx];

        matrix[i][max\_indx] = tmp\_el;

    }

    tmp\_indx = matrix[n][k];

    matrix[n][k] = matrix[n][max\_indx]

    matrix[n][max\_indx] = tmp\_indx

    return matrix

}

const gaus = (a) => {

    const n = a.length;

    let q;

    for (let k = 0; k < n-1; ++k){

        max = maximum(a, k)

        m = swapper(a, k, max[1])

    }

    for (let i = 0; i < n; ++i){

        for (let k = i+1; k < n; ++k){

            let aki = a[k][i]/a[i][i];

            for (let j = i; j < n + 1; ++j){

                a[k][j] -= aki \* a[i][j]

            };

        };

    };

    const x = new Array(n);

    for (let i = n-1; i >= 0; --i){

        x[i] = a[i][n] / a[i][i];

        for (let c = n - 1; c > i; c = c - 1){

            x[i] = x[i] - a[i][c]\*x[c] / a[i][i];

        };

    };

    return x;

}

const sol = gaus(matrix);

console.log(`Solution 1: ${sol}`);

**Результат работы программы:**

