ЗВІТ

про виконання лабораторної роботи № 1

Логіка

з дисципліни «Програмування дискретних структур»

студента групи КН-2226Б

Медвідь Дмитро Степанович

Завдання №1

1. Умова завдання;

Задано значення істинності висловлювань *p* та *q*. Знайти значення істинності кон’юнкції, диз’юнкції, альтернативного «або», імплікації й еквівалентності цих висловлювань.

1. опис алгоритму (блок-схема або структурна схема);
2. текст програми;

// logic.js

function conjunction(p, q) {

    return p && q;

}

function disjunction(p, q) {

    return p || q;

}

function exclusiveOr(p, q) {

    return p !== q;

}

function implicationPtoQ(p, q) {

    return !p || q;

}

function implicationQtoP(p, q) {

    return !q || p;

}

function equivalence(p, q) {

    return (!p || q) && (!q || p);

}

// Ввід значень p та q

const p = true;

const q = false;

// Обчислення та виведення результатів

console.log("p:", p);

console.log("q:", q);

console.log("Conjunction (p && q):", conjunction(p, q));

console.log("Disjunction (p || q):", disjunction(p, q));

console.log("Exclusive Or (p XOR q):", exclusiveOr(p, q));

console.log("Implication (p -> q):", implicationPtoQ(p, q));

console.log("Implication (q -> p):", implicationQtoP(p, q));

console.log("Equivalence (p ~ q):", equivalence(p, q));

module.exports = {

    conjunction,

    disjunction,

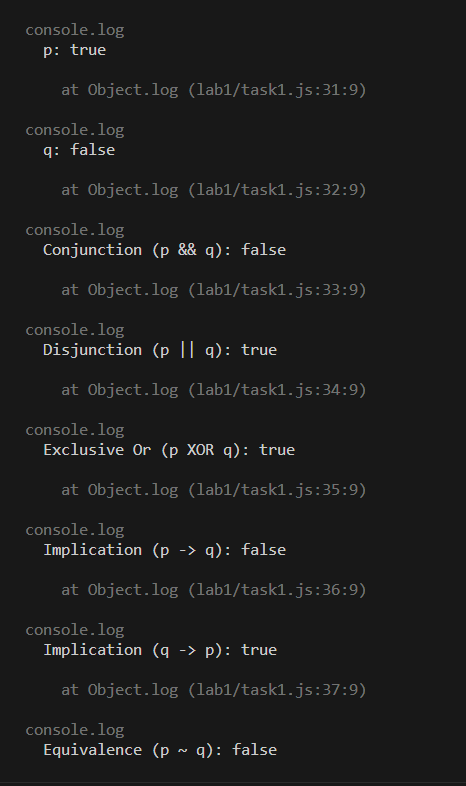
    exclusiveOr,

    implicationPtoQ,

    implicationQtoP,

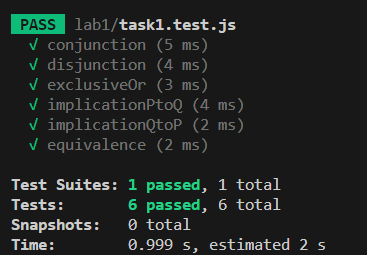
    equivalence,

};

1. значення вхідних даних та результати виконання програми; 
2. посилання на публічний репозиторій GitHub з кодом програми;

https://github.com/DartVeyder/Discrete-programming/blob/main/lab1/task1.js

1. результати unit-тесту



1. Висновок

За допомогою цих операторів можна обчислити значення істинності виразів, які будуть визначати результат відповідних логічних операцій над висловлюваннями p та q.

Завдання №2

1. умова завдання;

Задано два бітові рядки довжиною *n*. Знайти результати виконання порозрядних операцій OR, AND, XOR цих рядків.

1. опис алгоритму (блок-схема або структурна схема);
2. текст програми;

function processBitStrings(a, b, n) {

    let x = "",

        y = "",

        z = "";

    for (let i = 0; i < n; i++) {

        // Знайти c := a[i] OR b[i]

        let c = (a[i] === "1" || b[i] === "1") ? "1" : "0";

        x += c;

        // Знайти c := a[i] AND b[i]

        c = (a[i] === "1" && b[i] === "1") ? "1" : "0";

        y += c;

        // Знайти c := a[i] XOR b[i]

        c = (a[i] !== b[i]) ? "1" : "0";

        z += c;

    }

    return { x, y, z };

}

// Приклад використання

const a = "11100010";

const b = "10101010";

const n = 8;

const result = processBitStrings(a, b, n);

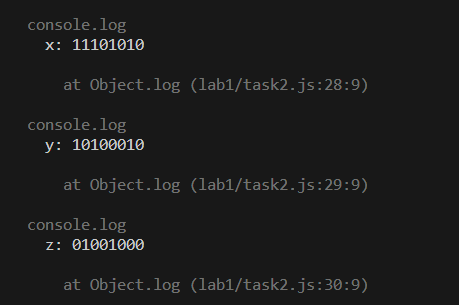
console.log("x:", result.x);

console.log("y:", result.y);

console.log("z:", result.z);

module.exports = processBitStrings;

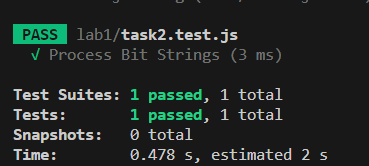
1. значення вхідних даних та результати виконання програми;



1. посилання на публічний репозиторій GitHub з кодом програми;

https://github.com/DartVeyder/Discrete-programming/blob/main/lab1/task2.js

1. результати unit-тесту



1. Висновок

Розробив програму для виконання порозрядних операцій OR, AND, XOR цих рядків.