ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ №2

з дисципліни «Основи програмування на мові Javascript»   
студента групи ПЗ-22у-1

Полтавця Микити Дмитровича

Варіант 11

кафедра комп’ютерних технологій, ДНУ

2024/2025

Постановка задачі

Написати програми на мові JavaScript для Node.js, що ведуть діалог з користувачем в консолі ОС. Програма питає користувача про значення вхідних даних, розв’язує поставлену задачу і виводить результати на консоль ОС.

* Завдання один. На площині задано область. Напишіть програму, яка пропонує користувачеві "влучити" точкою в цю область, пропонуючи декілька спроб. Програма супроводжує спроби користувача коментарями: "тепліше", "холодніше", "нарешті ви влучили з N-ї спроби". Область на площині утворено за допомогою трьох заданих фігур таким чином: об'єднання перших двох фігур (фігури-1 і фігури-2) у перетині з третьою фігурою (фігурою-3).
* Завдання два. Задано послідовність цілих чисел. Напишіть програму, яка за один (за можливості) перегляд послідовності, для елементів, що задовольняють заданому критерію, обчислює задану величину. Критерій і шукана величина визначені індивідуальним варіантом. Не використовуйте масиви, якщо в цьому немає потреби.
* Завдання три. Напишіть програму, яка вводить послідовність цілих чисел a[0], a[1], ... , a[N-1] і упорядковує її згідно до варіанту завдання певним методом за певним критерієм у певному напрямку. Передбачте можливість файлового вводу і виводу даних за допомогою перенапрямлення вводу і виводу в командному рядку. Метод упорядкування реалізуйте у вигляді окремої функції (стандартні функції упорядкування використовувати не дозволяється).

1. Опис розв’язку

Для виконання лабораторної роботи, створив три файли .js та вніс туди код програм трьох лабораторних робіт. Протестував код у консолі. Зберіг файл до репозиторія на GitHub.

1. Вихідний текст програми розв’язку задачі (основні фрагменти з коментарями)

Лістинг 1.1 – Код програми першого завдання з коментарями

const readline = require('readline');

const strip = { x1: 1, x2: 3 };

const circle1 = { xc: 2, yc: 2, r: 2 };

const circle2 = { xc: 3, yc: 3, r: 3 };

let prevDistance = null;

let attempt = 0;

const rl = readline.createInterface({

input: process.stdin,

output: process.stdout

});

*console*.log("🎯 Спробуйте влучити точкою у фігуру: (стрічка ∪ коло-1) ∩ коло-2");

function askPoint() {

rl.question(`Спроба #${attempt + 1}. Введіть координати точки (x y): `, (input) => {

attempt++;

const [x, y] = input.split(' ').map(Number);

const inStrip = x >= strip.x1 && x <= strip.x2;

const inCircle1 = *Math*.pow(x - circle1.xc, 2) + *Math*.pow(y - circle1.yc, 2) <= *Math*.pow(circle1.r, 2);

const inCircle2 = *Math*.pow(x - circle2.xc, 2) + *Math*.pow(y - circle2.yc, 2) <= *Math*.pow(circle2.r, 2);

const inUnion = inStrip || inCircle1;

const inTarget = inUnion && inCircle2;

if (inTarget) {

*console*.log(`🎉 Нарешті ви влучили з ${attempt}-ї спроби!`);

rl.close();

} else {

const distToCenter = *Math*.sqrt(*Math*.pow(x - circle2.xc, 2) + *Math*.pow(y - circle2.yc, 2));

if (prevDistance === null) {

*console*.log("❌ Не влучили. Спробуйте ще.");

} else if (distToCenter < prevDistance) {

*console*.log("🔥 Тепліше!");

} else {

*console*.log("❄️ Холодніше!");

}

prevDistance = distToCenter;

askPoint();

}

});

}

askPoint();

Лістинг 1.2 – Код програми другого завдання з коментарями

const readline = require('readline');

const rl = readline.createInterface({

input: process.stdin,

output: process.stdout

});

*console*.log("Вводьте цілі числа через пробіл (наприклад: 45 -891 82 23 944 100).");

rl.question("Введіть послідовність чисел: ", (input) => {

const tokens = input.split(' ').map(Number);

let max = -*Infinity*;

let secondMax = -*Infinity*;

for (let num of tokens) {

const firstDigit = parseInt(*Math*.abs(num).toString()[0]);

if (firstDigit !== 9) {

if (num > max) {

secondMax = max;

max = num;

} else if (num > secondMax && num !== max) {

secondMax = num;

}

}

}

if (secondMax === -*Infinity*) {

*console*.log("❗ Немає достатньо чисел, що відповідають критерію.");

} else {

*console*.log("✅ Друге за величиною число зі старшою цифрою не 9:", secondMax);

}

rl.close();

});

Лістинг 1.3 – Код програми третього завдання з коментарями

const readline = require('readline');

const rl = readline.createInterface({

input: process.stdin,

output: process.stdout

});

function leadingDigit(num) {

return parseInt(*Math*.abs(num).toString()[0]);

}

function selectionSortIndexes(arr) {

const indexes = arr.map((\_, i) => i);

for (let i = 0; i < indexes.length - 1; i++) {

let minIdx = i;

for (let j = i + 1; j < indexes.length; j++) {

const a = leadingDigit(arr[indexes[j]]);

const b = leadingDigit(arr[indexes[minIdx]]);

if (a < b) {

minIdx = j;

}

}

[indexes[i], indexes[minIdx]] = [indexes[minIdx], indexes[i]];

}

return indexes;

}

let arr = [];

let n = null;

let count = 0;

*console*.log("🔢 Програма впорядковує індекси чисел за старшою цифрою (методом вибору).");

*console*.log("📝 Спочатку введіть КІЛЬКІСТЬ чисел (наприклад: 5)");

*console*.log("➡️ Потім введіть КОЖНЕ ціле число з нового рядка, ось приклад:");

*console*.log(" 2413");

*console*.log(" 943");

*console*.log(" 81");

*console*.log(" 234");

*console*.log(" 512");

*console*.log("🔄 Після кожного числа натискайте Enter");

rl.on('line', (line) => {

if (n === null) {

n = parseInt(line.trim());

if (isNaN(n) || n <= 0) {

*console*.log("❗ Помилка: Введіть правильне число елементів.");

n = null;

return;

}

*console*.log(`📥 Тепер введіть ${n} цілих чисел по одному з нового рядка:`);

} else {

const value = parseInt(line.trim());

if (isNaN(value)) {

*console*.log("❗ Це не ціле число. Спробуйте ще раз:");

return;

}

arr.push(value);

count++;

if (count < n) {

*console*.log(`✅ Введено: ${count}/${n}. Введіть наступне число:`);

}

if (count === n) {

rl.close();

}

}

}).on('close', () => {

*console*.log("\n📋 Введена послідовність:", arr.join(' '));

const sortedIndexes = selectionSortIndexes(arr);

*console*.log("📌 Упорядковані індекси за старшою цифрою чисел:");

*console*.log(sortedIndexes.join(' '));

*console*.log("🔎 Значення у цьому порядку:");

sortedIndexes.forEach(i => process.stdout.write(arr[i] + " "));

*console*.log();

});

1. Опис інтерфейсу програми (керівництво користувача)

При виконанні коду першого завдання, користувач намагається влучити в об’єднання перших двох фігур у перетині з третьою.

При виконанні коду другого завдання користувач вводить дійсні числа, а програма обчислює друге за величиною значення серед тих, що не більше за середнє геометричне усього масиву.

При виконання коду третього завдання виводиться відсортований масив.

1. Опис тестових прикладів

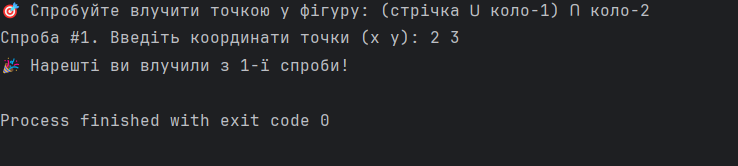


Рисунок 1.1 – Результат виконання коду першого завдання

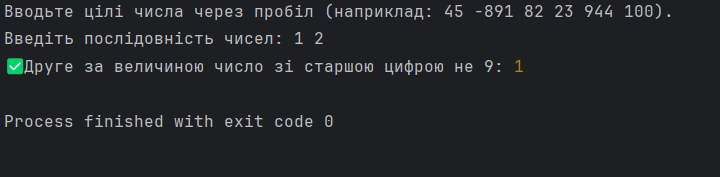


Рисунок 1.2 – Результат виконання коду другого завдання

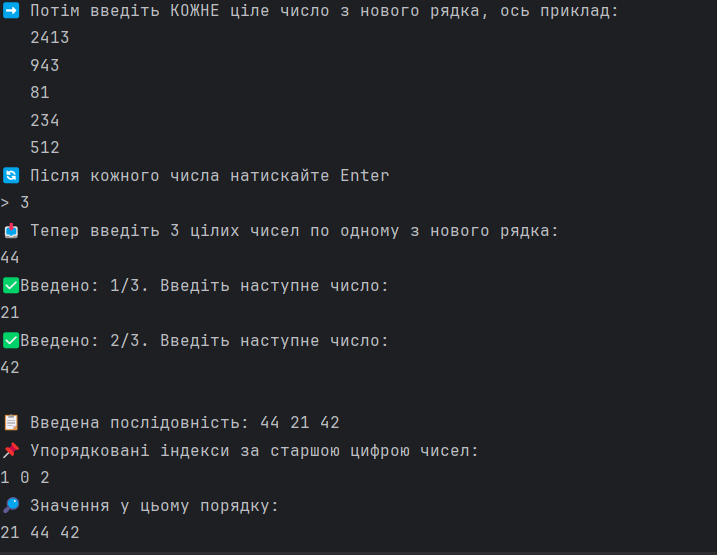


Рисунок 1.3 – Результат виконання коду третього завдання