Міністерство освіти і науки України

Національний університет «Запорізька політехніка»

кафедра програмних засобів

ЗВІТ

з лабораторної роботи №1

з дисципліни «**Спортивне програмування**» на тему:

**«**РЕКУРЕНТНІ ПОСЛІДОВНОСТІ»

Виконали:

ст. гр. КНТ-113сп Ярослав ПАНЧЕНКО  
Михайло ГРИГОРОВ

Прийняв:

ст.викл. Сергій ЛЕОЩЕНКО

2023

# Мета роботи

Вивчити основні можливості та принципи роботи з мовою рекурентні послідовності та співвідношення.

# Завдання до лабораторної роботи

3.1 Ознайомитися з основними теоретичними відомостями за темою роботи, використовуючи ці методичні вказівки, а також рекомендовану літературу.

3.2 Обрати та виконати три задачі із запропонованого переліку.

3.2.1 Задано масив M[1:N] натуральних чисел, упорядкований за не спаданням, тобто: M[1] ← M[2] ← ⋯ ← M[N]. Знайти перше натуральне число, яке не представляється сумою ніяких елементів цього масиву, при цьому сума може складатися і з одного доданка, але кожен елемент масиву може входити в неї тільки один раз.

3.2.3 Задано масив M[1:N] натуральних чисел, упорядкований за не спаданням, тобто: M[1] ← M[2] ← ⋯ ← M[N]. Написати алгоритм виплати заданої суми S мінімальною кількістю купюр гідністю M(1), ... , M(N).

3.2.8 Звести число a в натуральну ступінь n за якомога меншу кількість множень.

# Текст розробленої програми.

const M = [1, 2, 4, 8, 10, 20, 30, 40, 50];

const smallestUnobtainable = (arr) => {

  let target = 1;

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

    if (arr[i] <= target) {

      target += arr[i];

    } else {

      break;

    }

  }

  return target;

};

const result = smallestUnobtainable(M);

console.log('Не можливо отримати:', result);

const bills = [1, 2, 5, 10, 20, 50, 100, 200, 500, 1000];

const S = 1013;

const minCountOfBills = (arr, S) => {

  let result = [];

  let i = arr.length - 1;

  while (S > 0 && i >= 0) {

    while (S >= arr[i]) {

      S -= arr[i];

      result.push(arr[i]);

    }

    i--;

  }

  return result;

};

const payment = minCountOfBills(bills, S);

console.log('Кіькість купюр:', payment.length);

console.log('Купюри для виплати:', payment);

const a = 2;

const n = 10;

const exponentiation = (num, n) => {

  if (n === 0) return 1;

  if (n === 1) return num;

  let result = 1;

  let base = num;

  while (n > 0) {

    if (n % 2 === 1) {

      result \*= base;

    }

    base \*= base;

    n = Math.floor(n / 2);

  }

  return result;

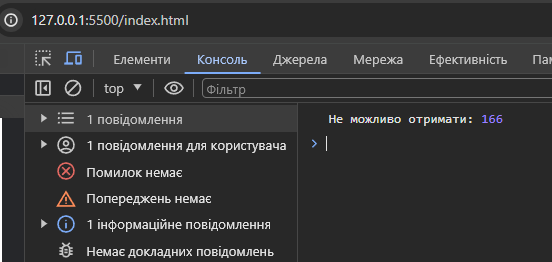
};

const result = exponentiation(a, n);

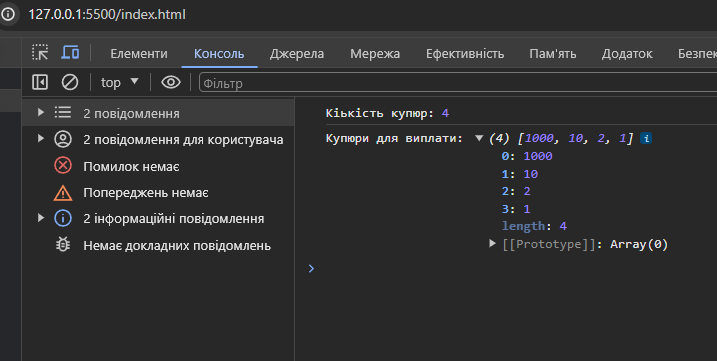
console.log(`${a} в ступені ${n} = ${result}`);

# Копії екрану, що відображають результати виконання лабораторної роботи.

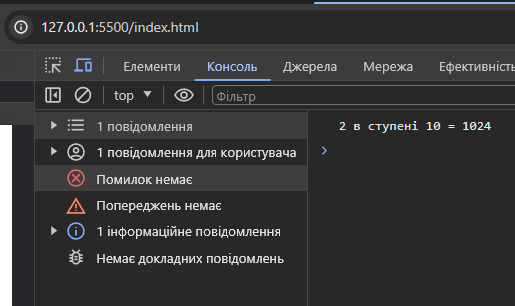
3.2.1



3.2.3



3.2.8



# 5. Висновки

Вивчив основні можливості та принципи роботи з мовою рекурентні послідовності та співвідношення.