**Державний вищий навчальний заклад**

**Ужгородський національний університет**

**Факультет інформаційних технологій**

**ПРАКТИЧНА РОБОТА № 2**

**Тема:** “Односпрямовані та двоспрямовані списки”

Виконав студент І курсу

спеціальності «Інженерія

програмного забезпечення»

Боршош Арсеній Сергійович

**Ужгород-2025**

**Мета:** набути практичних навичок створення та застосування односпрямованих та двоспрямованих списків

**Завдання до роботи:**

1. Написати програму, у якій потрібно створити лінійний односпрямований список та виконати дії згідно з власним варіантом завдання табл. 1.1.
2. Написати програму, у якій потрібно створити двоспрямований список та виконати дії згідно з власним варіантом завдання табл 1.2.
3. Оформити звіт про виконання роботи та завантажити його в системі електронного навчання ДВНЗ «УжНУ» в установлений термін.
4. Підготувати відповіді на питання до лекції з односпрямованого та двоспрямованого списку.

**Хід роботи**

**Завдання 1.1**

Створити лінійний односпрямований список із цілих чисел. Видалити зі списку перший парний елемент, який знаходиться на непарній позиції

    class Node {

        constructor(value) {

        this.value = value;

        this.next = null;

        }

    }

    class LinkedList {

        constructor() {

        this.head = null;

        }

        add(value) {

        const newNode = new Node(value);

        if (this.head === null) {

            this.head = newNode;

            return;

        }

        let current = this.head;

        while (current.next) {

            current = current.next;

        }

        current.next = newNode;

        }

        printList() {

        let current = this.head;

        let result = '';

        while (current) {

            result += current.value + ' ';

            current = current.next;

        }

        return result;

        }

        removeFirstEvenOnOddPosition() {

        let current = this.head;

        let previous = null;

        let position = 1;

        while (current) {

            if (position % 2 !== 0 && current.value % 2 === 0) {

            if (previous === null) {

                this.head = current.next;

            } else {

                previous.next = current.next;

            }

            return;

            }

            previous = current;

            current = current.next;

            position++;

        }

        }

    }

    function main() {

        const list = new LinkedList();

        let inputData = [13, 15, 10, 18, -7, -4, 20];

        inputData.forEach(i => {

        list.add(i);

        });

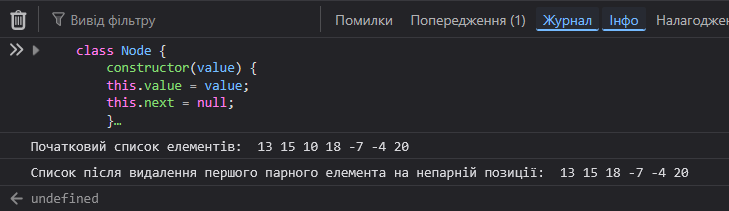
        console.log("Початковий список елементів: ", list.printList());

        list.removeFirstEvenOnOddPosition();

        console.log("Список після видалення першого парного елемента на непарній позиції: ", list.printList());

    }

    main();

****

**Завдання 1.2**

Створити двоспрямований список із дійсних чисел. Включити до списку число 2.5 перед кожним елементом з додатним значенням

class Node {

    constructor(value) {

      this.value = value;

      this.next = null;

      this.prev = null;

    }

  }

  class DoublyLinkedList {

    constructor() {

      this.head = null;

      this.tail = null;

    }

    add(value) {

      const newNode = new Node(value);

      if (this.head === null) {

        this.head = newNode;

        this.tail = newNode;

        return;

      }

      this.tail.next = newNode;

      newNode.prev = this.tail;

      this.tail = newNode;

    }

    printList() {

      let current = this.head;

      let result = '';

      while (current) {

        result += current.value + ' ';

        current = current.next;

      }

      return result;

    }

    insertBeforePositive() {

      let current = this.head;

      while (current) {

        if (current.value > 0) {

          const newNode = new Node(2.5);

          newNode.next = current;

          newNode.prev = current.prev;

          if (current.prev) {

            current.prev.next = newNode;

          } else {

            this.head = newNode;

          }

          current.prev = newNode;

        }

        current = current.next;

      }

    }

  }

  function main() {

    const list = new DoublyLinkedList();

    let inputData = [12, -3, 6, 0, -5];

    inputData.forEach(i => {

      list.add(i);

    });

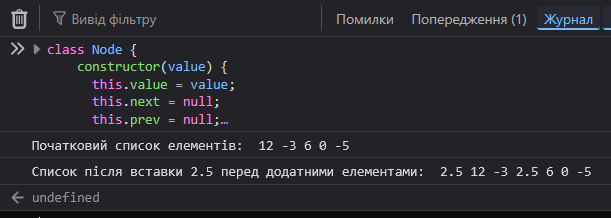
    console.log("Початковий список елементів: ", list.printList());

    list.insertBeforePositive();

    console.log("Список після вставки 2.5 перед додатними елементами: ", list.printList());

  }

  main();



**Висновок**

У ході виконання практичної роботи було реалізовано динамічні структури даних – **односпрямований та двоспрямований списки.**

Для **односпрямованого списку** було реалізовано базові операції: додавання елементів, виведення списку та видалення першого парного елемента, який знаходиться на непарній позиції.

Для **двоспрямованого списку** передбачено додавання елементів, виведення списку та вставку значення 2.5 перед кожним додатним елементом.

Робота дозволила закріпити знання про **динамічні структури даних** та їхню практичну реалізацію мовою програмування **JavaScript**, що є важливим етапом у подальшому вивченні алгоритмів та розробці програмного забезпечення.