

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

«ЛЬВІВСЬКА ПОЛІТЕХНІКА»



Лабораторна робота №1

з курсу «Дискретні моделі в САПР»:

## АЛГОРИТМ ПОБУДОВИ ДЕРЕВ

**Виконав:**

Ст.гр.КН-409

Погуляєв В.В.

Львів – 2023

## Мета роботи

Вивчення алгоритмів рішення задач побудови остових дерев.

## Теоретичні відомості

Графом  $G$  називають скінчену множину  $V$  з нереклексивним симетричним відношенням  $R$  на  $V$ . Визначим  $E$  як множину симетричних пар в  $R$ . Кожний елемент  $V$  називають вершиною. Кожний елемент  $E$  називають ребром, а  $E$  множиною ребер  $G$ . Граф називається зв'язним, якщо в ньому для будь-якої пари вершин знайдеться ланцюг, який їх з'єднує, тобто, якщо по ребрах (дугах) можна потрапити з будь-якої вершини в іншу. Цикл - це ланцюг, в якого початкова і кінцева точки співпадають. Дерево - це зв'язний граф без циклів. Остовним деревом графа називається будь-яке дерево, яке утворене сукупністю дуг, які включають всі вершини графа. В графі, який показано на рис.1, сукупність дуг  $\{a, \gamma, \sigma\}$  утворює остовне дерево, так як вона включає всі вершини даного графа  $a, b, c, d$ . Будь-який зв'язний граф має остовне дерево. Коренем орієнтованого дерева (прадерева) називається його вершина, в яку не входить жодна з дуг. Орієнтований ліс визначається як звичайний, тільки складається не з простих дерев, а орієнтованих. Остовним орієнтованим деревом називається орієнтоване дерево, яке одночасно є і остовним деревом. Остовним орієнтованим лісом називається орієнтований ліс, який включає всі вершини відповідного графа. Вага дерева - це сума ваг його ребер. Поставимо у відповідність кожній дузі  $(x, y)$  графа  $G$  вагу  $a(x, y)$ . Вага орієнтованого лісу (або орієнтованого дерева) визначається як сума ваг дуг, що входять в даний ліс (дерево). Максимальним орієнтованим лісом графа  $G$  називається орієнтований ліс графа  $G$  з максимально можливою вагою. Максимальним орієнтованим деревом графа  $G$  називається орієнтоване дерево графа  $G$  з максимально можливою вагою. Мінімальні орієнтовані ліс і дерево визначаються аналогічним чином. Куш(букет) - зв'язний фрагмент графа.

## Лабораторне завдання

1. Отримати у викладача індивідуальне завдання.
2. Підготувати програму для вирішення виданого завдання.
3. Запустити на покрокове виконання програму побудови мінімального покриваючого дерева і максимального покриваючого дерева.
4. Здійснити перевірки роботи програм з результатами розрахунків проведених вручну.
5. Зафіксувати результати роботи. 6. Оформити і захистити звіт.

**Індивідуальний варіант** – реалізувати алгоритм Крускала.

## Код програми

Нижче наведено фрагмент коду в якому реалізовується алгоритм Крускала.

```

Void kruskalMST() {
    List<Edge> result = new ArrayList<>();
    int[] parent = new int[vertices];
    int[] rank = new int[vertices];
    for (int i = 0; i < vertices; i++) {
        parent[i] = i;
        rank[i] = 0;
    }
    Collections.sort(edges);
    int index = 0;
    while (result.size() < vertices - 1) {
        Edge edge = edges.get(index);
        index++;
        int x = find(parent, edge.source);
        int y = find(parent, edge.destination);
        if (x != y) {
            result.add(edge);
            union(parent, rank, x, y);
        }
    }
    for (Edge edge : result) {
        System.out.println(edge.source + « - « + edge.destination + «: « +
        edge.weight);
    }
}

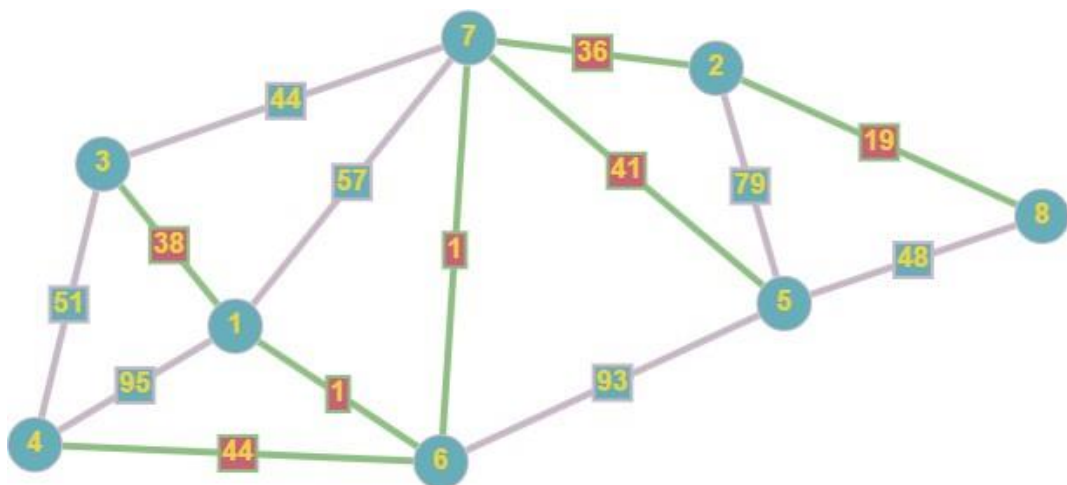
```

Посилання на GitHub – [https://github.com/flippflopp/DM\\_Pohuliaiev](https://github.com/flippflopp/DM_Pohuliaiev)

### Аналіз результатів

Використовуючи файл `I1\_3.txt` з тестовими даними, виконав аналітичну побудову остового дерева і порівняв з результатом, який видала програма.

Аналітична побудова:



Побудова за допомогою програми:

```
0 - 5: 1
5 - 6: 1
1 - 7: 19
1 - 6: 36
0 - 2: 38
4 - 6: 41
3 - 5: 44
```

### **Висновок**

В ході виконання лабораторної роботи, вивчив алгоритми рішення задач побудови остових дерев.