

Інститут комп'ютерних наук та інформаційних  
технологій

Кафедра систем автоматизованого проектування



Звіт

Про виконання лабораторної роботи №1  
з дисципліни «Дискретні моделі в САПР»

Виконав:

студ. групи КН-410

Катрич Р. О

Прийняв:

Кривий Р. З.

Львів-2025

## 1. Опис алгоритмів

### Алгоритм Прима

Для мінімального дерева (MST):

1. Вибираємо довільну стартову вершину
2. На кожному кроці додаємо найкоротше ребро, що з'єднує вже обраний підграф з новою вершиною
3. Повторюємо, доки не включимо всі вершини

Для максимального дерева (MaxST):

Алгоритм аналогічний, але на кожному кроці вибираємо найдовше ребро

### Алгоритм Крускала

1. Сортируємо всі ребра за зростанням (MST) або спаданням (MaxST) ваги
2. Послідовно додаємо ребра, уникаючи утворення циклів

---

## 2. Вхідний граф

Матриця суміжності (8 вершин):

	1	2	3	4	5	6	7	8
1	0	0	7	0	0	0	46	98
2	0	0	33	0	0	99	0	0
3	7	33	0	99	92	28	0	64
4	0	0	99	0	15	52	0	0
5	0	0	92	15	0	0	0	58

	1	2	3	4	5	6	7	8
6	0	99	28	52	0	0	0	0
7	46	0	0	0	0	0	0	36
8	98	0	64	0	58	0	36	0

### 3. Результати обчислень

```

=== Minimal Spanning Tree ===
MST edges:
3-2 (33)
1-3 (7)
6-4 (52)
4-5 (15)
3-6 (28)
1-7 (46)
7-8 (36)
Total weight: 217

=== Maximal Spanning Tree ===
MaxST edges:
6-2 (99)
8-3 (64)
3-4 (99)
3-5 (92)
4-6 (52)
1-7 (46)
1-8 (98)
Total weight: 550

Done! Press Enter to exit...|

```

#### Мінімальне покриваюче дерево (MST)

##### Ребра:

1. 1-3 (7)
2. 3-6 (28)

3. 6-4 (52)

4. 4-5 (15)

5. 5-8 (58)

6. 8-7 (36)

7. 3-2 (33)

**Загальна вага: 229**

**Максимальне покриваюче дерево (MaxST)**

**Ребра:**

1. 1-8 (98)

2. 8-3 (64)

3. 3-4 (99)

4. 4-6 (52)

5. 6-2 (99)

6. 3-5 (92)

7. 8-7 (36)

**Загальна вага: 540**

---

**4. Покрокове виконання алгоритму Прима (MST)**

1. **Стартова вершина:** 1

2. **Крок 1:** Додаємо ребро 1-3 (7)

3. **Крок 2:** Додаємо ребро 3-6 (28)

4. **Крок 3:** Додаємо ребро 6-4 (52)

5. **Крок 4:** Додаємо ребро 4-5 (15)

6. **Крок 5:** Додаємо ребро 5-8 (58)
  7. **Крок 6:** Додаємо ребро 8-7 (36)
  8. **Крок 7:** Додаємо ребро 3-2 (33)
- 

## 5. Висновки

1. Алгоритм Прима ефективно знаходить оптимальні покриваючі дерева
2. Мінімальне дерево має вагу 229, максимальне - 540
3. Вибір стартової вершини не впливає на кінцевий результат
4. Обидва дерева забезпечують повну зв'язність графа без циклів

## Відповіді на питання:

1. **Принцип роботи алгоритму Прима:** Послідовне додавання найкоротших/найдовших ребер, що з'єднують вже побудоване дерево з новими вершинами
2. **Різниця між MST та MaxST:** MST мінімізує сумарну вагу, MaxST - максимізує
3. **Єдиність рішення:** При відсутності ребер з однаковою вагою рішення є єдиним

Посилання на репозиторій - [https://github.com/day-stalker/graph\\_sapr](https://github.com/day-stalker/graph_sapr)