

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ
УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра системи штучного інтелекту

Лабораторна робота 3
з дисципліни
“Дискретна математика”

Виконав:
студент групи КН-109
Гладун Ярослав
Викладач:
Мельникова Н. І.

Львів - 2018 р.

Тема: Основні операції над графами. Знаходження остова мінімальної ваги за алгоритмом Пріма-Краскала

Мета: набуття практичних вмінь та навичок з використання алгоритмів Пріма і Краскала

Теоретичні відомості:

Теорія графів дає простий, доступний і потужний інструмент побудови моделей прикладних задач, є ефективним засобом формалізації сучасних інженерних і наукових задач у різних областях знань. Графом G називається пара множин V, E , де V – множина вершин, перенумерованих числами $1, 2, \dots, n$; V, E – множина упорядкованих або неупорядкованих пар $e = (v', v'')$, $v' \in V, v'' \in V$, називаних дугами або ребрами, $E = \{e\}$. При цьому не має примусового значення, як вершини розташовані в просторі або площині і які конфігурації мають ребра.

Неорієнтованим графом G називається граф у якого ребра не мають напрямку. Такі ребра описуються неупорядкованою парою (v', v'') .

Орієнтований граф (орграф) – це граф ребра якого мають напрямок та можуть бути описані упорядкованою парою (v', v'') . Упорядковане ребро називають дугою. Граф є змішаним, якщо наряду з орієнтованими ребрами (дугами) є також і неорієнтовані.

При розв'язку задач змішаний граф зводиться до орграфа. Кратними (паралельними) називаються ребра, які зв'язують одні і ті ж вершини.

Якщо ребро виходить та й входить у дну і ту саму вершину, то таке ребро називається петлею. Мультиграф – граф, який має кратні ребра.

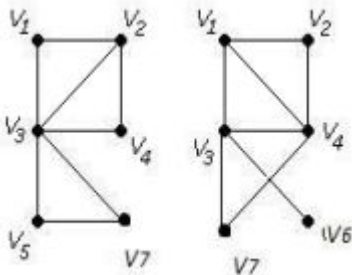
Варіант 2 .

Завдання 1.

Завдання № 1. Розв'язати на графах наступні задачі:

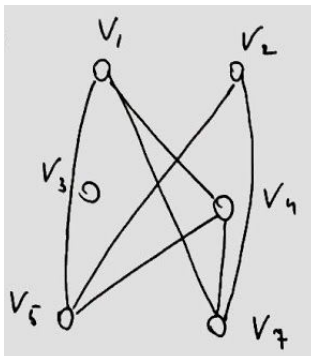
1. Виконати наступні операції над графами:

- 1) знайти доповнення до першого графу,
- 2) об'єднання графів,
- 3) кільцеву суму G_1 та G_2 ($G_1 + G_2$),
- 4) розщепити вершину у другому графі,
- 5) виділити підграф A , що складається з 3-х вершин в G_1 і знайти стягнення A в G_1 ($G_1 \setminus A$), 6) добуток графів.

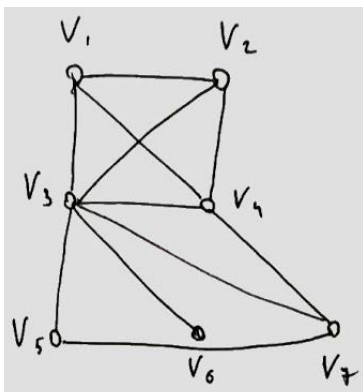


Розв'язок завдання 1.

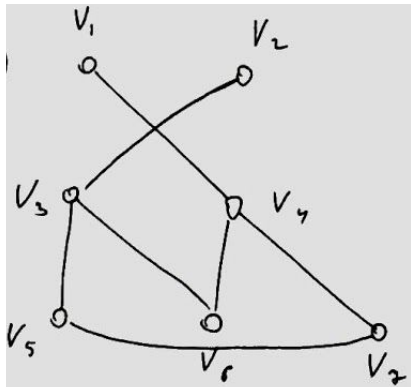
1)



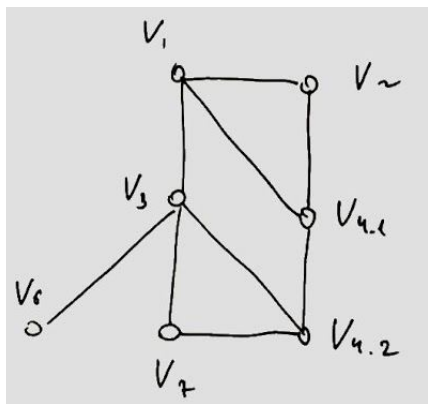
2)



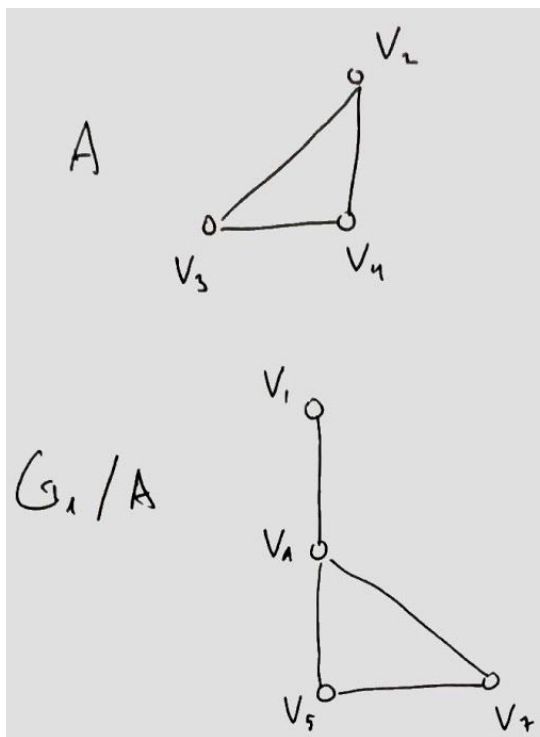
3)



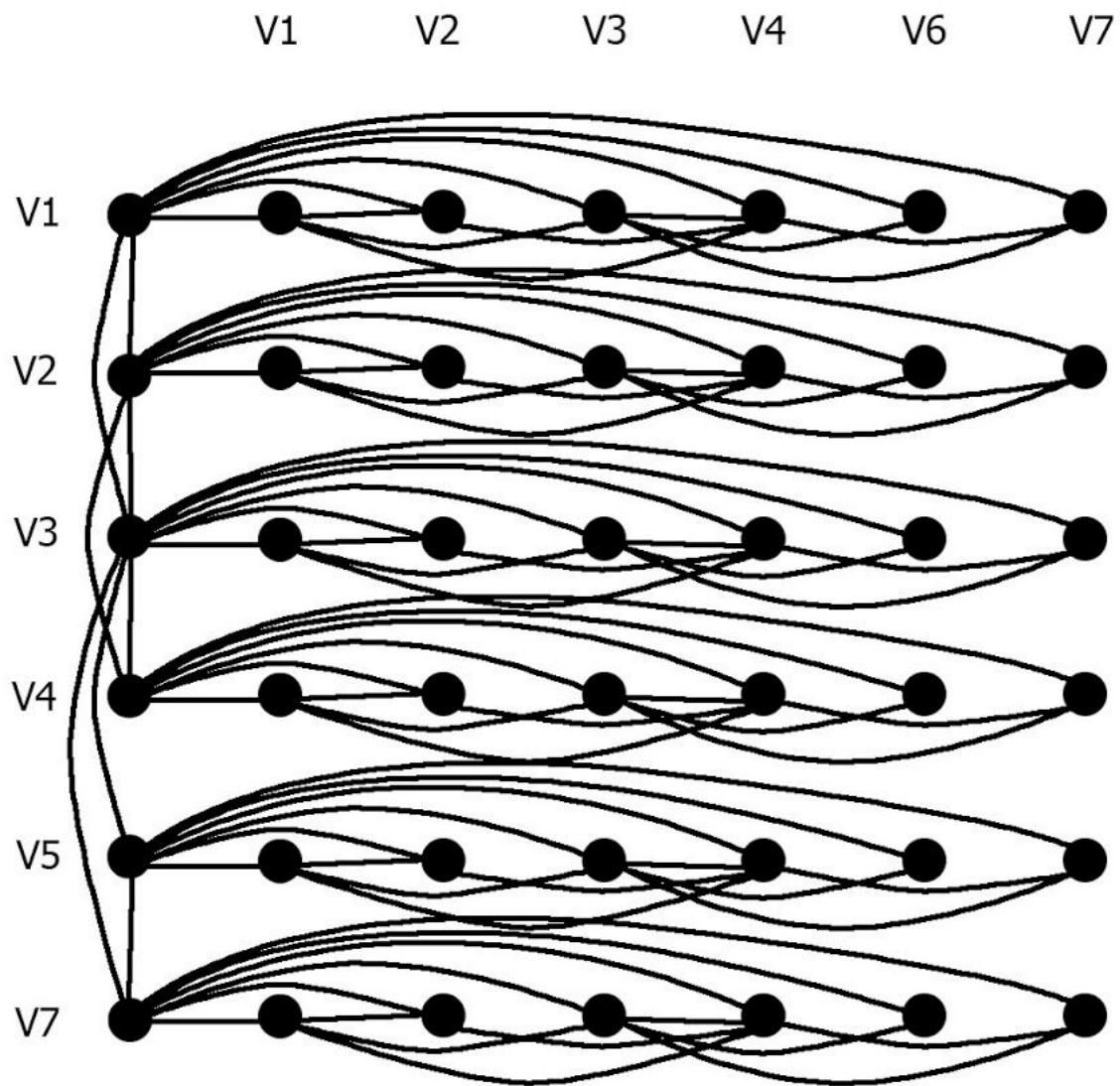
4)



5)

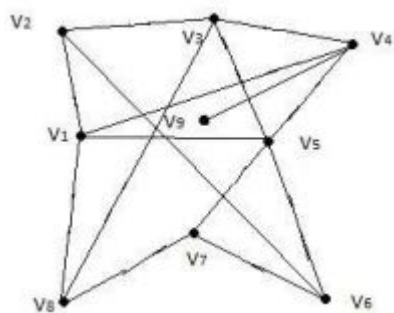


6)



Завдання 2.

Знайти таблицю суміжності та діаметр графа.



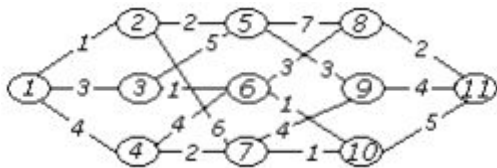
Розв'язок завдання 2.

	V1	V2	V3	V4	V5	V6	V7	V8	V9
V1	0	1	0	1	1	0	0	1	0
V2	1	0	1	0	0	1	0	0	0
V3	0	1	0	1	1	0	0	1	0
V4	1	0	1	0	1	0	0	0	1
V5	1	0	1	1	0	1	1	0	0
V6	0	1	0	0	1	0	1	0	0
V7	0	0	0	0	1	1	0	1	0
V8	1	0	1	0	0	0	1	0	0
V9	0	0	0	1	0	0	0	0	0

$D = 3$;

Завдання 3.

Знайти двома методами (Краскала і Прима) мінімальне остове дерево графа.



Розв'язок завдання 3.

За алгоритмом Краскала:

Додаємо ребра в такому порядку: (1, 2), (7, 10), (6, 10), (3, 6), (8, 11), (4, 7), (2, 5), (5, 9), (6, 8), (1, 3)

За алгоритмом Прима:

Додаємо ребра в такому порядку: (1, 2), (5, 2), (3, 1), (6, 3), (10, 6), (7, 10), (4, 7), (8, 6), (11, 8), (9, 5)

Код програми:

```
public Graph AlgorithmKruskal()
```

```

{
    Graph resultGraph = new Graph();
    List<Rib> sortRibs = new List<Rib>();

    for (int i = 0; i < this.ribs.Count; i++)
        sortRibs.Add(this.ribs[i]);
    sortRibs.Sort((a, b) => a.ValuePrice.CompareTo(b.ValuePrice));

    for (int i = 0; i < this.tops.Count; i++)
        resultGraph.AddTop(this.tops[i].Name);

    for (int i = 0; i < sortRibs.Count; i++)
    {
        resultGraph.AddRib(sortRibs[i].StartTop.Name,
sortRibs[i].EndTop.Name,
        sortRibs[i].ValuePrice);

        if (resultGraph.loopCheck())
        {
            resultGraph.DeleteRig(sortRibs[i].StartTop.Name,
sortRibs[i].EndTop.Name);
        }
        else
        {
            // <DEBUG>
            Console.WriteLine("ADD ({0} {1})", sortRibs[i].StartTop.Name,
sortRibs[i].EndTop.Name);
            resultGraph.writeMatrix();
            //<DEBUG/>
        }
    }
}

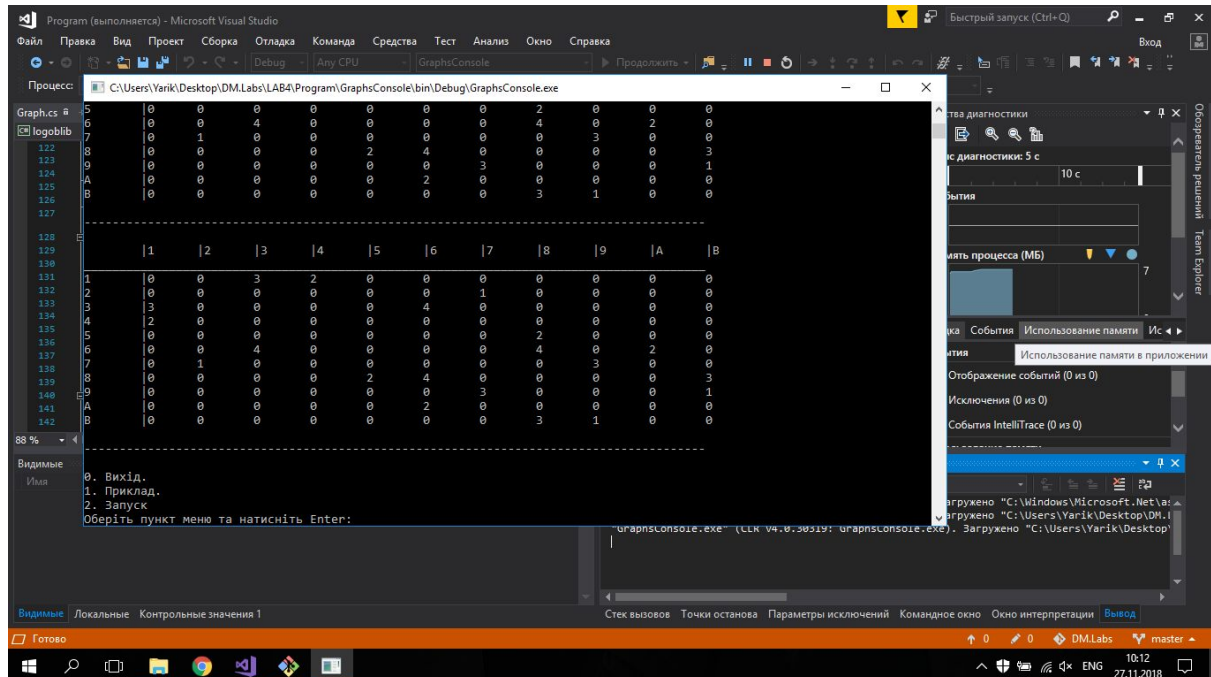
```

```

return resultGraph;
}

```

Результат виконання програми.



Висновок: Отже, на цій лабораторній роботі я ознайомився із графами, навчився виконувати різні операції над ними, набув практичних навичок з використання алгоритмів Пріма і Краскала. Також я поглибив свої знання ООП та платформи .NET.