КЕРІВНИК КУРСОВОЇ РОБОТИ
ДОКТОР ФІЗ.-МАТ НАУК, ПРОФЕСОР
ОЛІЙНИК Б. В.

ВЛАСТИВОСТІ БУЛЕВИХ ОПЕРАЦІЙ НА ДИСТАНЦІЙНО-ТРАНЗИТИВНИХ ГРАФАХ

МП ПМ-1
Будишевська Марина Олександрівна

ГРАФ

В математиці особливу роль грають графи, які володіють симетрією(великою групою автоморфізмів).

Саме у дистанційно-транзитивних графах, група автоморфізмів діє транзитивно на впорядкованих парах рівновіддалених вершин.

Варто підмітити, що широко використовувані у теорії алгебраїчних кодів схеми відношень Хемінга та Джонсона являють собою дистанційно-транзитивний граф

Неформально граф можна розглядати як геометричну конфігурацію точок та ліній, які з’єднують ці точки із стрілками або ні.У теорії графів як математичної дисципліни першою роботою вважають статтю Ойлера у 1735-1736 році. У напрацюванні розглядалась задача про «Сім мостів Кеніґсберґа».

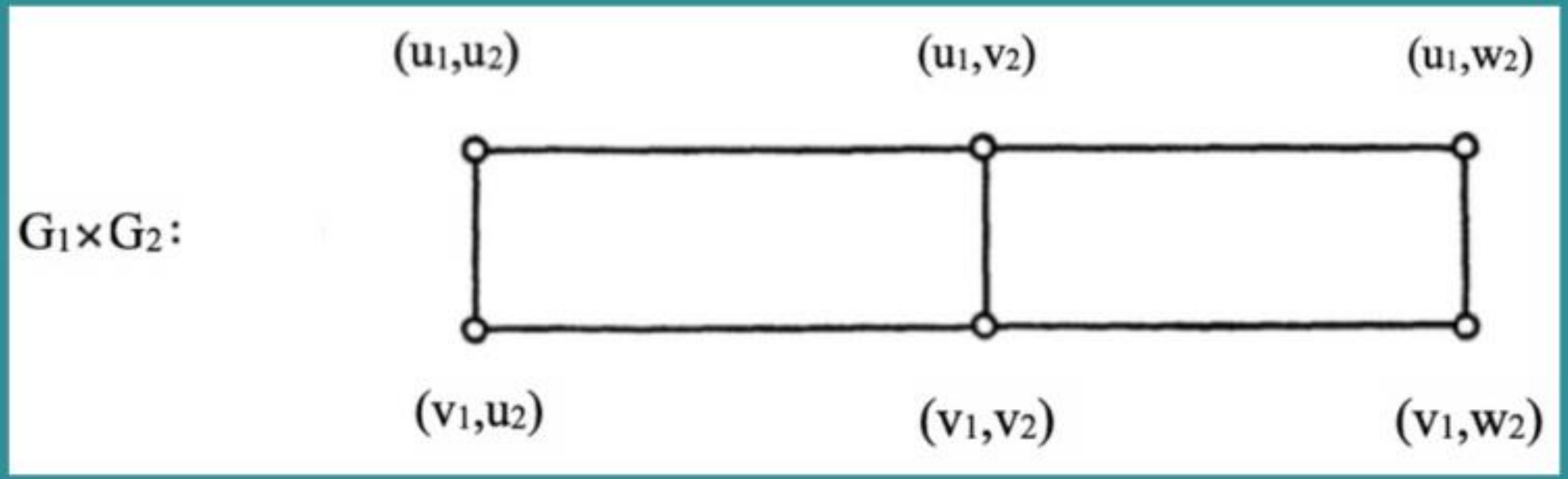
ДИСТАНЦІЙНО-ТРАНЗИТИВНІ ГРАФИ

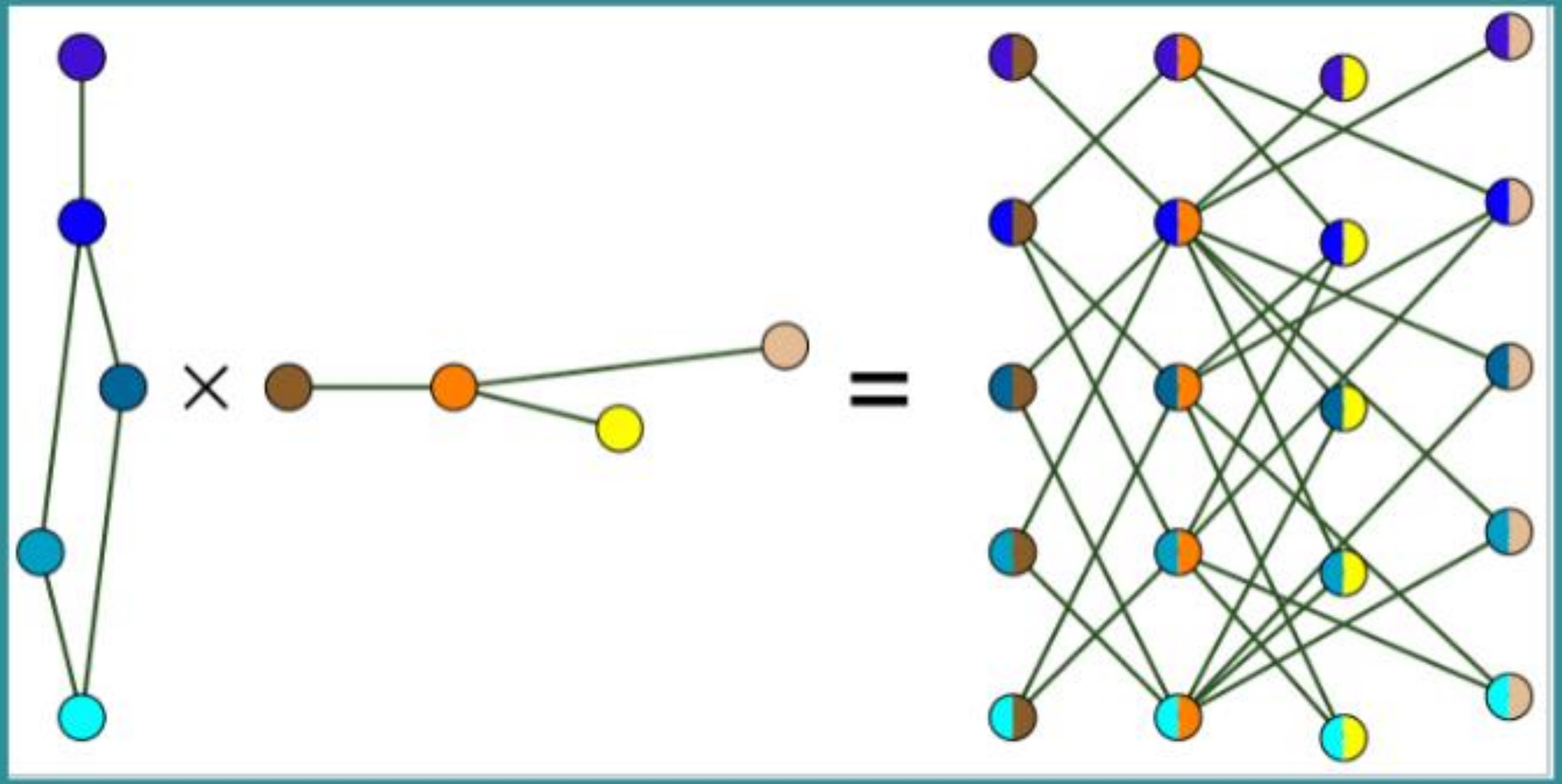
Нехай існує простий зв’язний граф G із функцією відстані dG. У такому випадку граф є дистанційно-транзитивним, якщо для довільних вершин графа G, для яких виконується рівність dG (u, v) = dG (x, y), існує автоморфізм g в G, такий що g(u) = x і g(v) = y. Інакшими словами – граф є дистанційно-транзитивним у тому випадку, якщо його група автоморфізмів Aut(G) діє транзитивно на впорядкованих парах рівновіддалених вершин.

БУЛЕВІ ОПЕРАЦІЇ

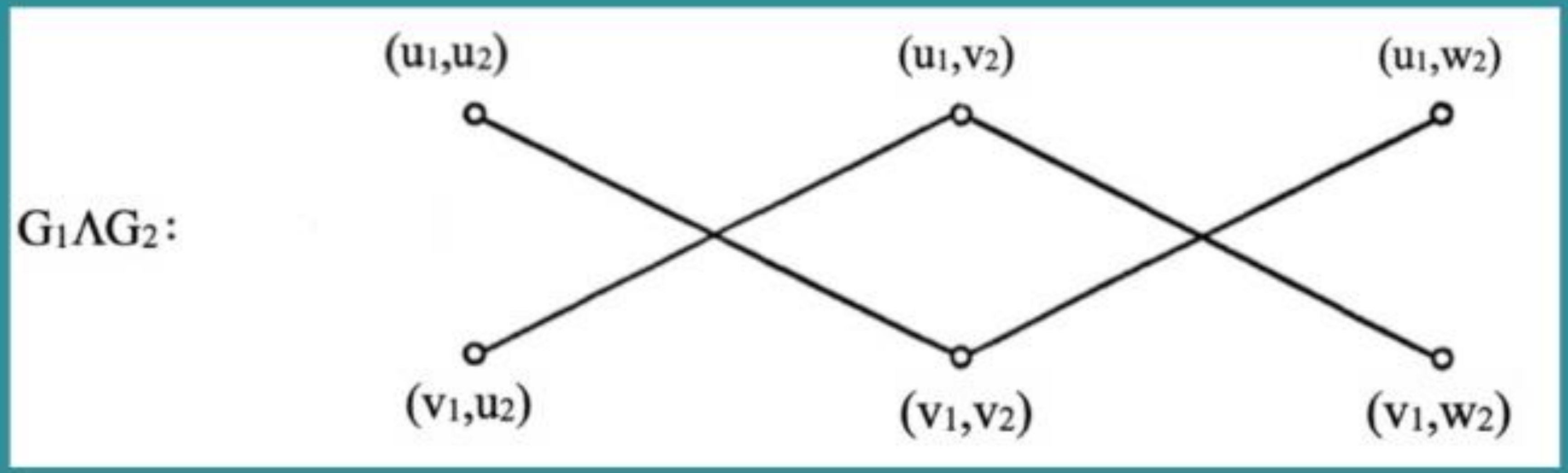
Добуток Кронекера(1)
Декартовий добуток(2)
Кон'юнкція(3)

2





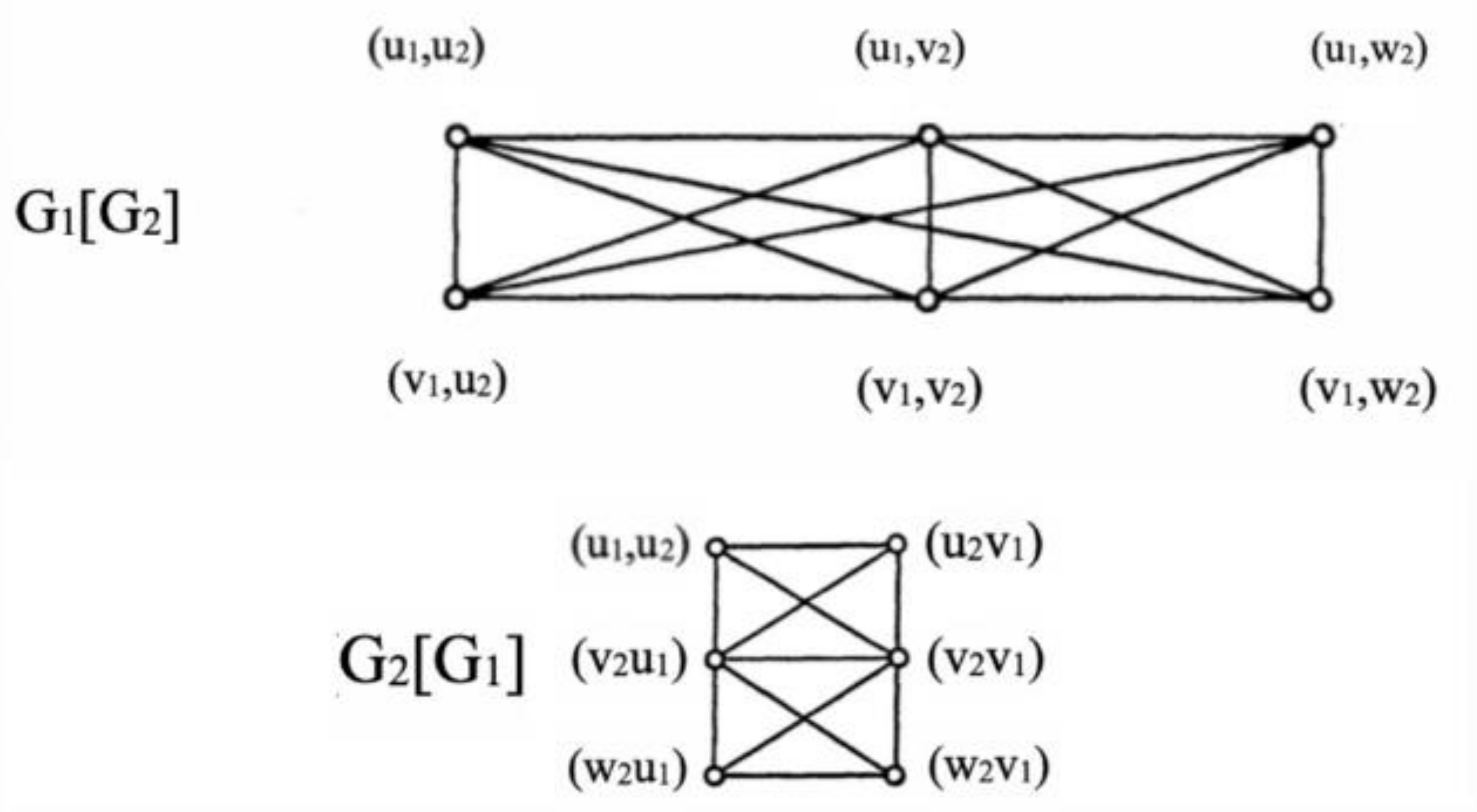
1



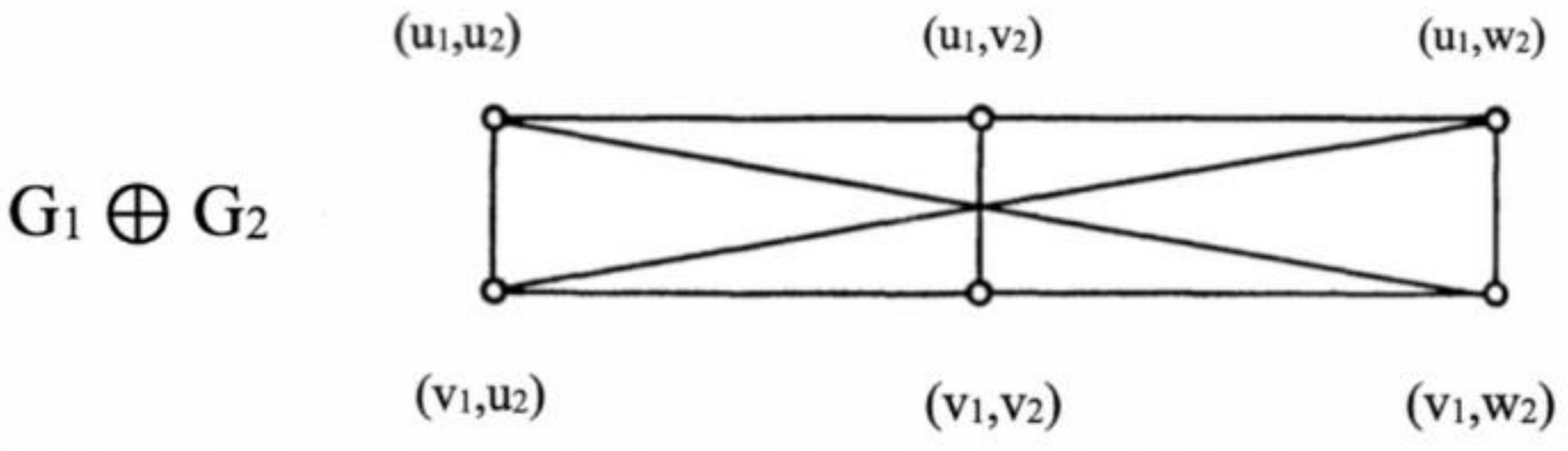
3

БУЛЕВІ ОПЕРАЦІЇ

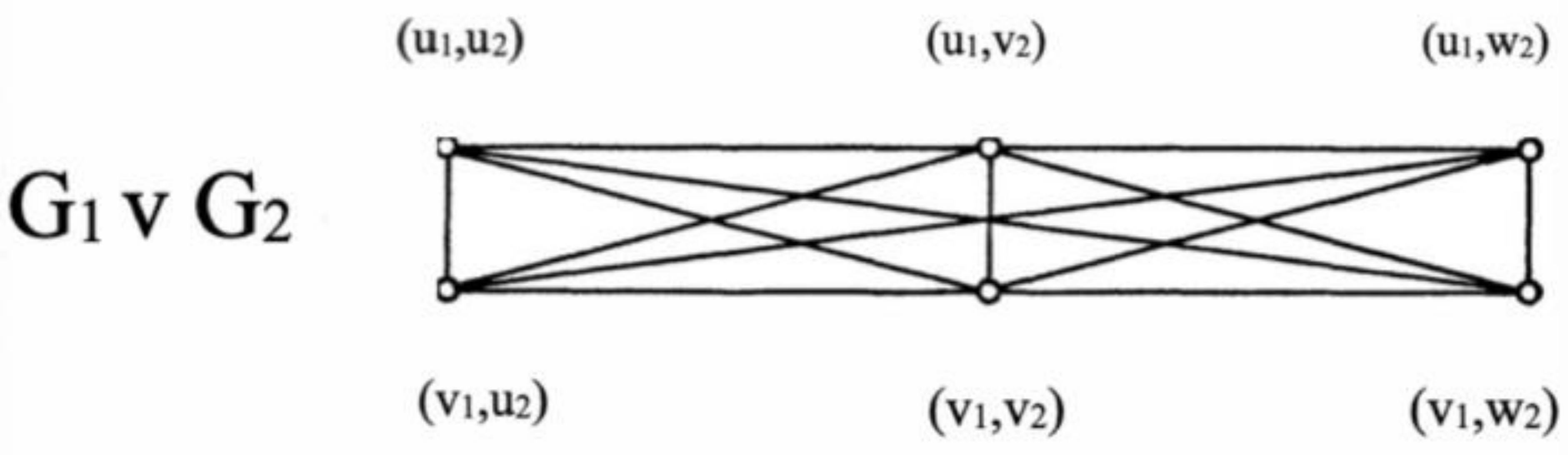
КОМПОЗИЦІЯ



СИМЕТРИЧНА РІЗНИЦЯ



ДИЗ'ЮНКЦІЯ



Булеві операції на дистанційно-транзитивних графах

- Нехай G та H зв'язні, дистанційно – транзитивні графи, тоді G + H є дистанційно-транзитивним, якщо і тільки якщо G має ті ж парам Ham(D(G), a_1^G+2)
 і H має той самий параметр, як
 Ham(D(H), a_1^G+2) (за $b_j^H = 1 + a_{j+1}^H + b_{j+1}^H$).

серед цих композицій лише сума графів дозволяє зберегти дистанційну транзитивність завдяки факторам, які можуть мати довільно великий діаметр. Це можна пояснити тим, що вона є єдиною з цих композицій для яких формула дистанційності не передбачає подальших співвідношень між відстанями координат (наприклад, max, min, od, ed тощо).

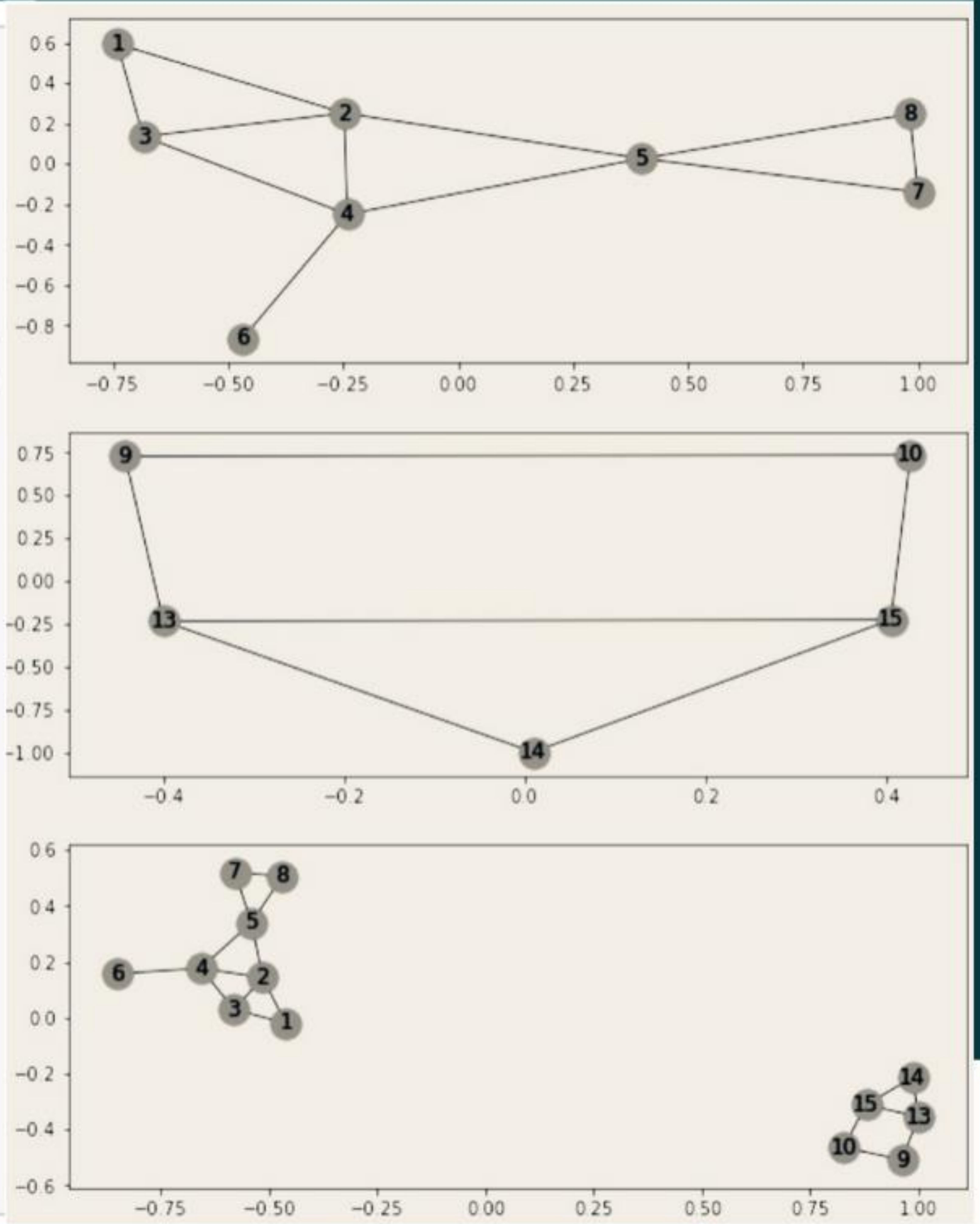
- G x H є дистанційно-транзитивним, якщо і тільки якщо кожен G ≅ H ≅ $K_{n,n}$ для деяких n∈ N (за $1 + a_j^H + c_j^H = c_{j+1}^H$)
- G ⊗ H є дистанційно-транзитивним, якщо і тільки якщо G ≅ K_m і H ≅ K_n для деяких m, n ∈ N.
- Якщо G має принаймні три вершини, тоді G ⊕ H є дистанційно – транзитивним якщо і тільки якщо існує m, n, t ∈ N таке що G ≅ $K_{m,\dots,m}(t)$ і H ≅ K_n.

K_2 ⊕ H є дистанційно–транзитивним якщо і тільки якщо параметри H задовольняє наступні відношення для всіх j = 0, 1, … , D(H)-1:

$$b_j^H = 1 + a_{j+1}^H + b_{j+1}^H, \qquad 1 + a_j^H + c_j^H = c_{j+1}^H$$

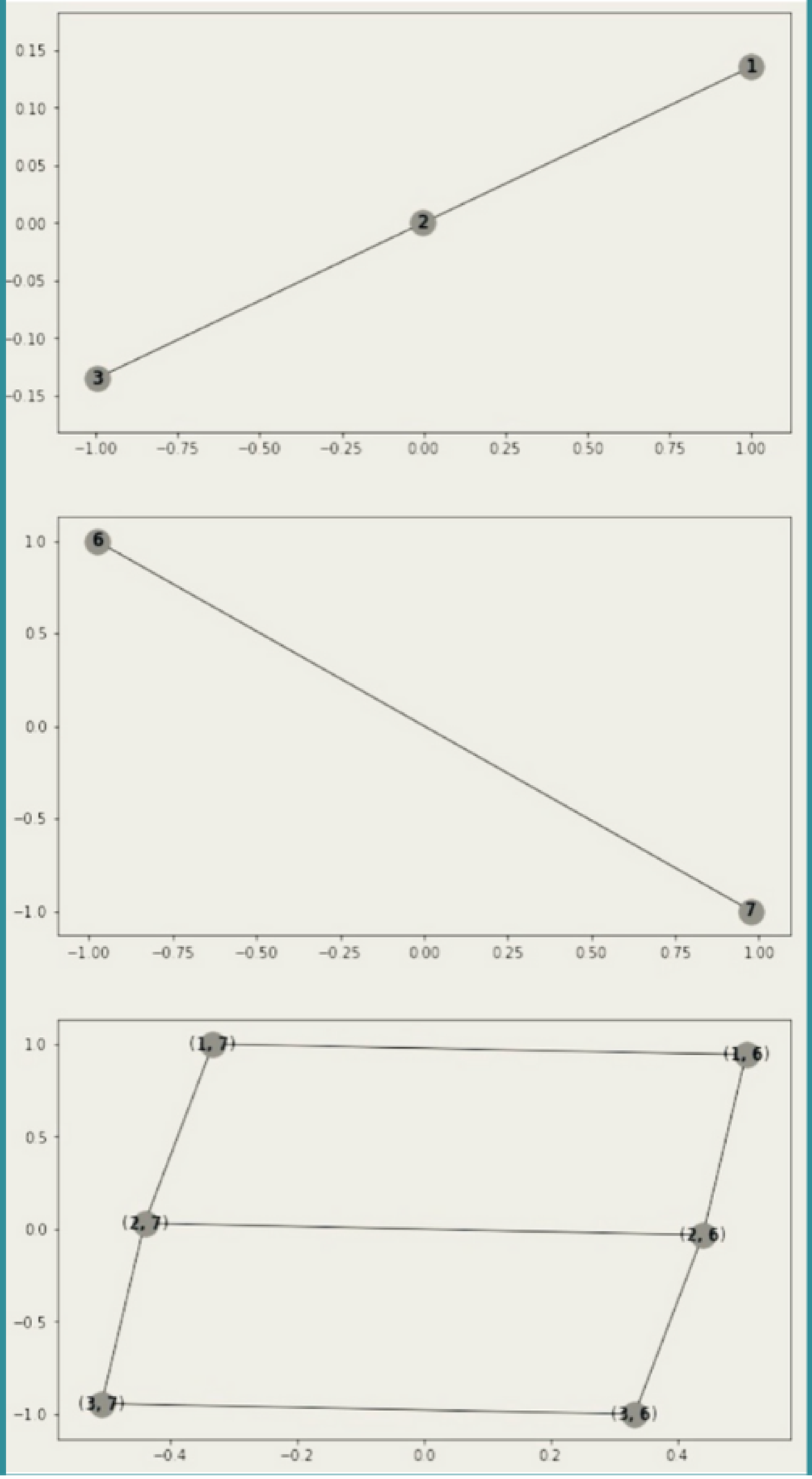
ПРОГРАМНА РЕАЛІЗАЦІЯ ДЕЯКИХ ОПЕРАЦІЙ НА **PYTHON**

In [ ]:
```
import networkx as nx
import matplotlib.pyplot as plt

G = nx.Graph()

plt.figure(figsize =(9, 12))
G.add_edges_from([(1, 2), (1, 3), (2, 3), (2, 4), (2, 5), (3, 4),
                          (4, 5), (4, 6), (5, 7), (5, 8), (7, 8)])

# створюємо перший граф
plt.subplot(311)
nx.draw_networkx(G)

H = nx.Graph()
H.add_edges_from([(13, 14), (13, 15), (13, 9),
                  (14, 15), (15, 10), (9, 10)])

# створюємо другий граф
plt.subplot(312)
nx.draw_networkx(H)


N = nx.union(G, H)
plt.subplot(313)
nx.draw_networkx(I)
```



Результат для кон'юнкції

```python
In [ ]: import networkx as nx
        import matplotlib.pyplot as plt

        G = nx.Graph()

        plt.figure(figsize =(9, 18))
        G.add_edges_from([(1, 2), (2, 3)])

        #перший граф
        plt.subplot(311)
        nx.draw_networkx(G)

        H = nx.Graph()
        H.add_edges_from([(6, 7)])
        # другий граф
        plt.subplot(312)
        nx.draw_networkx(H)


        N = nx.cartesian_product(G, H)
        plt.subplot(313)
        nx.draw_networkx(I)
```



Результат для декартового добутку

```
In [ ]: import networkx as nx
        import matplotlib.pyplot as plt

        G = nx.Graph()

        plt.figure(figsize =(9, 15))
        G.add_edges_from([(1, 2), (1, 3), (2, 3), (2, 4)])

        # перший граф
        plt.subplot(311)
        nx.draw_networkx(G)

        H = nx.Graph()
        H.add_edges_from([(3, 7), (7, 4), (3, 4)])
        # другий граф
        plt.subplot(312)
        nx.draw_networkx(H)


        N= nx.compose(G, H)
        plt.subplot(313)
        nx.draw_networkx(I)
```

Результат для композиції двох графів