Прикарпатський національний університет імені Василя Стефаника
Кафедра алгебри та геометрії

ДИПЛОМНА РОБОТА

на тему: ЗАДАЧІ НА ГРАФАХ

Студентки IV курсу, групи М-41
Спеціальності 111 «Математика»
Заліської Віталії Богданівни

Керівник:
Копорх Катерина Миколаївна
Рецензент:
Заторський Роман Андрійович

Національна шкала: _______________
Університетська шкала: ____________
Оцінка ECTS: ____________________

Члени комісії:

____
(підпис) (прізвище та ініціали)

____
(підпис) (прізвище та ініціали)

____
(підпис) (прізвище та ініціали)

м. Івано-Франківськ – 2025 рік

Прикарпатський національний університет імені Василя Стефаника

Факультет математики та інформатики
Кафедра алгебри та геометрії
Освітній рівень «бакалавр»
Спеціальність 111 математика

Затверджено на засіданні кафедри алгебри та геометрії
Протокол №2 від 21.10.2024р.
Завідувач кафедри Никифорчин О.Р.

ЗАВДАННЯ НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ
Заліській Вітілії Богданівні

1. Тема роботи: «Задачі на графах»
2. Керівник роботи: Копорх Катерина Миколаївна
3. Перелік питань, які потрібно розробити: дати визначення основних понять теорії графів. Сформулювати класифікацію графів за різними ознаками. Розкрити поняття задач на графах та надати приклади типових задач. Охарактеризувати сучасні підходи до розв'язування складних задач на графах. Розглянути задачі на графах з параметрами або змінними обмеженнями. Проаналізувати чисельні та обчислювальні методи реалізації графових алгоритмів, а також їх ефективність за обчислювальною складністю, стабільністю та масштабованістю. Пояснити практичне застосування графів у різних сферах, таких як інформатика, логістика, електроніка, соціологія, біоінформатика, транспортні системи, економіка.
4. Дата видачі завдання: 21.10.2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк виконання	Примітки
1	Попереднє вивчення стану питання в науці і практиці	21.10.2024 – 31.10.2024	Виконано
2	Обґрунтування актуальності дослідження та формулювання мети	01.11.2024 – 10.11.2024	Виконано
3	Огляд і аналіз основних понять теорії графів та класифікації графів	11.11.2024 – 25.11.2024	Виконано
4	Дослідження задач на графах та методів їх розв'язання	26.11.2024 – 10.12.2024	Виконано
5	Опис алгоритмів для розв'язування задач на графах	11.12.2024 – 25.12.2024	Виконано
6	Збір і аналіз прикладів практичного застосування задач на графах	01.01.2025 – 31.01.2025	Виконано
7	Проведення власного дослідження	01.02.2025 – 31.03.2025	Виконано
8	Оформлення повного тексту дипломної роботи відповідно до вимог	01.04.2025 – 30.04.2025	Виконано
9	Підготовка презентаційних та супровідних матеріалів до захисту	01.05.2025 – 31.05.2025	Виконано

Студент ______________ Заліська В. Б.
(підпис) (Прізвище та ініціали)

Керівник роботи ______________ Копорх К. М.
(підпис) (Прізвище та ініціали)

АНОТАЦІЯ

до дипломної роботи на здобуття першого (бакалаврського) рівня вищої освіти

Заліської Віталії Богданівни

на тему:

«Задачі на графах»

Дипломна робота присвячена теоретичному та практичному дослідженню задач на графах як важливої частини дискретної математики та інформатики. Метою дослідження є аналіз основних типів задач на графах, вивчення алгоритмів їх розв’язування та можливостей практичного застосування в різних галузях.

У роботі охарактеризовано базові поняття теорії графів: вершини, ребра, шляхи, цикли, дерева, орієнтовані та неорієнтовані графи. Подано класифікацію графів за структурними властивостями та особливостями задач. Розглянуто типові задачі: пошук найкоротшого шляху, побудова мінімального остовного дерева, визначення ейлерового і гамільтонового циклів, знаходження компонент зв’язності тощо. Особливу увагу приділено класичним алгоритмам: пошуку в глибину (DFS), пошуку в ширину (BFS), алгоритмам Дейкстри, Прима, Крускала, Флойда–Уоршелла. Проаналізовано їхню складність, ефективність та застосовність до реальних задач. Також розглянуто сучасні підходи, включно з евристичними та чисельними методами.

У практичній частині подано приклади розв’язування задач із використанням графових структур, зокрема у логістиці, маршрутизації, аналізі соціальних мереж. Проведено моделювання та аналіз ефективності обраних алгоритмів.

Результати дослідження можуть бути використані в навчанні дискретної математики, алгоритмів та програмування у школі, коледжі й закладах вищої освіти.

Загальний обсяг роботи становить 35 сторінок, основний текст – 28 сторінки.

Ключові слова: граф, вершина, ребро, алгоритм, задача на графі, найкоротший шлях, теорія графів.

ANNOTATION

of the thesis for the first (bachelor's) level of higher education

by Vitaliia Bohdanivna Zaliska

on the topic:

«Graph Problems»

The bachelor’s thesis is devoted to the theoretical and practical study of graph problems as an essential part of discrete mathematics and computer science. The aim of the research is to analyze the main types of graph problems, study algorithms for their solution, and explore their practical applications in various fields.

The work characterizes the basic concepts of graph theory: vertices, edges, paths, cycles, trees, directed and undirected graphs. A classification of graphs is presented based on their structural properties and the specifics of problems. Typical tasks are considered: shortest path search, construction of a minimum spanning tree, identification of Eulerian and Hamiltonian cycles, determination of connected components, and others.

Special attention is given to classical algorithms: depth-first search (DFS), breadth-first search (BFS), Dijkstra’s algorithm, Prim’s algorithm, Kruskal’s algorithm, and Floyd–Warshall algorithm. Their computational complexity, efficiency, and applicability to real-world tasks are analyzed. Modern approaches, including heuristic and numerical methods, are also examined.

The practical part includes examples of solving problems using graph structures, particularly in logistics, routing, and social network analysis. Modeling and analysis of the efficiency of selected algorithms are provided.

The research results can be used in teaching discrete mathematics, algorithms, and programming in schools, colleges, and higher education institutions.

The total length of the thesis is 35 pages, with the main text comprising 28 pages.

Keywords: graph, vertex, edge, algorithm, graph problem, shortest path, spanning tree, graph theory.

ЗМІСТ

ЗАВДАННЯ………………………………………………………………………………….. 2

АНОТАЦІЯ……………………………………………………………………………….. 3

ANNOTATION…………………………………………………………………………. 4

ВСТУП…………………………………..……………………………………………… 6

РОЗДІЛ 1. ТЕОРЕТИЧНІ ВІДОМОСТІ ПО ТЕМІ ДОСЛІДЖЕННЯ……………………………. 8

1.1 Предмет теорії графів. Основні означення……………………………………………… 8

1.2 Класифікація графів та їх спеціальні види ……………………………………....... 9

Висновки до розділу 1…………………………………………………..……………. 11

РОЗДІЛ 2. ПРАКТИЧНЕ ЗАСТОСУВАННЯ ТА ПІДХОДИ ДО РОЗВ'ЯЗАННЯ ЗАДАЧ У ГАЛУЗІ ТЕОРІЇ ГРАФІВ……………………………………………………...…. 12

2.1 Приклади застосування теорії графів………………………………………………... 12

2.2 Практичні задачі та їх розв'язання…………………………………………………... 13

2.3 Приклади графових моделей………………………………………………………… 16

Висновки до розділу 2…………………………………………………………........ 20

РОЗДІЛ 3. ЗАДАЧІ НА ГРАФАХ……………………………………………………… 21

3.1 Динамічні процеси на графах……………………………………………………... 22

3.2 Комбінаторні задачі та симетрії в графах…………………….. ……………………… 25

3.3 Каркасні многочлени та властивості графів……………………………………… 29

Висновки до розділу 3………………………………………………………………. 33

ВИСНОВКИ……………………………………………………………………………. 34

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ……………………………………… 35

ВСТУП

Актуальність дослідження. Теорія графів є важливим розділом дискретної математики, що має широкий спектр застосувань у сучасних технологіях, зокрема в математичній кібернетиці, комп'ютерних науках, програмуванні, а також у проектуванні та управлінні інформаційними системами. Це навчальне поле охоплює основні поняття, ідеї та методи, які сприяють розвитку логічного та аналітичного мислення у студентів.

Сучасний розвиток електронних обчислювальних машин стимулює зростання ролі дискретних задач, серед яких графи займають одне з провідних місць завдяки своїй здатності наочно моделювати різноманітні об'єкти і процеси. Графи застосовуються для аналізу електричних і транспортних мереж, комп'ютерних систем, картографічних моделей, а також у теорії ігор, хімії та багатьох інших наукових галузях.

Теорія графів тісно пов'язана з іншими математичними дисциплінами, такими як теорія множин, теорія матриць, математична логіка та теорія ймовірностей. Це забезпечує глибоке і комплексне розуміння математичних об'єктів, що вивчаються. Знання основ теорії множин і комбінаторного аналізу, отримані під час вивчення вищої математики, є підґрунтям для ефективного освоєння теми теорії графів.

Метою роботи – є вивчення основних понять і властивостей теорії графів, а також аналіз їх застосування в різних сферах науки та техніки.

Для досягнення цієї мети поставлено такі основні **завдання:**

1. Огляд базових понять теорії графів, включаючи визначення графів, вершин, ребер, а також видів графів.
2. Аналіз основних властивостей графів, таких як зв'язність, циклічність, деревоподібність.
3. Розгляд алгоритмів, пов'язаних із графами, таких як алгоритм Дейкстри, алгоритм Флойда-Уоршелла.
4. Вивчення застосування теорії графів у різних областях, таких як комп'ютерні мережі, транспортні системи, біоінформатика.
5. Розробка практичних завдань для закріплення знань з теми теорії графів

Об'єктом дипломної роботи є теорія графів як важливий розділ дискретної математики.

Предметом дипломної роботи є основні поняття, властивості та методи дослідження графів, а також їх застосування у практичній діяльності.

Методи дослідження. Теоретичний аналіз літературних джерел з теорії графів, що дозволяє ознайомитися з основними концепціями та поняттями теорії графів;

Структура роботи. Дипломна робота складається зі змісту, вступу, трьох розділів, підрозділів, висновків та списку використаних джерел.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ВІДОМОСТІ ПО ТЕМІ ДОСЛІДЖЕННЯ

1.1 Предмет теорії графів. Основні означення.

Теорія графів у математиці досліджує особливий вид математичних структур – графи, які використовуються для моделювання взаємозв'язків між парами об'єктів. Графи складаються з вершин (точок), з'єднаних між собою ребрами (лініями). Завдяки здатності множини вершин абстрагувати будь-які комп'ютерні дані, теорія графів детально вивчає їхні взаємозв'язки та пропонує рішення для задач, що стосуються розташування, оптимізації, налаштування мереж і багатьох інших аспектів [5].

Графи є засобом візуалізації даних та зв'язків між ними, що дозволяє наочно представляти складну чи об'ємну інформацію, яку важко описати текстом або алгоритмами. Вони забезпечують ефективність використання даних завдяки чіткому та правильному їх поданню. Основними поняттями теорії графів є структури графів, вершини, ребра, орієнтовані графи, напівграфи, ваговані графи та ізоморфізм графів.

Основні складові графів. Граф є абстрактним математичним об'єктом, що включає множину вершин і множину ребер, які їх з'єднують. Вершини позначають об'єкти, а ребра – зв'язки між ними. Різноманітність форм і характеристик графів дозволяє моделювати широкий спектр ситуацій.

Вершини та ребра. Вершини графа подані точками, а ребра – лініями, які їх з'єднують. Ребра можуть мати напрямки та бути без напрямку, а їх кількість між вершинами варіюється залежно від типу графа.

Орієнтовані графи. У таких графах кожне ребро має визначений напрямок, що вказує на початкову та кінцеву вершини. Орієнтовані графи широко застосовуються для моделювання процесів і напрямлених зв'язків.

Напівграфи. Особливістю напівграфів є можливість з'єднання ребрами більше ніж двох вершин. Вони використовуються для опису складніших зв'язків, що виходять за межі парних взаємодій.

Ваговані графи. У таких графах кожному ребру присвоюється числове значення, яке відображає вагу або вартість зв'язку між вершинами. Такі графи застосовуються у випадках, коли важливо врахувати значущість чи силу зв'язків.

Ізоморфізм графів. Ізоморфізм означає, що два графи мають однакову структуру, хоча вершини та ребра можуть мати різні позначення. Такі графи вважаються структурно еквівалентними, що дозволяє використовувати їх для аналізу схожих систем [1].

1.2 Класифікація графів та їх спеціальні види.

Графи виступають однією з фундаментальних структур у математиці та інформатиці. Їх класифікація дозволяє систематизувати різноманіття графів, виділяючи специфічні властивості, що робить їх корисними для різних сфер застосування [6].

Основні типи класифікацій графів включають такі категорії:

1. За типом вершин і ребер:

- Простий граф: не містить петель (ребер, які з'єднують вершину із собою) або кратних ребер. Використовується для моделювання простих зв'язків між об'єктами.
- Мультиграф: допускає кратні ребра між одними і тими ж вершинами, що корисно для моделювання багаторазових зв'язків.
- Псевдограф: містить і петлі, і кратні ребра, що дає змогу описувати зв'язки об'єкта із самим собою.

2. За напрямком ребер:

- Неорієнтовані графи: ребра не мають напрямку, і шлях можна пройти в будь-якому напрямку.
- Орієнтовані графи (орграфи): кожне ребро має напрямок, що визначає порядок проходження між вершинами.
- Змішані графи: містять одночасно орієнтовані та неорієнтовані ребра.

3. За наявністю ваг:

- Вагові (зважені) графи: кожне ребро має числову вагу, що відображає його значення (наприклад, відстань чи вартість).
- Невагові графи: всі ребра однакові за значенням або не мають ваг.

4. За зв'язністю:

- Зв'язні графи: між будь-якими вершинами існує шлях.
- Незв'язні графи: містять вершини чи їх групи, які не з'єднані між собою.
- Сильно зв'язні графи (для орієнтованих графів): існує шлях між будь-якими двома вершинами в обидва напрямки.
- Слабко зв'язні графи (для орієнтованих графів): зв'язність забезпечується, якщо розглядати всі ребра як неорієнтовані.

5. За циклічністю:

- Ациклічні графи: не містять жодних циклів.

- Циклічні графи: включають хоча б один цикл.

6. Спеціальні види графів:

- Дерева: ациклічні зв’язні графи, де існує єдиний шлях між будь-якими двома вершинами.
  - *Кореневе дерево*: має особливу вершину-корінь.
  - *Бінарне дерево*: вершина має не більше двох дочірніх вершин.
- Ліс: набір незв’язних дерев.
- Повний граф: кожна пара вершин з’єднана ребром.
- Біпартиний граф: вершини поділені на дві групи, де кожне ребро з’єднує вершини різних груп.
  - *Повний біпартиний граф*: кожна вершина однієї групи з’єднана з усіма вершинами іншої.
- Ейлерів граф: містить цикл, що проходить через кожне ребро рівно один раз (усі вершини мають парний ступінь).
- Планарний граф: його можна намалювати на площині без перетину ребер.
- Гамільтонів граф: має цикл, що проходить через кожну вершину рівно один раз.

7. Матричне представлення графів:

- Матриця суміжності: квадратна матриця, що відображає наявність (або вагу) ребер між вершинами.
- Матриця інцидентності: показує зв’язки між вершинами й ребрами.

8. Інші види графів:

- Кубічний граф: кожна вершина має степінь 3.
- Граф-цикл: замкнений шлях, що моделює повторювані процеси.
- Колесний граф: утворений шляхом з’єднання циклу з однією центральною вершиною.
- Зірковий граф: одна центральна вершина з’єднана з усіма іншими.
- Гіперкуб: граф, вершини якого відповідають двійковим векторним комбінаціям [3].

Висновок до розділу 1

У розділі розглянуто теоретичні основи теми дослідження, що стосуються предмета теорії графів, її основних понять і класифікації графів. Теорія графів є важливою галуззю математики та інформатики, яка займається вивченням структур, що моделюють зв’язки між об’єктами. Основними складовими графів є вершини та ребра, які можуть мати різні властивості залежно від типу графа.

Проведено огляд класифікації графів за такими критеріями, як типи вершин і ребер, наявність напрямку чи ваги, рівень зв’язності, циклічність і спеціальні властивості. Зокрема, виділено такі ключові поняття, як прості графи, орієнтовані та неорієнтовані графи, вагові й невагові графи, зв’язні й незв’язні графи, а також спеціальні типи графів, зокрема дерева, планарні, ейлерові й гамільтові графи.

Розуміння означень і класифікації графів є необхідною основою для їх практичного застосування в різних галузях, включаючи розв’язання задач оптимізації, моделювання мереж і аналіз структур даних. Отримані знання формують фундамент для подальшого дослідження і розробки методів роботи з графами.

РОЗДІЛ 2. ПРАКТИЧНЕ ЗАСТОСУВАННЯ ТА ПІДХОДИ ДО РОЗВ’ЯЗАННЯ ЗАДАЧ У ГАЛУЗІ ТЕОРІЇ ГРАФІВ

2.1 Приклади застосування теорії графів.

Сучасний стрімкий розвиток науки й технологій сприяє постійному дослідженню актуальних проблем, що постають перед людством. Часто ефективне вирішення завдань можливе завдяки використанню графів, які дозволяють формалізувати проблему, а отримані результати інтерпретувати у необхідних вихідних термінах.

Теорія графів сьогодні є простим, доступним і потужним інструментом для розв’язання широкого спектра завдань у різних сферах. Зокрема, графи використовуються для моделювання дорожніх схем, електричних ланцюгів, архітектурних задач, географічних карт, молекул хімічних сполук, а також зв’язків між людьми чи групами. Вони знаходять застосування у проектуванні інтегральних схем, управлінських процесах, дослідженні автоматів, логічних ланцюгів, блок-схем програм, а також в економіці, статистиці, хімії, біології та теорії розкладів.

У математичній сфері графи служать для розв’язання геометричних та прикладних задач. Вони є ключем для розв’язання багатьох комп’ютерних проблем, що забезпечують можливість сучасної комунікації та технологічних процесів. Таким чином, графи сприяють інтеграції математичних методів у сучасну науку та техніку [8]. Теорія графів є розділом математики, що вивчає множини з заданими на них відношеннями між елементами. Такі об’єкти можуть бути візуалізовані як геометричні конфігурації, що складаються з точок, кіл чи інших фігур, з’єднаних лініями. Точки в таких моделях відповідають елементам множини, а лінії – зв’язкам або відношенням між ними.

Хоча такі зображення називають графами, сам термін має набагато ширше значення, а рисунок є лише одним із способів представлення графа. Відмінність графів від геометричних конфігурацій, які також складаються з точок і ліній, полягає в тому, що в графах не має значення відстань між точками, форма ліній або кути між ними. Це робить граф топологічним об’єктом – таким, чиї властивості залишаються незмінними під час розтягування, стиснення чи викривлення [12].

Однією з найвідоміших задач теорії графів є знаменита топологічна проблема чотирьох фарб, що виникла під час створення географічних карт. Ця гіпотеза стверджує, що будь-яку карту можна розфарбувати максимум чотирма кольорами так, щоб суміжні області мали різні кольори. Проблема належить до теорії графів, адже кожна карта утворює граф, у якому країни (включаючи зовнішню область) відповідають вершинам, а ребра з'єднують вершини, якщо відповідні країни мають спільний кордон. Такий граф можна побудувати на площині без перетину ребер, якщо точки перетину не є вершинами графа.

Теорія графів також активно застосовується в хімії для аналізу та моделювання хімічних і хіміко-технологічних сполук. Графи, що використовуються в цій галузі, акцентують увагу на характері зв'язків між вершинами. Вершини представляють атоми, а ребра – хімічні зв'язки між ними. Наприклад, молекулярні графи, що є неорієнтованими, моделюють будову молекул і знаходять застосування у стереохімії, хімії кластерів і полімерів.

Теорія графів знаходить широке використання й у психології. У 1936 році Курт Левін використав планарну карту для зображення життєвого середовища індивіда, що фактично відповідає графовому підходу. Аналіз цієї моделі з погляду теорії графів дозволив психологам розробити інтерпретацію графа, де вершини представляють людей, а ребра – взаємини між ними. Такі графові моделі стали основою для вивчення групової динаміки та соціальних зв'язків.

2.2 Практичні задачі та їх розв'язання.

Алгоритм Флойда – Уоршелла. Якщо потрібно визначити найкоротші відстані між усіма вершинами, то для розв'язування цієї задачі можна n разів повторити алгоритм Дейкстри, приймаючи по черзі кожну вершину за стартову.

Ці дії потребують виконання $O(n3)$ операцій. Однак, для розв'язування цієї задачі є більш економний (по порядку також $O(n3)$, але в константу разів економніший) алгоритм Флойда – Уоршелла:

$$For\ k:=\ 1\ to\ n\ do$$

$$For\ i:=\ 1\ to\ n\ do$$

$$For\ j:=\ 1\ to\ n\ do$$

$$d[i,j]:=min(d[i,j],d[i,k]\ +\ d[k,j]);$$

Тут:

$d[i,j]$ – спочатку довжина дуги $[i,j]$, а в кінці – довжина найкоротшого шляху.

Застосування: У місті є система доріг, що з'єднує кілька районів. Дороги мають різну довжину, і між деякими районами може бути кілька маршрутів через інші райони. Міська адміністрація хоче знайти мінімальну відстань між будь-якими двома районами, щоб спланувати оптимальний рух транспорту.

Вхідні дані: Місто розбито на N районів (номери районів: $1, 2, \ldots, N$). Є M доріг між районами, кожна з яких описується трьома числами A, B, W:

- A та B – номери районів, які з’єднує дорога.
- W – довжина дороги (відстань).

Якщо дороги між районами немає, вважати відстань нескінченністю.

Розв’язання: N – кількість вершин (районів).

M – кількість ребер (доріг).

Граф представлений у вигляді матриці суміжності D, де:

- $D[i][j] \ = \ W$ – довжина дороги між вершинами i та j.
- $D[i][j] \ = \ \infty$, якщо дороги між i та j немає.
- $D[i][i] \ = \ 0$ (довжина від вершини до себе).

$(1,2,5), (2,3,3)(2, 3, 3)(2,3,3), (1,4,10)(1, 4, 10)(1,4,10),$ $(4,3,1)(4, 3, 1)(4,3,1), (2,4,2)(2, 4, 2)(2,4,2)$, матриця суміжності виглядає так:

$$D = \begin{bmatrix} 0 & 5 & \infty & 10 \\ \infty & 0 & 3 & 2 \\ \infty & \infty & 0 & \infty \\ \infty & \infty & 1 & 0 \end{bmatrix}$$

Алгоритм працює у три вкладені цикли:

1. Для кожної проміжної вершини k $(1, 2, \ldots, N)$:
   - ○ Для кожної пари вершин i і j:
     - ✦ Оновлюємо відстань $D[i][j]$:

$$D[i][j] \ = \ min(D[i][j], D[i][k] \ + \ D[k][j])$$

Це перевіряє, чи коротший шлях між i та j проходить через k.

1. Для $k \ = \ 1$:

   Перевіряємо всі шляхи через вершину 1.

   Наприклад, D[2][3] залишається 3, бо шлях через 1 довший.

   Зміни у матриці не відбуваються.

2. Для $k \ = \ 2$:

   Перевіряємо шляхи через вершину 2:

$D[1][3] = min(\infty, D[1][2] + D[2][3]) = min(\infty, 5 + 3) = 8$

$D[1][4]$ залишається 7 (через 2 вже найкоротший).

3. Для $k = 3$:

   Перевіряємо через вершину 3.

   Шляхів, що покращуються, немає.

4. Для $k = 4$:

   Перевіряємо через вершину 4:

   Шляхи залишаються оптимальними.

Остаточна матриця найкоротших відстаней:

$$D = \begin{bmatrix} 0 & 5 & 8 & 7 \\ \infty & 0 & 3 & 2 \\ \infty & \infty & 0 & \infty \\ \infty & \infty & 1 & 0 \end{bmatrix}$$

Висновок:

1. Найкоротший шлях між будь-якими двома районами визначено у матриці D.
2. Наприклад, відстань між районами 1 і 3 – це $D[1][3] = 8$.
3. Якщо $D[i][j] = \infty$, то вершини i і j недосяжні.

Алгоритм Дейкстри. Дано орієнтований граф із вершинами A, B, C, D, E та такими вагами ребер:

- $A \to B = 1$
- $A \to C = 4$
- $B \to C = 2$
- $B \to D = 6$
- $C \to D = 3$
- $D \to E = 1$

Необхідно знайти найкоротші шляхи від вершини A до всіх інших вершин графа.

Початкові відстані до вершин: $A: 0, B: \infty, C: \infty, D: \infty, E: \infty$

Обираємо вершину з мінімальною відстанню серед невідвіданих. Це A, відстань 0. Оновлюємо відстані до сусідів B та C:

$$A \to B: 0 + 1 = 1$$

$$A \rightarrow C: 0 + 4 = 4$$

Отримуємо нові відстані: $A: 0, B: 1, C: 4, D: \infty, E: \infty$

Обираємо B (відстань 1). Оновлюємо відстані до сусідів C та D:

$B \rightarrow C: 1 + 2 = 3$ (менше за 4, тому оновлюємо)

$$B \rightarrow D: 1 + 6 = 7$$

Отримуємо: $A: 0, B: 1, C: 3, D: 7, E: \infty$

$C \rightarrow D: 3 + 3 = 6$ (менше за 7, тому оновлюємо)

$$A: 0, B: 1, C: 3, D: 6, E: \infty$$

$$D \rightarrow E: 6 + 1 = 7$$

$$A: 0, B: 1, C: 3, D: 6, E: 7$$

Обираємо E (відстань 7). Усі вершини відвідані, роботу завершено.

Найкоротші відстані від вершини A до всіх інших:

- $A \rightarrow B = 1$
- $A \rightarrow C = 3$
- $A \rightarrow D = 6$
- $A \rightarrow E = 7$

2.3 Приклади графових моделей

Графова модель, або імовірнісна графова модель (ІГМ) (*англ. probabilistic graphical model, PGM*), – це ймовірнісна модель, у якій умовні залежності між випадковими змінними подано у вигляді графа. Ці моделі широко застосовуються в теорії ймовірностей, статистиці (зокрема, баєсовій), а також у машинному навчанні.

Імовірнісні графові моделі слугують для кодування повного розподілу ймовірностей у багатовимірному просторі, використовуючи представлення, засноване на графах. Граф у такій моделі є компактним або факторним представленням незалежностей, властивих певному розподілу.

Найбільш поширеними є дві галузі графових представлень розподілів:

баєсові мережі та марковські мережі.

- Баєсові мережі будуються на орієнтованих ациклічних графах і використовуються для кодування спрямованих умовних залежностей.
- Марковські мережі, своєю чергою, ґрунтуються на неорієнтованих графах і описують залежності у формі симетричних зв’язків.

Обидва типи графових моделей дозволяють ефективно працювати з властивостями розкладу й незалежностей, хоча вони відрізняються у тому, які залежності можуть бути закодовані та як розподіли факторизуються.

Баєсова мережа. Якщо мережеву структуру моделі представлено як орієнтований ациклічний граф, то ця модель представляє розклад спільної ймовірності всіх випадкових змінних. Точніше, якщо подіями є $X1, \ldots , Xn$, то спільна ймовірність задовольняє

$$P[X_1, \ldots, X_n] = \prod_{i=1}^{n} P[X_i | pa_i]$$

де єpa_i набором батьків вершини X_i . Іншими словами, спільний розподіл розкладається у добуток умовних розподілів. Наприклад, зображена вище статистична модель (що насправді є не орієнтованим ациклічним, а родовим графом) складається з випадкових змінних A, B, C, D з густиною спільного розподілу ймовірності, що розкладається як

$$P[A, B, C, D] = P[A] \cdot P[B] \cdot P[C|B, D] \cdot P[D|A, B, C].$$

Будь-які дві вершини в баєсових мережах вважаються умовно незалежними, якщо відомі значення їхніх батьків. Загалом, будь-які дві множини вершин є умовно незалежними щодо третьої множини, за умови, що граф задовольняє критерій о-розділеності. У баєсових мережах існує еквівалентність між локальною та глобальною незалежностями.

Цей тип графової моделі називається орієнтованою графовою моделлю, також відомою як баєсова мережа або мережа переконань. Багато класичних методів машинного навчання, таких як приховані марковські моделі і нейронні мережі, а також сучасніші підходи, наприклад, марковські моделі зі змінним порядком, є окремими випадками баєсових мереж.

Марковське випадкове поле. Марковське випадкове поле, також відоме як марковська мережа, являє собою модель на основі неорієнтованого графа. Графічна модель, що містить багато повторюваних підблоків, може бути зображена за допомогою пластинного позначення.

У послідовному моделюванні, граф, який становить інтерес, зазвичай є ланцюговим. Входова послідовність спостережуваних змінних XX представляє послідовність спостережень, а YY представляє приховану (або невідому) змінну стану, висновки про яку потрібно отримувати зі спостережень. YY_i структурують так, щоби утворити ланцюг, з ребрами між кожними Y_{i-1} та Y_i. Маючи просте представлення Y_i як «міток» для кожного з елементів послідовності входу, це компонування також уможливлює дієві алгоритми для:

- *тренування* моделі, навчання умовних розподілів між Y_i та функціями ознак для деякого корпусу тренувальних даних.
- *декодування*, визначення ймовірності заданої послідовності міток Y за заданої X .
- *висновування*, визначення *найправдоподібнішої* послідовності міток X за заданої Y .

Умовну залежність кожної з Y_i від визначають через фіксований набір *функцій ознак* вигляду $f(i, Y_{i-1}, Y_i, X)$, які можливо розглядати як вимірювання на послідовності входу, що частково визначають правдоподібність кожного з можливих значень Y_i. Ця модель призначує кожній ознаці числову вагу, й поєднує їх для визначення ймовірності певного значення Y_i.

Факторний граф – це неорієнтований двочастковий граф, який пов’язує змінні та фактори. Кожен фактор відповідає функції, визначеній на змінних, з якими він пов’язаний. Таке представлення особливо корисне для аналізу та реалізації алгоритмів поширення переконань.

Формально, фактор-граф є фактор-об’єктом у категорії графів. Категорія графів є конкретизованою: відображення графа в множину його вершин робить її конкретною категорією, завдяки чому об’єкти можна інтерпретувати як «множини з додатковою структурою». Фактор-граф відповідає графу, утвореному на фактор-множині множини вершин V.

Також існує гомоморфізм графів (фактор-відображення), який переводить кожну вершину або ребро у відповідний клас еквівалентності. Інтуїтивно це відповідає «склеюванню» (або ототожненню) вершин і ребер графа.

Граф тривіально є фактор-графом самого себе, якщо кожна вершина розглядається окремо. З іншого боку, граф, що складається з єдиної точки, є фактор-графом будь-якого непорожнього графа, якщо розбиття містить лише одну групу з усіх вершин. Найпростіший нетривіальний фактор-граф отримується шляхом «склеювання» двох вершин (їх ототожнення). Якщо ці вершини з’єднані ребром, це називається стягуванням ребра.

Дерево клік (або дерево злук) – це дерево, що складається з клік графа і використовується в алгоритмі дерева злук.

Інтуїтивно, деревна декомпозиція представляє вершини графа G як піддерева дерева, де вершини графа суміжні тільки тоді, коли відповідні піддерева перетинаються. У результаті G стає підграфом графа перетинів цих піддерев. Повний граф перетинів є хордальним графом. Кожне дерево пов’язує вершини графа з вузлами дерева. Формально кожен вузол дерева визначається множиною вершин, які з ним асоціюються.

Тоді для заданого графа $G = (V, E)$ деревна декомпозиція — це пара (X, T), де $X = \{X_1, \ldots, Xn\}$ є сімейством підмножин множини V, а T є деревом, вузлами якого служать підмножини X_i, які задовольняють таким властивостям [9]:

1. Об'єднання всіх множин X_i дорівнює V. Таким чином, будь-яка вершина графа пов'язана хоча б з одним вузлом дерева.

2. Для будь-якого ребра (v, w) графа існує підмножина X_i, що містить як v, так і w. Таким чином, вершини суміжні в графі, тільки якщо вони відповідають піддеревам, що мають спільний вузол.

3. Якщо X_i і X_j містять вершину v, то всі вузли Xk дерева в (унікальному) шляху між X_i і X_j містять v теж. Тобто вузли, пов'язані з вершиною v, утворюють зв'язну підмножину в T.

Ланцюговий граф – це граф, який може містити як орієнтовані, так і неорієнтовані ребра, але не має жодних орієнтованих циклів. Іншими словами, якщо почати з будь-якої вершини і рухатись за напрямками стрілок, то неможливо повернутися до початкової вершини, якщо пройти орієнтоване ребро.

Орієнтовані ациклічні графи та неорієнтовані графи є спеціальними випадками ланцюгових графів. Це дозволяє використовувати ланцюгові графи для уніфікації та узагальнення структур баєсових і марковських мереж [10].

Родовий граф є розширеним типом графа, що може включати орієнтовані, біорієнтовані та неорієнтовані ребра.

Умовне випадкове поле (CRF) – це розрізнювальна модель, яка визначається на неорієнтованому графі.

Обмежена машина Больцмана (RBM) є двочастковою породжувальною моделлю, яка також визначається на неорієнтованому графі.

Система моделей, яка пропонує алгоритми для виявлення та аналізу структур складних розподілів, дає змогу стисло описувати ці розподіли, витягувати неструктуровану інформацію та ефективно будувати й використовувати їх [9]. Застосування графових моделей включають витягування інформації, розпізнавання мовлення, комп'ютерний зір, декодування кодів з малою щільністю перевірок на парність, моделювання генних регуляторних мереж, пошуку генів та діагностування захворювань, та графові моделі структури білка.

Висновок до розділу 2

У цьому розділі досліджено практичне значення теорії графів, наведено приклади її широкого застосування та розглянуто алгоритмічні підходи до розв’язання задач у різних сферах науки й техніки.

Теорія графів є потужним інструментом моделювання систем і процесів, які можна подати у вигляді множин елементів та зв’язків між ними. Це робить її незамінною для вирішення задач у таких галузях, як інформатика, хімія, економіка, психологія, транспортна логістика, біологія, соціальні науки та багато інших. Зокрема, моделі на основі графів дозволяють аналізувати дорожні мережі, оптимізувати маршрути, досліджувати хімічні сполуки, проектувати електронні схеми та розуміти міжособистісні зв’язки.

У розділі представлено алгоритми Флойда–Уоршелла та Дейкстри, які є основними інструментами для розв’язання задач пошуку найкоротших шляхів у графах. Було проаналізовано їхні особливості, ефективність та сферу застосування. Наприклад, алгоритм Флойда–Уоршелла ефективно вирішує задачі пошуку найкоротших шляхів між усіма вершинами графа, що ілюструє його зручність у задачах міського планування. Алгоритм Дейкстри, своєю чергою, оптимально вирішує проблему пошуку найкоротших шляхів від заданої вершини до всіх інших, що демонструє його доцільність у навігаційних системах.

Також висвітлено графові моделі як базовий інструмент у сучасній обчислювальній техніці. Застосування баєсових мереж, марковських випадкових полів та інших графових структур дозволяє моделювати складні системи та ухвалювати оптимальні рішення у різних наукових і технічних сферах.

Таким чином, теорія графів та її практичне застосування підтверджують свою важливість у вирішенні реальних задач, забезпечуючи ефективність, універсальність та точність у дослідженні складних систем. Це підкреслює необхідність подальшого вивчення та розвитку методів, заснованих на графових структурах, у різних галузях науки і техніки.

РОЗДІЛ 3. ЗАДАЧІ НА ГРАФАХ

У цьому розділі буде розглянуто задачі, які моделюються за допомогою графів. Кожна з таких задач має цікаві математичні властивості та вимагає нетривіального аналізу. Розділ містить три підрозділи, у кожному з яких зібрано по кілька задач, що ілюструють певні особливості графових структур, алгоритмів або комбінаційних властивостей.

Задача 1: Міста деякої країни з`єднані дорогами, таким чином, що від будь-якого міста можна добратися до іншого міста *не проходячи через транзитне місто двічі. Причому зробити це можна тільки єдиним способом.* В кожному місті живуть мешканці, які мають свою релігію. Кожне місто поклоняється одному з трьох тотемів: вода, вогонь або повітря. Відомо, що вогонь сильніший ніж вода, вода сильніша ніж повітря, а повітря сильніше за вогонь. Кожен народ бажає, щоб їх ідол був не слабшим ніж в сусіда. З цією метою кожного полудня рівно в 12:00 кожен народ дивиться на своїх сусідів. Якщо встановлює, що в сусіда є сильніший тотем, то змінює свою релігію на сильнішу і починають поклонятися більш сильному ідолу. Чи правда, що рано чи пізно всі племена почнуть поклонятися в одному і тому ж тотему?

Задача 2: Грані куба мають розмір $5 \times 5 \times 5$ розграфлені на клітинки з стороною 1. Кожну клітинку пофарбували в синій, жовтий або зелений колір так, що клітинки, які мають спільну сторону пофарбовані в різні кольори. Яка найменша кількість синіх клітинок може бути?

Задача 3: Маємо дяку кількість міст. В кожне місто йде три дороги: синя, жовта і бузкова. В залежності від кольорів, які заходять по дорогах в місто (фіксуємо за годинниковою стрілкою) поділяємо міста на два типи СЖБ і СБЖ. Доведіть, що різниця кількості міст різних типів ділиться на чотири.

Задача 4: У графі степені всіх вершин дорівнюють трьом і між будь-якими двома вершинами є шлях довжиною не більше двох, яку найбільшу кількість вершин може мати цей граф?

Задача 5: У країні є 20 міст. Між будь-якими двома містами є дорога автодорога може закрити на ремонт будь-яку дорогу з чотирьох, які утворюють циклічний маршрут. Чи може після деяких таких операцій залишитись відкритими тільки 19 доріг.

Задача 6: Нехай задано граф $G(V, E)$ який задано на n вершинах. Покладемо кожній вершині v у відповідність деяку змінну x_v. Нехай T - це множина всіх каркасних дерев, які містяться в графі G, тобто піддерев, які містять всі вершини графа G. Розглянемо каркасний многочлен від n змінних $x_1, x_2, x_3 \dots x_n$ який будується наступним чином: $P_G(x_1, x_2, \dots, x_n) = \sum_{S \in T} \prod_{v \in V} x_v^{\deg_S v - 1}$

Назвемо зв'язний граф *хорошим* якщо формула розкладається на лінійні множники зокрема якщо P_G дорівнює нулю інші графи назвемо *поганими.*

Завдання знайти:

1) $P_{K_4}(x_1, x_2, x_3, x_4)$ де K_4 повний граф на чотирьох вершинах;
2) Доведіть, що цикл C_5 на п'яти вершинах є поганим графом;
3) Нехай G хороший граф U деяка підмножина його вершин, граф H складається з усіх вершин які лежать в U і всіх ребер графа G, які з'єднують ці вершини. Доведіть що H також хороший.
4) Назвем роздвоєнням вершини v операцію, яка додає в граф вершину v' яка з'єднується рівно з тими ж вершинами, що й вершина v. Доведіть, що граф, який ми отримаємо з однієї вершини операцією додавання висячої вершини, роздвоєння вершини v з добавленням ребра vv' чи з роздвоєння вершини без додавання ребра vv' є хорошим графом.

3.1 Динамічні процеси на графах.

Динамічні процеси на графах – це процеси, що розгортаються у часі та залежать від структури графа, зокрема — поширення інформації, епідемій, вогню, сигналів чи впливів між вершинами [2].

У цьому підрозділі вводяться основні моделі динамічних процесів, зокрема:

- Процеси поширення (diffusion processes) – з прикладами моделей SI, SIR, SIS, що широко застосовуються для моделювання епідеміологічних ситуацій.
- Моделі каскадного поширення – наприклад, незалежні каскади (Independent Cascade Model) та лінійні порогові моделі (Linear Threshold Model), які моделюють динаміку впливу у соціальних мережах.
- Динаміка на марковських графах – опис переходів між станами, які залежать від структури графа та ймовірнісних правил.

Також обговорюються характеристики графів, що впливають на динамічні процеси: ступінь вершини, центральність, щільність, кластеризація, спектральні властивості тощо.

Особлива увага приділяється математичному опису моделей за допомогою систем диференціальних або різницевих рівнянь, а також використанню матричного представлення графів (матриця суміжності, лапласіан тощо) для аналітичного дослідження поведінки процесів [2].

Задача 1. Єдина зміна тотемів. Нехай задано зв'язний ациклічний граф, у якому між будь-якою парою міст (вершин) існує лише один шлях. Це означає, що граф є деревом. Кожна вершина має початкову релігію (тотем) – воду, вогонь або повітря. Задається цикл переваги: вогонь $>$ вода $>$ повітря $>$ вогонь. Кожного дня о 12:00 кожна вершина аналізує своїх сусідів: якщо в сусіда тотем сильніший, то вона переходить до цього тотему.

Цей процес є дискретною динамічною системою на дереві з циклічно визначеною перевагою. Оскільки цикл переваги не є транзитивним і в кожного вузла є обмежена кількість сусідів (граф – дерево), процес не може стабілізуватись у єдиному стані для всіх вершин.

В умові сказано, що ∃ єдиний шлях від n_i та n_j (без повторень) $=>$ наш граф є деревом, тобто має листки.

Нехай А – листок, B – її батьківська вершина, якщо A i B однакові тотеми, то граф має цей тотем у всіх вершинах (з міркувань індукції).

Розглянемо випадок коли в B сильніший тотем, тоді А поміняє свій і знову всі вершини мають один тотем.

Нехай В не змінює свого ідола на тотем вершини А, тобто або в В сильніший, або в А і в В рівні тотеми. В кожному випадку тотеми А і В стануть рівні, за індукцією можна довести, що весь граф матиме один і той самий тотем.

Переформовуємо задачу мовою теорії графів. Кожному племені покладемо у відповідність вершину, між кожними двома сусідніми племенами є ребро. Для кожної вершини визначено тотем, якщо тотем сусідньої вершини сильніший, то в цей день вершина змінює свого ідола на більш сильнішого.

Розглянемо за індукцією:

$n = 1$ – всі тотеми вершин однакові, припустимо для деякого n твердження «всі тотеми однакові» істинне.

Доведемо, що твердження істинне для $(n = 1)$: Виберу листок А для якої батьківською вершиною є вершина В.

Кажемо, що вершина X діє на вершину Y якщо X і Y сусідні (з'єднані ребром) і тотем X сильніший ніж тотем Y і при цьому Y не має сусіда з більш сильним ідолом ніж в X.

Якщо немає дня, в якому А впливає на В. Тоді в графі, який ми отримаємо з даного видалення вершини А за припущенням індукції всі вершини будуть рівні, як наслідок його матиме вершина B, а значить його прийме і плем'я вершини A. Тепер якщо є такий день, коли вершини A вплинуло на B. після цього дня вершини A і B мають однакового ідола. Якщо вершина B змінює тотем на сильніший, то на наступний день A також може поклонятися ідолу B, а це означає, що не буде дня, в якому A отримає сильніший тотем ніж є зараз. Як наслідок A більше ніколи не впливатиме на B. Повторюючи попередні міркування, ми отримаємо, що всі племена будуть поклонятися одному ідолу.

Задача 4. Максимальний граф із діаметром 2 та степенем 3. У графі степінь усіх вершин дорівнює трьом, і відстань між будь-якими двома вершинами – не більше двох. Це означає, що граф має діаметр 2, а степінь вершин – сталий.

Потрібно знайти найбільший можливий розмір такого графа. Відомо, що такий граф – це приклад регулярного графа з діаметром 2. Скористаємося теоремою Мура, що встановлює межу кількості вершин таких графів. Для графа з n вершинами, степенем $d = 3$ та діаметром $D = 2$, межа Мура – $n \leq 1 + d +$
$d(d-1) = 1 + 3 + 6 = 10$.

Нехай задано граф без петель та кратних ребер, у якому:

- степінь кожної вершини дорівнює 3, тобто граф є регулярним степеня 3;
- між будь-якими двома вершинами існує шлях довжиною не більше 2, тобто діаметр графа не перевищує 2.

Потрібно визначити максимально можливу кількість вершин такого графа.

Нехай граф має n вершин. Кожна вершина має рівно 3 сусіди, тобто кількість ребер:

$$|E| = \frac{3n}{2}$$

Оскільки діаметр графа дорівнює 2, то будь-яка пара вершин або є безпосередніми сусідами (відстань 1), або з'єднані через спільного сусіда (відстань 2). Іншими словами, відстань між будь-якими двома вершинами не перевищує 2.

Оцінка за допомогою обмеження на кількість вершин на відстані ≤ 2 Нехай $v \in V$– довільна вершина графа. Вона має:

- 3 сусіди (на відстані 1),
- кожен із цих 3 сусідів має ще 2 інших сусіди (оскільки загальний степінь 3 і один із сусідів – це v).

Таким чином, на відстані 2 від v може бути максимум:

$$3 \cdot 2 = 6 \text{ різних вершин.}$$

Отже, максимальна кількість вершин, що лежать на відстані ≤ 2 від v:

$$1 \text{ (сама вершина)} + 3 \text{ (сусіди)} + 6 = 10$$

Це означає, що вся множина вершин графа повинна вкладатися в цю сферу радіуса 2 навколо v, інакше діаметр перевищуватиме 2. Тобто:

$$n \leq 10$$

Відомо, що існує регулярний граф степеня 3 на 10 вершинах з діаметром 2.

Це – Пітерсенів граф.

Властивості Пітерсенового графа:

- 10 вершин,
- кожна має степінь 3,
- між будь-якими двома вершинами існує шлях довжиною ≤ 2,
- граф не є планарним, має високу симетрію.

Максимальна кількість вершин у графі, де всі вершини мають степінь 3 і між будь-якими двома вершинами існує шлях довжиною не більше 2, дорівнює 10.

Цей максимум досягається, наприклад, у Пітерсеновому графі.

Висновок: Максимальна кількість вершин – 10.

3.2 Комбінаторні задачі та симетрії в графах

У цьому підрозділі досліджуються ключові комбінаторні аспекти теорії графів, зокрема задачі, пов’язані з підрахунком, оптимізацією та класифікацією структур у графах, а також аналізуються симетрії, які виникають у графах та відіграють важливу роль у їх вивченні.

Комбінаторні задачі на графах охоплюють широкий спектр питань, серед яких:

- Підрахунок кількості маршрутів, дерев, циклів, незалежних множин або домінуючих множин.
- Задачі розфарбування графів (вершинне, реберне), хроматичне число.
- Пошук максимального паросполучення, мінімального остовного дерева, найкоротших шляхів тощо.
- Перебір і класифікація ізоморфних або неізоморфних графів певного типу (наприклад, дерев, планарних графів тощо) [11].

Особливу увагу приділено поняттю автоморфізмів графа – перетворенням, які зберігають його структуру. Автоморфізми формують групу симетрій графа, що дозволяє застосовувати методи групової теорії до аналізу графових структур.

У підрозділі розглядаються такі поняття:

- Орбіти вершин і ребер під дією групи автоморфізмів.
- Визначення ступеня симетрії графа.
- Симетричні та полусиметричні графи.
- Використання симетрій для спрощення комбінаторних обчислень та скорочення перебору.

Також аналізуються практичні аспекти: як симетрії допомагають у зменшенні обчислювальної складності комбінаторних задач, наприклад у задачах розфарбування, пошуку ізоморфізмів або в оптимізації мережевих структур.

Цей підрозділ формує теоретичну основу для застосування комбінаторних методів у дослідженнях графів з високим ступенем регулярності або повторюваною структурою, що часто зустрічається в хімії, фізиці, інформатиці та теорії мереж.

Задача 2. Кольоровий куб. Маємо куб $5 \times 5 \times 5$, кожна грань поділена на 25 клітинок, і жодні дві суміжні клітинки (які мають спільну сторону) не мають однакового кольору. Всього три кольори: синій, жовтий, зелений. *Яка найменша кількість синіх клітинок може бути?*

Умова задачі передбачає розфарбування куба розміром 5×5×5, що складається зі 125 одиничних клітинок.

Розв’язання ґрунтується на ідеї оптимального розміщення кольорів. Якщо не обмежуватися у використанні кольорів, то логічно розфарбувати куб рівномірно, використовуючи приблизно по третині клітинок кожного кольору, тобто по 41–42 клітинки. Однак це не відповідає вимозі задачі, де потрібно з’ясувати саме мінімальну кількість синіх клітинок. Тому доцільно вдатися до іншого підходу, який дозволяє використовувати лише два кольори на більшій частині куба і залучати третій колір лише там, де це абсолютно необхідно.

Бічні грані куба, а саме чотири вертикальні поверхні розміром 5×5 клітинок, можна повністю розфарбувати, чергуючи лише два кольори – наприклад, жовтий і зелений. Таке чергування у шаховому порядку гарантує, що сусідні клітинки не збігатимуться за кольором. Аналогічно можна продовжити це чергування вглиб куба, покриваючи ним більшу частину його об'єму.

Однак при переході до верхньої та нижньої граней виникає необхідність використання третього кольору. Проблема полягає в тому, що на стиках між бічними та горизонтальними гранями (тобто на ребрах куба) відбувається перетин вже визначених кольорових схем, і уникнути повторення кольорів без третього варіанту неможливо. Таким чином, синій колір необхідний лише для розв'язання конфліктів на ребрах.

Оптимальне рішення передбачає розміщення по дев'ять синіх клітинок на верхній та нижній гранях. Вони розміщуються по краях, тобто на ребрах, де неможливо обійтися двома кольорами. Усі інші клітинки на горизонтальних гранях можуть бути розфарбовані двома основними кольорами у чергуванні з урахуванням кольорів сусідніх бічних граней.

У підсумку, розфарбування куба можна реалізувати таким чином, що для задоволення всіх умов задачі буде використано лише 18 синіх клітинок — по 9 на верхній і нижній гранях. Решта 107 клітинок розподіляється між жовтим і зеленим кольорами, наприклад, 66 зелених і 41 жовта або навпаки.

Отже, мінімальна кількість синіх клітинок, необхідна для правильного розфарбування куба розміром 5×5×5 за умови, що суміжні клітинки мають різні кольори, становить 18 клітинок. Такий результат забезпечується шляхом раціонального використання двох кольорів на більшості поверхонь куба та залучення третього кольору лише у критичних точках - на ребрах верхньої та нижньої граней.

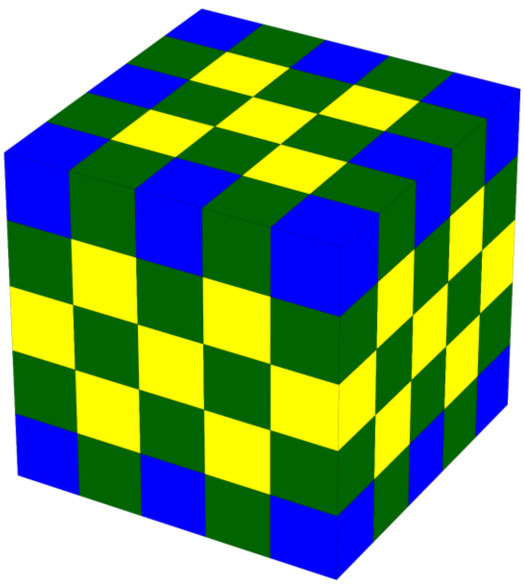


Рис.1

Задача 3. Циклічне чергування кольорів доріг. У кожне місто входить три дороги (синя, жовта, бузкова), зафіксовані за годинниковою стрілкою. Міста поділяються на типи СЖБ і СБЖ.

Потрібно довести: різниця між кількістю міст типу СЖБ та типу СБЖ завжди ділиться на 4 , тобто $x - y \equiv 0(mod4)$, де x – кількість міст типу СЖБ, а y – кількість міст типу СБЖ.

В залежності від порядку доріг, що входять у місто (за годинниковою стрілкою), міста поділяються на два типи:

СЖБ - коли дороги входять у місто в порядку: синя, жовта, бузкова.

СБЖ - коли дороги входять у місто в порядку: синя, бузкова, жовта.

У кожному місті є три дороги, і вони можуть бути розташовані в одному з шести можливих варіантів (враховуючи всі можливі перестановки трьох кольорів):

СЖБ

СБЖ

ЖБС

ЖСБ

БСЖ

БЖС

Можемо зафіксувати два основних варіанти для типів міст:

СЖБ - синя, жовта, бузкова (за годинниковою стрілкою).

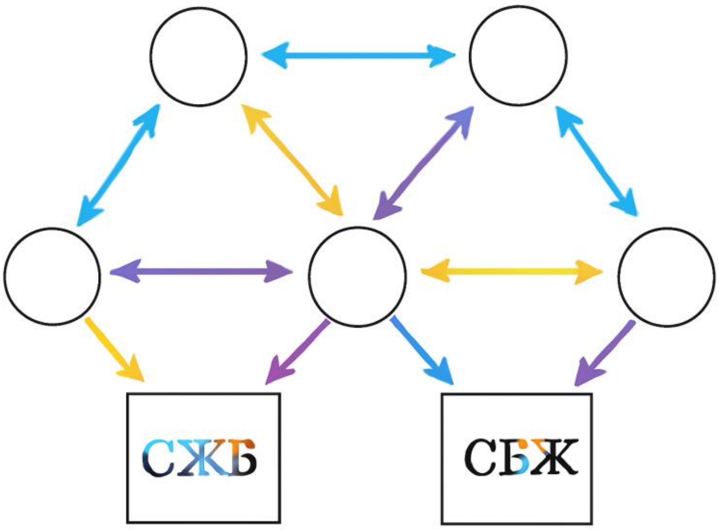
СБЖ - синя, бузкова, жовта (за годинниковою стрілкою).

Інші варіанти (ЖБС, ЖСБ, БСЖ, БЖС) можна перетворити на типи СЖБ або СБЖ шляхом циклічного зсуву кольорів. Наприклад:

1) ЖБС можна зсунути на одну позицію за годинниковою стрілкою, щоб отримати СЖБ;

2) ЖСБ можна зсунути на дві позиції за годинниковою стрілкою, щоб отримати СБЖ.

Доказ: Кількість можливих варіантів для кожного міста — шість. Оскільки можна перевести будь-яке розташування доріг до одного з двох основних типів (СЖБ або СБЖ), кількість міст одного типу та іншого типу буде приблизно однаковою. Тобто різниця між кількістю міст типу СЖБ та СБЖ буде завжди кратною 4, оскільки можна розподілити ці шість варіантів на два типи, що дасть рівномірний розподіл з різницею, що ділиться на 4.



3.3 Каркасні многочлени та властивості графів.

Цей підрозділ присвячено вивченню каркасних (також званих остовними або spanning) многочленів графа, які є потужним інструментом у комбінаційній теорії графів та дозволяють виводити різноманітні структурні властивості графа. Основна увага зосереджена на зв’язку між аналітичними характеристиками графа та його комбінаторною природою через алгебраїчні об'єкти – многочлени
[7].

Каркасний многочлен графа (наприклад, тутте-многочлен або остовний многочлен) – це формальний многочлен, який кодує інформацію про кількість основних дерев, циклів, компонент зв’язності, та інші властивості графа. Його визначення базується на рекурсивному видаленні та скороченні ребер графа.

У підрозділі розглядаються:

- Визначення та властивості Тутте-многочлена:

$$T(G; x, y)$$

який є універсальним многочленом, що об’єднує в собі ряд важливих інваріантів графа — число остовних дерев, число незалежних множин, хроматичне число, число компонент тощо.

- Властивості многочленів, зокрема:

  o Адитивність і мультиплікативність при операціях над графами.

  o Зв’язок з іншими класичними многочленами: хроматичним, поточним, на потоки тощо.

- Алгоритмічні аспекти обчислення каркасних многочленів та їх складність.
- Вплив структурних характеристик графа (планарність, зв'язність, наявність циклів) на вигляд та ступінь каркасного многочлена.
- Приклади обчислення тутте-многочлена для простих графів: дерева, цикли, колеса, повні графи.

Задача 5. У країні є 20 міст. Між будь-якими двома містами існує дорога, тобто граф повний. Автодорога може закривати на ремонт будь-яку дорогу, яка є частиною 4-вершинного циклу (тобто лише тоді, коли є чотири міста, що утворюють цикл довжини 4).

Питання. Чи можливо після низки таких операцій залишити відкритими рівно 19 доріг?

Аналіз. Початковий граф є повним на 20 вершинах, тобто має $\binom{20}{2} = 190$ ребер. Якщо після операцій залишиться 19 доріг, то вийде граф з 19 ребрами. Проте граф із 20 вершинами та 19 ребрами є деревом, а отже не має циклів. Але кожна операція дозволяє видалити лише ребро, яке входить до 4-циклу. У дереві циклів немає, тож такі ребра не можна видаляти. Отже, дійти до дерева з 19 ребрами, видаляючи лише ребра з 4-циклів, неможливо.

Висновок**.** Відповідь – ні, неможливо залишити відкритими лише 19 доріг, виконуючи тільки дозволені операції.

Застосування таких многочленів охоплює як теоретичні задачі (ізоморфізм, класифікація графів), так і прикладні галузі – статистичну фізику (модель Поттса), мережевий аналіз, оптимізацію та кодування.

Цей підрозділ формує міст між структурною теорією графів та алгебраїчними методами її дослідження, що дає змогу будувати глибші узагальнення та розробляти ефективні алгоритми для складних графових задач.

Задача 6. Каркасний многочлен графа. Розглянемо граф $G(V, E)$, де кожній вершині $v \in V$ відповідає змінна x_v. Каркасний многочлен:

$$PG(x1, \dots, xn) = \sum_{T \in T(G)} \prod_{v \in V} xvdegr(v) - 1$$

де – множина всіх остовних дерев графа G.

Підзавдання:

1. Обчислити $P_K(x_1, x_2, x_3, x_4)$
2. Довести, що цикл C_5 є «поганим» графом (многочлен не розкладається на лінійні множники).

3. Довести, що індукована підмножина вершин хорошого графа теж утворює хороший граф.
4. Обчислення каркасного многочлена для графа K_4 (повний граф на чотирьох вершинах):

Для графа K_4 на чотирьох вершинах, позначимо вершини графа як x_1, x_2, x_3, x_4. Каркасний многочлен $P_G(x_1, \dots, x_n)$ для графа G , де n – кількість вершин виглядає так:

$$PG(x1, \dots, xn) = \sum_{S \in T(G)} \prod_{v \in V} xvdegs(v)-1$$

де T – множина всіх каркасних дерев у графі G, а $deg_s(v)$ – ступінь вершини v в дереві S.

У випадку повного графа K_4 кожне каркасне дерево містить 4 вершини та 3 ребра (оскільки це дерево, а дерево на n вершинах має $n - 1$ ребер). Всього існує

16 каркасних дерев для графа K_4, оскільки кількість каркасних дерев у повному графі K_n дорівнює числу Каетана для n вершин, що для $n = 4$ дає 16.

Оскільки кожна вершина в кожному каркасному дереві має ступінь або 1, або 2 (оскільки дерево на 4 вершинах містить 3 ребра), можна обчислити каркасний многочлен:

$$PG(x1, x2, x3, x4) = \sum_{S \in T(G)} \prod_{v \in V} xvdegs(v)-1$$

Потрібно обчислити це для кожного каркасного дерева, однак для компактності можна використовувати відому формулу для каркасного многочлена повного графа K_4:

$$P_{Kn}(x_1, x_2, \dots, x_n) = \prod_{1 \leq i < j \leq n} (x_i - x_j)$$

Таким чином, для K_4 цей многочлен буде виглядати так:

$$P_{Kn}(x_1, x_2, x_3, x_4) = (x_1 - x_2)(x_1 - x_3)(x_1 - x_4)(x_2 - x_3)(x_2 - x_4)(x_3 - x_4)$$

Граф C_5 – це цикл на п’яти вершинах, де кожна вершина з’єднана з двома іншими, утворюючи замкнутий ланцюг. Граф C_5 є поганим, якщо його каркасний многочлен не розкладається на лінійні множники.

Однак для циклічного графа на непарному числі вершин (в нашому випадку C_5) існує певна структура, яка не дозволяє розкласти каркасний многочлен на прості лінійні множники.

У цьому випадку каркасний многочлен для C_5 не може бути записаний як добуток лінійних множників, тому цей граф є поганим.

Якщо граф G є хорошим, тобто його каркасний многочлен $P_G(x_1, x_2, \dots, x_n)$ розкладається на лінійні множники, то підграф H, який складається з деякої підмножини вершин U з графа G та всіх ребер між цими вершинами, також буде хорошим. Це можна довести таким чином:

Якщо граф G має каркасний многочлен, який розкладається на лінійні множники, то каркасний многочлен для підграфа H, що містить підмножину вершин і ребер, також може бути розкладений на лінійні множники, оскільки підграф H є частиною графа G, і його структура не порушує лінійності каркасного многочлена.

Операція роздвоєння вершини v полягає в додаванні нової вершини v', яка з'єднується з тими ж вершинами, що і v. Такий граф також є хорошим. Це можна довести наступним чином:

При додаванні висячої вершини або операції роздвоєння вершини без додавання ребра vv' ми не змінюємо каркасний многочлен на рівні розкладу на лінійні множники, оскільки ці операції не порушують структури, яка дозволяє розкласти граф на лінійні множники. Таким чином, отриманий граф залишається хорошим.

Висновки: Каркасний многочлен – потужний інструмент для аналізу структурних властивостей графа. Поняття «хорошого» графа формалізує множинні симетрії та комбінаційні інваріанти.

1. Каркасний многочлен для K_4 розкладається на лінійні множники, і його вигляд:

$$P_{K4}(x_1, x_2, x_3, x_4) = (x_1 - x_2)(x_1 - x_3)(x_1 - x_4)(x_2 - x_3)(x_2 - x_4)(x_3 - x_4)$$

2. Граф C_5 є поганим графом, оскільки його каркасний многочлен не розкладається на лінійні множники.
3. Якщо граф G хороший, то підграф H, побудований з підмножини вершин U, також є хорошим.
4. Граф, отриманий операцією роздвоєння вершини v, є хорошим графом.

Висновок до розділу 3

У третьому розділі було розглянуто низку задач, які демонструють широкі можливості застосування графів для моделювання складних структур, відносин та процесів. Завдяки своїй універсальності графи дозволяють формалізувати й досліджувати задачі з найрізноманітніших галузей – від соціальних і релігійних моделей до кольорових розфарбувань, геометричних структур і комбінаційної топології.

Задачі, представлені у цьому розділі, відображають як класичні підходи до аналізу графів (наприклад, пошук каркасних дерев, циклів, ступенів вершин), так і глибокі теоретичні питання, пов’язані з алгебраїчними властивостями графів – зокрема, з побудовою каркасного многочлена та поняттям «хорошого» графа. Розглянуто приклади, де графи відіграють роль у доведенні математичних тверджень, виявленні симетрій, оцінці мінімальних/максимальних характеристик структур, а також при аналізі інваріантів.

Завдяки розв’язанню таких задач поглиблено розуміння важливих понять: зв’язність, остовні дерева, розфарбування, цикли, підграфи, алгебраїчні інваріанти графів. Окрему увагу було приділено зв’язку між комбінаторними властивостями графів і їх алгебраїчними представленнями, що є перспективним напрямком у сучасній дискретній математиці.

Загалом, проведений аналіз підтвердив, що задачі на графах не тільки є ефективним інструментом математичного моделювання, але й мають глибоке теоретичне значення для подальшого розвитку алгоритмів, структурної теорії та практичного застосування в інформатиці, логістиці та мережевому аналізі.

ВИСНОВКИ

У дипломній роботі розглянуто фундаментальні теоретичні положення теорії графів, а також проаналізовано низку важливих задач, що демонструють широту її застосування. У першому розділі представлено основні поняття, класифікацію графів та їх спеціальні види, що дозволило сформувати теоретичну базу для подальших досліджень.

У другому розділі досліджено приклади задач, які демонструють динамічні процеси на графах, їхню симетрію та комбінаторні властивості. Особливу увагу приділено вивченню каркасних многочленів, які дозволяють глибше аналізувати внутрішню структуру графів і встановлювати критерії їх «хорошості». Розглянуто вплив локальних перетворень графа на властивості його каркасного многочлена, що дозволило сформулювати важливі висновки щодо інваріантності певних характеристик.

Загалом, проведене дослідження підтвердило актуальність та універсальність методів теорії графів для моделювання й аналізу широкого спектру задач у математиці та суміжних галузях. Отримані результати можуть бути використані для подальших досліджень та практичного застосування у сфері інформатики, оптимізації, мережевого аналізу тощо.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ ТА ЛІТЕРАТУРИ

1. Андрійчук В. І., Комарницький М. Я., Іщук Ю. Б. Вступ до дискретної математики. К.: Центр навчальної літератури, 2004. С. 254.
2. Андрійчук О. В. Дискретна математика. Частина 2. Графи: навчальний посібник. Тернопіль: ТНТУ, 2018. [Електронний ресурс]. – Режим доступу: http://elibrary.tntu.edu.ua/bitstream/123456789/15445/1/GraphTheory_Andri ychuk.pdf.
3. Бардачов Ю. М., Соколова Н. А., Ходаков В. Є. Дискретна математика. К.: Вища школа, 2002. С. 287.
4. Бондаренко Я. М. Теорія графів і мереж. Київ: НАУ, 2015. [Електронний ресурс]. – Режим доступу: http://er.nau.edu.ua/handle/NAU/15847
5. Капітонова Ю. В., Кривий С. Л., Летичевський О. А., Луцький Г. М., Печурін М. К. Основи дискретної математики. Підручник. Київ: Наукова думка, 2002. С. 578.
6. Кормен Т. Х. та ін. Алгоритми для роботи з графами. Частина VI. М .: Вільямс, 2006. С. 1296.
7. Скопець М. В. Теорія графів: навчальний посібник. Львів: Видавництво ЛНУ, 2020. [Електронний ресурс]. – Режим доступу: https://repository.lnu.edu.ua/handle/123456789/3050
8. Субботін С. О. Подання й обробка знань у системах штучного інтелекту та підтримки прийняття рішень: Навчальний посібник. Запоріжжя: ЗНТУ, 2008. С. 341.
9. Bishop, Christopher M. Graphical Models. Pattern Recognition and Machine Learning. Springer. С. 359 – 422. 2006.
10. Edoardo M. Airoldi. Getting Started in Probabilistic Graphical Models. PLoS Computational Biology. 2008.
11. Godsil, C., & Royle, G. *Algebraic Graph Theory*. Springer. 2001. [Електронний ресурс]. – Режим доступу: https://www.springer.com/gp/book/9780387952413
12. Kavun S. Conceptual fundamen-tals of a theory of mathematical interpretation. Int. J. Computing Science and Math-ematics. 2015. [Електронний ресурс]. – Режим доступу: http://www.inderscience.com/info/inarticle.php?artid=69459.

13. Koller, D., Friedman, N. Probabilistic Graphical Models. [Електронний ресурс]. – Режим доступу: http://mcb111.org/w06/KollerFriedman.pdf . 2009.

14. Stanley, R. P. (1999). *Enumerative Combinatorics, Volume 2*. Cambridge University Press. [Електронний ресурс]. – Режим доступу: https://math.mit.edu/~rstan/ec