**Державний університет телекомунікацій**

**Комп’ютерні дискретні структури**

**Частина 3**

**Елементи теорії графів**

**Завдання комплексної семестрової роботи № 3**

**з методичними вказівками і зразками розв’язувань**

**для студентів денної форми навчання галузі 12 – Інформаційні технології**

**Київ – 2017**

**Комп’ютерні дискретні структури. Частина 3.** **Елементи теорії графів. Завдання комплексної семестрової роботи** **№ 3** **з методичними вказівками і зразками розв’язувань** для студентів денної форми навчання галузі 12 – Інформаційні технології / Укладач: Жданова Ю. Д., Шевченко С.М. – К.:ДУТ, 2017. – 53 с.

Відповідає другому розділу програми курсу комп’ютерні дискретні структури для студентів денної форми навчання галузі 12 – Інформаційні технології. Містить варіанти індивідуальних контрольних завдань для перевірки знань студентів денної форми навчання з другому розділу програми дисципліни Комп’ютерні дискретні структури.

Наведені типові приклади контрольних завдань та методика їх розв’язування.

Довідкові матеріали забезпечують вивчення всіх тем другого розділу програми.

**Зміст**

1. Зміст другого розділу навчальної програми

з дисципліни Комп’ютерні дискретні структури 4

1. Перелік базових понять, задач, методів, знань і вмінь,

якими повинен оволодіти студент після вивчення

розділу **Елементи теорії графів**  4

1. Інформаційно-методичне забезпечення 6
2. Методичні вказівки до опрацьовування тем 6 6
3. Рекомендації до виконання комплексної семестрової роботи № 3 7
4. Основні вимоги до оформлення комплексної семестрової роботи № 3 8
5. Завдання комплексної семестрової роботи № 3 10
6. Методичні вказівки і зразки виконання і оформлення завдань

комплексної семестрової роботи № 3 55

1. Довідкові матеріали 67

##### 1. ЗМІСТ ТРЕТЬОГО РОЗДІЛУ НАВЧАЛЬНОЇ ПРОГРАМИ

з дисципліни Комп’ютерні дискретні структури

**МОДУЛЬ 3**

**ЕЛЕМЕНТИ ТЕОРІЇ ГРАФІВ**

Основні характеристики графів. Зображення графів. Матричні способи задання графа. Ізоморфізм графів. Маршрути в графі. Обходи в графах.

Досяжність і зв’язність графа. Компоненти зв’язності. Дерева.

Задача пошуку мінімального остовного дерева зваженого графа. Задача пошуку мінімального шляху в зваженому орграфі**.**

**2. Перелік базових понять, задач, методів, знань і**

**вмінь, якими повинен оволодіти студент після вивчення розділу**

**ЕЛЕМЕНТИ ТЕОРІЇ ГРАФІВ**

**Базові поняття**

Граф. Неорієнтований граф. Орієнтований граф. Відношення інцидентності. Геометричне зображення графа. Степінь вершини гарфа. Підграф. Остовний підграф. Матриця суміжності. Матриця інцидентності. Маршрут. Цикл. Відстань між вершинами графа. Зв’язність. Матриця досяжності. Зважений граф. Матриця довжин ребер. Мінімальний шлях між вузлами орієнтованого графа. Дерево. Мінімальне остовне дерево.

**Основні задачі, які виникають на основі базових понять**

1. Задання графа матричними способами.
2. Побудова геометричного зображення графа.
3. Визначення маршрутів в графі.
4. Визначення відстані між вершинами графа.
5. Знаходження ейлеревого циклу в графі.
6. Визначення числа маршрутів певної довжини.
7. Визначення компонентів зв’язності графа.
8. Пошук мінімального шляху між двома різними вузлами зваженого орграфа.
9. Побудова мінімального остовного дерева зваженого графа.

**Базові методи розв’язування основних задач**

1. Побудова геометричного зображення графа, заданого матричним способом.
2. Побудова матриці суміжності та матриці інцидентності графа, заданого графічно.
3. Застосування матриці суміжності для визначення числа маршрутів певної довжини.
4. Застосування матриці суміжності для визначення компонентів зв’язності графа.
5. Алгоритм Дейкстри пошуку мінімального шляху між двома різними вузлами зваженого орграфа.
6. Алгоритм Краскала побудови мінімального остовного дерева зваженого графа.

**Знання на рівні понять, означень, описів, формулювань**

1. Граф як сукупність двох скінченних множин.
2. Відношення інцидентності між множинами вершин і ребер графа.
3. Неорієнтований граф.
4. Орієнтований граф.
5. Геометричне зображення графа.
6. Степінь вершини графа.
7. Теорема Ейлера.
8. Підграф.
9. Остовний підграф.
10. Операції над графами.
11. Матриця суміжності графа.
12. Матриця інцидентності графа.
13. Маршрут в графі.
14. Цикл.
15. Відстань між вершинами графа.
16. Ейлеровий цикл.
17. Звязний граф.
18. Теорема про число маршрутів в графі.
19. Матриця досяжності графа.
20. Компонента зв’язності графа.
21. Конденсація графа.
22. Зважений граф.
23. Матриця довжин дуг графа.
24. Мінімальний шлях в зваженому орграфі.
25. Опис алгоритму Дейкстри.
26. Дерево.
27. Мінімальне остовне дерево.
28. Опис алгоритму Краскала.

**Знання на рівні** **доведень**

1. Теорема Ейлера.
2. Теорема про число маршрутів в графі.

**Вміння, необхідні для розв’язування задач**

1. Побудова геометричного зображення графа, заданого сукупністю двох скінченних множин.
2. Побудова геометричного зображення графа, заданого матричним способом.
3. Побудова матриці суміжності графа, заданого геометричним зображенням.
4. Визначення характеристик графа за його матрицею суміжності.
5. Побудова матриці інцидентності графа, заданого геометричним зображенням.
6. Побудова матриці досяжності графа.
7. Побудова матриці довжин дуг графа.
8. Застосування алгоритму Дейкстри.
9. Застосування алгоритму Краскала.

**3**. **ІНФОРМАЦІЙНО-МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ**

**3.1.** **Література**

1. Бардачов Ю.М. та ін. Дискретна математика. – К.: Вища школа, 2002. – 287 с.
2. Биркгоф Г., Барти Т. Современная прикладная алгебра. – М.: Лань, 2005. – 400 с.
3. Гаврилов Г.П., Сапоженко А.А. Задачи и упражнения по дискретной математике. – М.: ФИЗМАТЛИТ, 2004. – 416 с.
4. Донской В.И. Дискретная математика. – Симферополь:СОНАТ, 2000. – 360 с.
5. Капітонова Ю.В., Кривий С.Л., Летичевський О.А., та ін. Основи дискретної математики. Підручник. – Київ: Наукова думка, 2002. – 580 с.
6. Кузнецов О.П., Адельсон-Вельский Г.М. Дискретная математика для инженера. – М.: Энергоатомиздат, 1988. – 480 с.
7. Новиков Ф.А. Дискретная математика для программистов. – СПб: Питер. – 2000. – 304 с.
8. Поздняков С.Н., Рыбин С.Н. Дискретная математика. – М.: Издательский центр «Академия», 2008. – 448 с.
9. Сигорский В.П. Математический аппарат инженера. – Киев: Техника, 1976. – 786 с.
10. Таран Т.А., Мыценко Н.А., Темникова Е.Л. Сборник задач по дискретной математике. – К.; Просвіта, 2001. – 61 с.
11. Тевяшев А.Д., Гусарова И.Г. Основы дискретной математики в примерах и задачах. – Харьков: ХНУРЭ, 2003. – 272 с.
12. Яблонский С.В. Введение в дискретную математику. – М.: Наука,1986. – 384с.

**3.2. Інформаційні ресурси**

1. Тексти лекцій з дискретної математики (електронний варіант).
2. Електронна бібліотека ДУІКТ.
3. Інтернет-ресурси:
4. <http://elibrary.ru/> – Научная электронная библиотека.
5. http://[www.scientific-library.net](javascript:window.external.AddFavorite('http://www.scientific-library.net',%20'%20www.scientific-library.net%20-%20Огромная%20библиотека%20научно-технической%20литературы')) – Электронная библиотека научно-технической литературы
6. http://[www.allbest/ru/](http://www.allbest/ru/) – Бесплатные электронные библиотеки: математика.
7. <http://www.exponenta.ru/> – Образовательный математический сайт: задачи с решениями, справочник по математике, программы курсов и т.п.
8. <http://www.allmath.ru/> – Электронные материалы по математике.
9. <http://www.mccme.ru/free-books/> – Сайт свободно распространяемых изданий, а также записи лекций, сборники задач.
10. <http://www.matburo.ru/st_subject.php?p=dm> – Учебники, On-line ресурсы по дискретной математике

**4. Методичні вказівки до опрацьовування теми**

В темах **розділу Елементи теорії графів** вивчаються скінченні графи. Скінченний граф ** – це пара множин **, де (ребра) – довільна підмножина множини пар елементів множини ** (вершини). Само по собі дуже просте поняття графу є найважливішим математичним поняттям в дискретній математиці, теоретичній інформатиці, програмуванні.

Як розділ дискретної математики, теорія графів має численні застосування. Вона використовується в задачах керування виробництвом, при проектуванні електричних і комп’ютерних мереж, плануванні транспортних перевезень, побудові молекулярних схем. Застосовується теорія графів також в економіці, психології, соціології, біології. На основі теорії графів будуються моделі різних задач: маршрутизації, розподілу ресурсів, дискретної оптимізації і керування виробництвом, мінімізації алгоритмів, автоматів, при розробці сучасних електронних модулів, при рішенні проблем автоматизації проектування (САПР) та ін.

Найбільш широке застосування методи теорії графів знаходять у програмуванні, тому що теорія графів надає дуже зручну мову для опису програмних (і багатьох інших) моделей. Струнка система спеціальних термінів і позначень теорії графів дозволяють просто і доступно описувати складні і тонкі речі. Особливо важлива наявність наочної графічної інтерпретації поняття графа. Сама назва «граф» має на увазі наявність графічної інтерпретації. Наявність графічної інтерпретації робить цю тему легкою для засвоєння.

Іноді абстрактне поняття графа як сукупність двох множин, виявляється недостатнім і доводиться розглядати більш загальні об'єкти, в яких дві вершини можуть з'єднуватися більш ніж одним ребром. Так виникає «мультиграф». Якщо в графі існують петлі, то маємо «псевдо граф». Також вивчаються орграфи (орієнтовані графи), в яких множина пар елементів замінюється декартовим квадратом *.*

При вивченні цієї теми треба звернути увагу на способи задання графів: геометричний, алгебраїчний. Треба навчитися мові теорії графів, яка є мовою дискретної математики.

При розв’язуванні багатьох практичних задач виникає необхідність пошуку мінімального шляху між двома довільними вузлами(шляху з найменшою сумою довжин дуг) зваженого графа. Для знаходження мінімального шляху між двома довільними вузламиіснують певні алгоритми, наприклад, алгоритм Дейкстри.

Типове застосування остовних дерев мінімальної вартості можна знайти при розробці комунікаційних мереж. Тут вершини графа являють собою міста, ребра – можливі комунікаційні лінії між містами, а вартість ребер відповідає вартості комунікаційних ліній. У цьому випадку остовне дерево мінімальної вартості представляє комунікаційну мережу, що поєднує всі міста комунікаційними лініями мінімальної вартості.

Для побудови остовного дерева мінімальної вартості для зваженого графа існують певні алгоритми, наприклад, алгоритм Краскала. Цей алгоритм відповідає «жадібній» стратегії: на кожному кроці вибирається локально найкращий варіант.

**5. Рекомендації до виконання комплексної семестрової роботи № 3**

Завдання комплексної семестрової роботи розподіляються по варіантах, номери яких відповідають номерам прізвищ студентів в журналі академгрупи.

Комплексна семестрова робота здається на перевірку в термін, вказаний викладачем. Виконати самостійно завдання всієї комплексної семестрової роботи і оформити результати після завершення вивчення усіх тем достатньо складно, тому кожне завдання комплексної семестрової роботи рекомендується виконувати після опрацювання і засвоєння відповідної теми на практичному занятті і самостійно.

Оцінка цієї роботи значно залежить не тільки від правильного розв’язання прикладів та задач, але і від своєчасного його подання викладачу для перевірки та оцінювання, від повноти пояснень розв’язків та акуратності оформлення роботи.

Наслідком несамостійного виконання цієї роботи будуть значні складності у подальшому вивченні багатьох розділів математики та багатьох розділів інших навчальних дисциплін.

**6. Основні вимоги до оформлення комплексної семестрової роботи № 3**

При оформленні комплексної семестрової роботи потрібно дотримуватись наступних правил:

– завдання слід виконувати в окремому зошиті на 12 аркушів, або на подвійних листах в клітинку, або на білому папері А4. Титульний лист роботи оформлюється так:

Державний університет телекомунікацій

*Комплексна семестрова робота № 3*

*з дисципліни комп’ютерні дискретні структури*

*Варіант №\_\_\_\_\_*

# Виконав(ла): студент(ка) групи

Прізвище І.Б.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата здачі\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата МК 3\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оцінка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Перевірив\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

20\_\_\_

– завдання потрібно виконувати в такому порядку, як вони задані. Перед розв'язуванням треба **виписувати повністю умову завдання**;

– розв’язування задач та пояснення до них слід викладати детально, охайно, без скорочення слів, з необхідними поясненнями та малюнками;

– розв’язування задач мають доводитися до відповідей, які вимагаються умовами задач;

– в кінці роботи треба вказати використану літературу, поставити дату виконання і особистий підпис.

Комплексна семестрова робота, що виконана неохайно, без проміжних обчислень та необхідних пояснень, повертається студенту для переробки. Повертаються також роботи, в яких виконано не свій варіант.

Після отримання перевіреної роботи студент повинен виправити вказані викладачем помилки та недоліки у тому самому зошиті до захисту.

**7. Завдання комплексної семестрової роботи № 3**

**Варіант 1**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 2**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 3**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер



**Варіант 4**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 5**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 6**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 7**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 8**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла до вузла в зваженому орграфі , заданому матрицею довжин дуг:



.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа , заданого матрицею довжин ребер



.

**Варіант 9**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла до вузла в зваженому орграфі , заданому матрицею довжин дуг:



.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа , заданого матрицею довжин ребер



.

**Варіант 10**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла до вузла в зваженому орграфі , заданому матрицею довжин дуг:



.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа , заданого матрицею довжин ребер



.

**Варіант 11**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 12**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 13**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 14**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 15**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 16**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 17**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 18**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 19**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 20**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 21**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 22**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 23**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 24**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 25**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 26**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 27**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 28**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 29**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**Варіант 30**

1. Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.
10. Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

1. Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

**8. Методичні вказівки і зразки виконання і оформлення завдань**

**комплексної семестрової роботи № 2**

Перед виконанням завдань комплексної семестрової роботи № 3 треба проробити відповідну теорію за підручником (наприклад, [1]), за текстом лекцій, за довідковими матеріалами (стор. 50–53 ).

Завдання 1 розв’язується з використанням означень основних понять теорії графів.

**Зразок виконання і оформлення *Завдання 1***

Для графа, заданого матрицею суміжності



1. намалювати геометричне зображення;
2. знайти всі маршрути довжини 2 і 3;
3. знайти матрицю досяжності;
4. знайти компоненти зв’язності ( сильної зв’язності);
5. знайти матрицю зв’язності;
6. побудувати граф конденсації;
7. знайти матрицю інцидентності;
8. знайти степені вершин;
9. знайти ейлерів цикл. Якщо такого циклу в графі нема, то вилучити мінімально можливе число дуг, і в новому графі знайти такий цикл.

***Розв’язання.*** Оскльки матриця суміжності  несиметрична, то маємо орієнтований граф .

1. Намалюємо орграф:

















1. Знайдемо в графі  всі шляхи довжини 2. Для цього матрицю суміжності  піднесемо до другого степеня:



Оскільки елемент  в матриці  дорівнює 1, в  існує один шлях довжини 2 з вершини  у вершину . Це шлях  ,, .

Оскільки елемент  в матриці  дорівнює 1, в  існує один шлях довжини 2 з вершини  у вершину . Це шлях , , .

Оскільки елемент  в матриці  дорівнює 2, в  існує два шляхи довжини 2 з вершини  у вершину . Це шляхи , ,  і , , .

Аналогічно знаходяться інші шляхи довжини 2.

Зведемо знайдені шляхи довжини 2 у таблицю:

|  |  |  |
| --- | --- | --- |
| Початок | Кінець | Шлях довжини 2 |
|  |  | ,, |
|  |  | , , . |
|  |  | , , |
|  |  | , , ;  , , . |
|  |  | , , |
|  |  | , , |
|  |  | , , |
|  |  | , , |
|  |  | , , |

Знайдемо в графі  всі шляхи довжини 3. Для цього матрицю суміжності  піднесемо до третього степеня:



Оскільки елемент  в матриці  дорівнює 1, в  існує один шлях довжини 3 з вершини  у вершину . Це шлях , , , .

Оскільки елемент  в матриці  дорівнює 1, в  існує один шлях довжини 3 з вершини  у вершину . Це шлях , , , .

Оскільки елемент  в матриці  дорівнює 2, в  існує два шляхи довжини 3 з вершини  у вершину . Це шляхи , , ,  і , , , .

Аналогічно знаходяться інші шляхи довжини 3.

Зведемо знайдені шляхи довжини 3 у таблицю:

|  |  |  |
| --- | --- | --- |
| Початок | Кінець | Шлях довжини 3 |
|  |  | , , , |
|  |  | , , , |
|  |  | , , , ;  , , , |
|  |  | , , , |
|  |  | , , , |
|  |  | , , , |
|  |  | , , , |
|  |  | , , , |
|  |  | , , , |

1. Побудуємо матрицю досяжності . Для цього обчислимо вираз



і застосуємо до нього булеве відображення:



.

1. Використаємо матрицю досяжності для виявлення компонентів зв’язності графа . Обчислимо матрицю , де  – поелементне множення матриць:

.

Ця матриця має блоково-діагональий вигляд, де кожний блок визначає сильну компоненту зв’язності. Так, вершина  складає одну сильну компоненту зв’язності. Вершини і , яким відповідають одиниці на головній діагоналі матриці , також складають сильні компоненти зв’язності. Вершини , , ,  складають одну сильну компоненту зв’язності, оскільки їм відповідає один блок в матриці . Вершина  складає ще одну сильну компоненту зв’язності.

Піднесемо матрицю досяжності  до другого степеня:



Діагональні елементи  матриці  показують число елементів в тій сильній компоненті зв’язності, в яку входить вершина *і*.

1. Побудуємо граф конденсації  графа . Для цього позначимо сильні компоненти графа  наступним чином:

,

,

,

,

.

Граф конденсації  графа  має вигляд:

І

І

І

І

І

1. Визначимо тип зв’язності графа . Це легко зробити за графом конденсації . Оскільки в графі  є вершини, жодна з яких не досяжна з іншої, наприклад,  і , то граф є слабко зв’язаним.
2. Знайдемо матрицю інцидентності  графа . Для цього позначимо дуги графа  наступним чином:





































Елемент  матриці інцидентності  дорівнює –1, якщо дуга  виходить з вершини , дорівнює 1, якщо дуга  входить у вершину , і дорівнює 0, якщо дуга  не є інцидентною вершині .



1. Знайдемо степені вершин: , , , .

|  |  |  |  |
| --- | --- | --- | --- |
| Вершина |  |  |  |
|  | 1 | 2 | 3 |
|  | 1 | 1 | 2 |
|  | 3 | 0 | 3 |
|  | 1 | 1 | 2 |
|  | 1 | 2 | 3 |
|  | 2 | 1 | 3 |
|  | 1 | 2 | 3 |
|  | 1 | 0 | 1 |

1. Оскільки в асоційованому графі ****** (графі, отриманому з  скасуванняморієнтації) існують вершини з непарним степенем, то ейлерового циклу в графі нема.

Вилучимо з графа ****** ребра , , :































Тоді, наприклад, ейлеровий цикл: , , , , , , , .

Завдання 2,3 розв’язуються за алгоритмом Дейкстри (стор. 52) і алгоритмом Краскала (стор. 52) відповідно.

**Зразок виконання і оформлення *Завдання 2***

Користуючись алгоритмом Дейкстри, знайти мінімальний шлях від вузла  до вузла  в зваженому орграфі , заданому матрицею довжин дуг:

.

***Розв’язання.*** Протокол роботи будемо оформлювати у вигляді таблиці, яку побудуємо в процесі розв’язування.

***Крок 0***. Всі вузли помічаються: стартовий вузол  отримує постійну мітку 0, всі інші вузли отримують тимчасові мітки :

, .

Ці значення занесемо в нульовий рядок таблиці.

***Крок 1.*** Для вузлів  з тимчасовими мітками, в які заходять дуги з вузла  з постійною міткою, оновлюємо мітки:

;

**;

**;

**

**.

Постійну мітку отримує вузол .

Ці значення занесемо в перший рядок таблиці.

***Крок 2.*** Для вузлів  з тимчасовими мітками, в які заходять дуги з вузла  з постійною міткою, оновлюємо мітки:

;

**;

**;

**;

**.

Постійну мітку отримує вузол .

Ці значення занесемо в другий рядок таблиці.

***Крок 3.*** Для вузлів  з тимчасовими мітками, в які заходять дуги з вузла  з постійною міткою, оновлюємо мітки:

;

** ;

**;

**.

Постійну мітку отримує вузол .

Ці значення занесемо в третій рядок таблиці.

***Крок 4.*** Для вузла  з тимчасовою міткою, в яку заходять дуга з вузла  з постійною міткою, оновлюємо мітку:

**

**.

Постійну мітку отримує вузол .

Ці значення занесемо в четвертий рядок таблиці.

***Крок 5.*** З вузла  з постійною міткою не входить жодна дуга.

;

**.

Постійну мітку отримує вузол .

Ці значення занесемо в п’ятий рядок таблиці.

***Крок 6.*** Для вузла  з тимчасовою міткою, в яку заходять дуга з вузла  з постійною міткою, оновлюємо мітку:

**

**.

Постійну мітку отримує вузол .

Ці значення занесемо в шостий рядок таблиці.

**Таблиця кроків алгоритму Дейкстри**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Вузли  Кроки |  |  |  |  |  |  |  |
| 0 | 0 |  |  |  |  |  |  |
| 1 | 0 | 2 |  |  |  |  |  |
| 2 | 0 | 2 | 4 |  |  |  |  |
| 3 | 0 | 2 | 4 |  |  | 5 |  |
| 4 | 0 | 2 | 4 |  |  | 5 | 15 |
| 5 | 0 | 2 | 4 |  | 10 | 5 | 15 |
| 6 | 0 | 2 | 4 | 18 | 10 | 5 | 15 |

Всі вузли отримали постійні мітки. Алгоритм завершено: **, **, **, **, **, **.

Щоб знайти мінімальний шлях від вузла  до вузла , використаємо тимчасові індекси в постійних міток: у вузла  – , у вузла  – , у вузла  – , у вузла  – . Отже, мінімальний шлях  з  до  є , , , , . Його довжина



.

**Зразок виконання і оформлення *Завдання 3***

Користуючись алгоритмом Краскала, знайти мінімальне остовне дерево для зваженого графа **, заданого матрицею довжин ребер

.

***Розв’язання.***

***Крок 1*.** Вибираємо в графі ** ребро мінімальної довжини. Ребер мінімальної довжини 1 три: , , . Беремо . Будуємо граф **, що складається з даного ребра і інцидентних йому вершин. Покладаємо **. Оскільки **, то переходимо до кроку 2.





1

***Крок 2*.** Будуємо граф **, додаючи до графа ** нове ребро мінімальної довжини, вибране серед всіх ребер графа **, кожне з яких інцидентне одній з вершин  графа ** і одночасно інцидентне якій-небудь вершині графа **, що не міститься в **, тобто одній з вершин . Таким чином, треба вибрати ребро мінімальної довжини з ребер , , , , . Ребер мінімальної довжини 1 два: , . Беремо . Разом з цим ребром включаємо в ** й інцидентну йому вершину , що не міститься в **. Покладаємо **. Оскільки **, то переходимо до кроку 3.







1

1

***Крок* 3.** Будуємо граф **, додаючи до графа ** нове ребро мінімальної довжини, вибране серед всіх ребер графа **, кожне з яких інцидентне одній з вершин  графа ** і одночасно інцидентне якій-небудь вершині графа **, що не міститься в **, тобто одній з вершин . Таким чином, треба вибрати ребро мінімальної довжини з ребер , , , . Ребро мінімальної довжини 2 одне: . Разом з цим ребром включаємо в ** й інцидентну йому вершину , що не міститься в **. Покладаємо **. Оскільки **, то переходимо до кроку 4.









2

1

1

***Крок* 4.** Будуємо граф **, додаючи до графа ** нове ребро мінімальної довжини, вибране серед всіх ребер графа **, кожне з яких інцидентне одній з вершин  графа ** і одночасно інцидентне вершині графа **, що не міститься в **, тобто вершині . Таким чином, треба вибрати ребро мінімальної довжини з ребер , , . Ребер мінімальної довжини 3 два: , . Беремо . Разом з цим ребром включаємо в ** й інцидентну йому вершину , що не міститься в **. Покладаємо **. Оскільки **, то граф ** – шукане мінімальне остовне дерево.











2

1

1

3

**9. Довідкові матеріали**

**Основні поняття теорії графів**

**Граф (скінченний граф) ** – сукупність двох множин: скінченої множини  і множини  пар елементів з . Елементи множини  – **вершини** графа, елементи множини  – його **ребра.**

**Відношення інцидентності**– відношення між множинами вершин ** і ребер ** типу «проходить через» або «лежить на»

**Суміжні вершини (ребра)** – дві вершини (два ребра), інцидентні одному ребру (одній вершині).

**Орієнтований граф (орграф)**  – граф, в якого елементи множини  – впорядковані пари. В орграфа елементи множини  називаються **вузлами**, а елементи множини  – **дугами**.

**Псевдограф** – граф з петлями (**петля** – елемент множини , який складається з пари однакових елементів множини ).

**Мультиграф** – граф з кратними ребрами (**кратне ребро** – декілька однакових елементів множини ).

**Геометричне зображення графа  –** геометричнафігура, вершинам якої відповідають вершини графа, ребрам – ребра графа відповідні ребра у фігурі сполучають вершини, відповідні вершинам в графі.

**Степінь вершини**  – число ребер, інцидентних вершині . Позначається .

**Висяча вершина** – вершина, степінь якої дорівнює 1*.*

**Ізольована вершина** – вершина, степінь якої дорівнює 0*.*

**Півстепінь виходу вузла**  **орграфа –** число дуг, що виходять з вузла . Позначається .

**Півстепінь заходу вузла**  **орграфа –** число дуг, що вхлдять у вузол . Позначається .

**Підграф** даного графа – граф, що містить деяку підмножину множини  **вершин** графа і деяку підмножину інцидент них їм ребер.

**Остовний підгра**ф – підграф, що містить всі вершини даного графа.

**Матриця суміжності графа**  – квадратна -матриця , де 

**Матриця інцидентності графа**  – -матриця , де 

**Матриця інцидентності орграфа** ** – *–*матриця *,* де



**Ізоморфні графи** – графи, між множинами вершин і ребер яких існує взаємно однозначна відповідність, яка зберігає відношення інцидентності.

**Інваріант графа –** числова характеристика, однакова для всіх ізоморфних графів. Наприклад, число вершин, число ребер.

**Маршрут в графі** – послідовність вершин і ребер, які чергуються: , в який будь-які два сусідніх елементи інцидентні.

**Замкнений маршрут** – маршрут у разі **.

**Ланцюг в графі** – маршрут, всі ребра якого різні.

**Простий ланцюг** – ланцюг, в якого всй вершини різні.

**Цикл** – замкнений ланцюг.

**Ациклічний граф** – граф, що не містить циклів.

**Простий цикл** – замкнений простий ланцюг.

**Ейлеровий цикл** – цикл, який містить всі ребра графа один раз (вершини можуть повторюватись).

**Ейлеровий** граф – граф, що містить ейлерові цикли.

**Шлях** – ланцюг в орграфі.

**Контур –** цикл в орграфі.

**Довжина маршруту** – число ребер маршруту порядку їхнього проходження.

**Відстань**  між вершинами  і  – мінімальна довжина простого ланцюга з початком  і кінцем .

**Зв’язний граф** – граф, будь-яка пара вершин якого може бути з'єднана маршрутом.

**Матриця досяжності** графа  – матриця , де 

**Компонента зв’язності** графа  – підграф  такий, що для будь-яких двох вершин з множини  існує маршрут з однієї в іншу і не існує маршруту з вершини з множини  у вершину з .

**Матриця зв’язності** графа  – матриця , де 

**Компонента сильної зв’язності** орграфа  – максимальний сильнозв'язний підграф , тобто такий, що для будь-яких двох вершин з множини  існує шлях як з однієї в другу, так і з другої в першу.

**Конденсація**  () графа  (орграфа ) (або фактор-граф) – граф, який отриманий стягуванням в одну вершину кожної компоненти зв’язності (сильної зв’язності) графа  (орграфа ).

**Зважений орграф** – орграф, кожній дузі якого зіставлене деяке число (вага, або довжина дуги).

**Матриця довжин ребер** або **матриця** **вагів** орграфа  – матриця , де 

**Мінімальний шлях** між двома довільними вузламизваженого органа – шлях з найменшою сумою довжин дуг.

**Алгоритм Дейкстри пошуку мінімального шляху**

**між двома довільними вузлами**

***Крок 0***. Всі вузли помічаються: стартовий вузол  отримує постійну мітку 0, всі інші вузли отримують тимчасові мітки .

***Крок і*** для будь-якого . Для всіх вузлів  з тимчасовими мітками, в які заходять дуги з вузла  з постійною міткою, оновлюємо мітки. За нову мітку обираємо мінімум із старої мітки і суми мітки вузла  і відстані між  і . Серед вузлів  з тимчасовими мітками, знаходимо вузол з найменшою міткою. Мітку цього вузла проголошуємо постійною. (При чисельному виконанні алгоритму вручну мітку підкреслюємо).

Якщо всі вузли отримали постійні мітки, закінчити роботу алгоритму. Отримані постійні мітки дають відстані від стартового вузла  до всіх інших вузлів орграфа.

Якщо залишилися вузли з тимчасовими мітками, повторити крок ***і***.

**Зважений граф** – граф, кожному ребру якого зіставлене деяке число (вага, або довжина ребра).

**Матриця довжин ребер** або **матриця** **вагів** графа  – матриця , де 

**Дерево** – зв'язний неорієнтований граф без циклів.

**Корінь дерева** – виділена вершина дерева.

**Остовне дерево** для графа  – остовний підграф, який є деревом.

**Мінімальне остовне дерево зваженого графа –** остовне дерево, в якого сума довжин ребер мінімальна.

**Алгоритм Краскала побудови** **мінімального остовного дерева**

**Крок 0.** Установка початкових значень.

Вводимо матрицю довжин ребер  графа .

**Крок 1.** Вибираємо в графі  ребро мінімальної довжини (якщо таких ребер декілька, беремо будь-яке з них). Будуємо граф , що складається з даного ребра і інцидентних йому вершин. Покладаємо . Оскільки , то переходимо до кроку 2.

**Крок і** для будь-якого . Побудувати граф **, додаючи до графа ** нове ребро мінімальної довжини, вибране серед всіх ребер графа **, кожне з яких інцидентне якій-небудь вершині графа ** і одночасно інцидентне якій-небудь вершині графа , що не міститься в **. Разом з цим ребром включаємо в ** й інцидентну йому вершину, що не міститься в **. Якщо **, де ** – число ребер графа, то граф ** – шукане мінімальне остовне дерево (задача розв’язана), якщо ** – перейти до кроку **.