МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Київський національний університет будівництва і архітектури

Прикладна теорія графів

Методичні вказівки
до виконання практичних та лабораторних робіт
для студентів спеціальності
122 Комп’ютерні науки, 126 Інформаційні системи і технології

Київ 2024

УДК 004.042

Укладачі: О.Л. Соловей, канд. техн. наук

Відповідальна за випуск О.О. Терентьєв, доктор.тех. наук, професор

Затверджено на засіданні кафедри інформаційних технологій проектування та прикладної математики, протокол № 4 від 06 грудня 2023 року.

В авторській редакції.

Прикладна теорія графів: Методичні вказівки до виконання практичних та лабораторних робіт / Уклад. О.Л. Соловей. – Київ: КНУБА, 2024. – 46 с

Містять теоретичні відомості і рекомендації щодо виконання лабораторних робіт з дисципліни та вимоги до оформлення звіту. Спрямовані на організацію самостійної роботи студентів.

Призначені для студентів спеціальності 122 Комп'ютерні науки, 126 Інформаційні системи і технології для практичного використання при виконанні лабораторних робіт.

Оглавление

Вступ

Лабораторні роботи є логічним продовженням лекційного курсу з дисципліни "Прикладна теорія графів" і є перехідною ланкою від теоретичного курсу до набуття практичних навичок з розвязку задач на графах.

Кожна лабораторна робота містить наступні види робіт:

1. аналіз умови задачі і розробка підходу до її розв'язку.
2. покрокову розробку алгоритму розв'язку і його опис.
3. написання програми, що реалізує цей алгоритм.
4. демонстрація правильної роботи програми на обраному наборі тестів.
5. складання і захист звіту.

Звіт з лабораторної роботи містить наступні елементи і оформлюється в Word

- ✓ Умова задачі.
- ✓ Опис алгоритму.
- ✓ Лістінг.
- ✓ Результати роботи програми.
- ✓ Відповіді на контрольні запитання.

Лабораторний практикум №1

Розробка програми для опису об'єкту типу «граф» та основних характеристик неорієнтовних графів.

Мета роботи: здобути навики опису складних об'єктів типу граф мовою програмування.

Завдання: розробити клас «граф», методи якого для заданого графу визначатимуть його характеристики: 1-5.

1. простий ланцюг графу G.
2. Матрицю відстаней.
3. Діаметр D та радіус R.
4. Центральні та периферійні вершини.
5. Міст графу G (за наявності).

Користувацькій інтерфейс має підтримувати можливість задати вихід граф теоретико-множинним способом.

Приклад виконання:

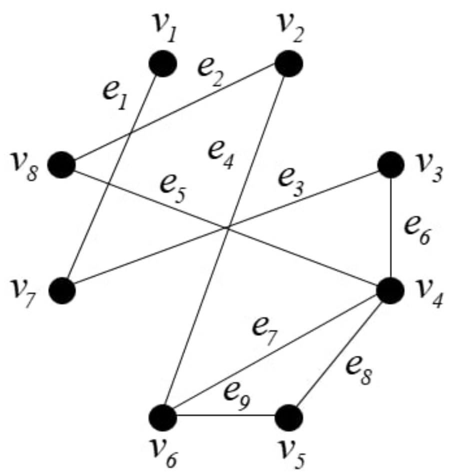


Рисунок 1. Вихідний неорієнтовний граф

1. Визначення простого ланцюга графу G:

$$G\langle v_1, v_5\rangle = v_1 e_1 v_7 e_3 v_3 e_6 v_4 e_5 v_8 e_2 v_2 e_4 v_6 e_9 v_5$$

2. Визначення матриці відстаней:

$$v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6 \quad v_7 \quad v_8 \quad e(v_i)$$

$$\begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \end{array}\begin{pmatrix} 0 & 5 & 2 & 3 & 4 & 4 & 1 & 4 \\ 5 & 0 & 3 & 2 & 2 & 1 & 4 & 1 \\ 2 & 3 & 0 & 1 & 2 & 2 & 1 & 2 \\ 3 & 2 & 1 & 0 & 1 & 1 & 2 & 1 \\ 4 & 2 & 2 & 1 & 0 & 1 & 3 & 2 \\ 4 & 1 & 2 & 1 & 1 & 0 & 3 & 2 \\ 1 & 4 & 1 & 2 & 3 & 3 & 0 & 3 \\ 4 & 1 & 2 & 1 & 2 & 2 & 3 & 0 \end{pmatrix} \begin{array}{c} 5 \\ 5 \\ 3 \\ 3 \\ 4 \\ 4 \\ 4 \\ 4 \end{array}$$

3. Знаходження діаметру D та радіусу R для графу G:

$$D(G) = max\{e(v_i)\} = 5$$

$$R(G) = min\{e(v_i)\} = 3$$

4. Визначення центральних та периферійних вершин:

Центральними вершинами в графі G є такі: v_3, v_4.

Периферійними вершинами в графі G є такі: v_1, v_2.

5. Визначення мостів графу G:

При проведенні перебору ребер графу, було виявлено 3 ребра, що є мостами: e_1, e_3, e_6 , які виділено червоним кольором на рис. 2 нижче. При видаленні будь-якого з них утворяться дві компоненти зв’язності.

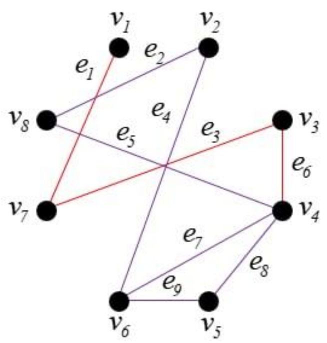
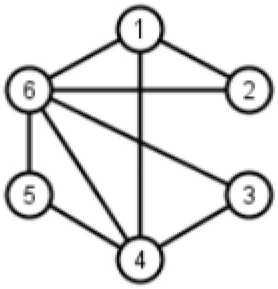
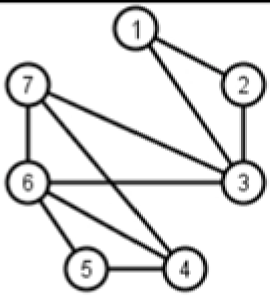


Рисунок 2. Граф з результатами перебору

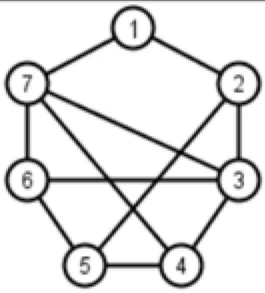
Варіанти



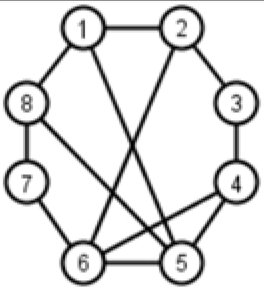
1.1



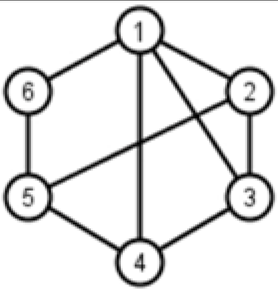
1.2



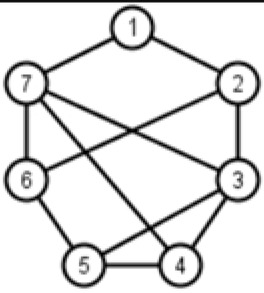
1.3



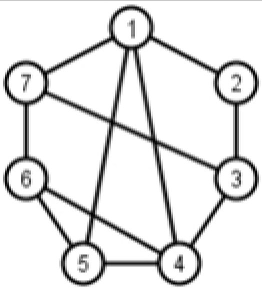
1.4



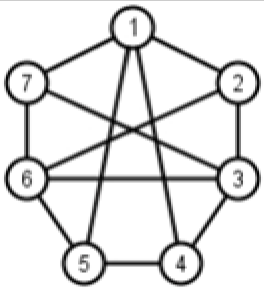
1.5



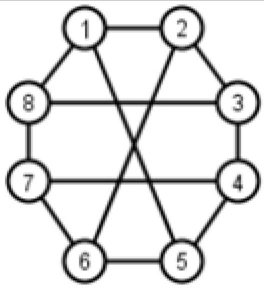
1.6



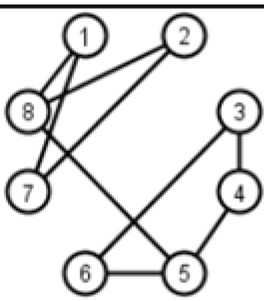
1.7



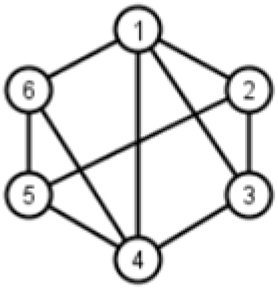
1.8



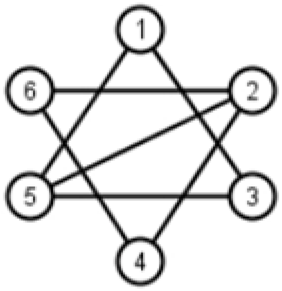
1.9



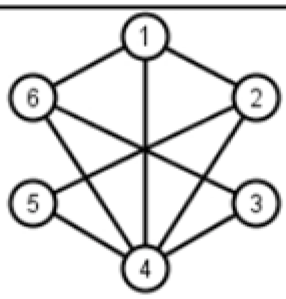
1.10



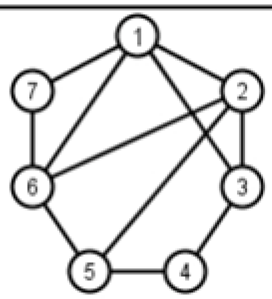
1.11



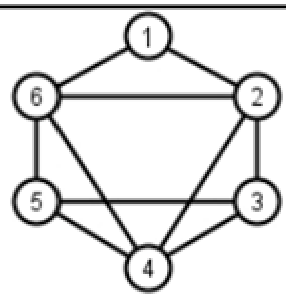
1.12



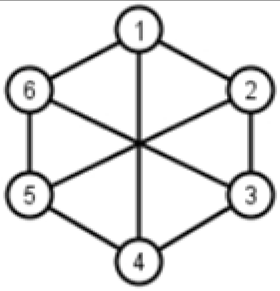
1.13



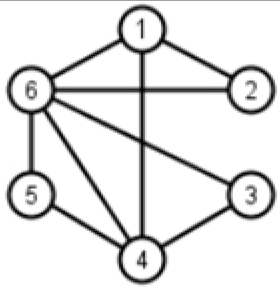
1.14



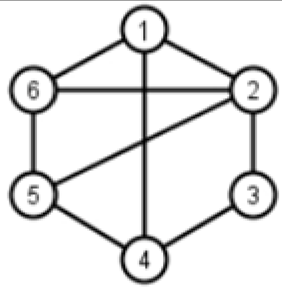
1.15



1.16



1.17



1.18

Контрольні запитання

1. Що називають неорієнтованим графом, або просто графом?
2. Що розуміють під ребрами, вершинами, граничними вершинами
3. ребра графа?
4. У якому разі ребро графа називають інцидентним вершині графа,
5. і навпаки?
6. Які ребра (вершини) графа називають суміжними, або сусідніми?
7. Як називають вершину графа, не інцидентну жодному ребру
8. (інцидентну тільки одному ребру)?
9. Яке ребро графа називають петлею?
10. Які ребра графа називають кратними, або паралельними?
11. Що таке простий ланцюг графу?
12. Який граф називається зв’язний?
13. Що таке міст графу?
14. Що таке ексцентриситет e(v) вершини v?
15. Що таке діаметр та радіус графа?

Лабораторний практикум №2

Розробка програми з ієрархією класів для опису операцій над неорієнтовними графами.

Мета роботи: здобути навики опису операцій над графами мовою програмування.

Завдання: розробити клас «граф», методи якого для заданих графів (відповідно варіанту) визначатимуть нові графи, які отримуються в результаті операцій над графами:$G_1 \cup G_2$ $G_1 \cap G_2$ $G_1 \oplus G_2$

Користувацькій інтерфейс має підтримувати можливість задати вихідні графи матрицею суміжності.

Приклад:

1. Пронумеруємо вершини та ребра заданих графів:

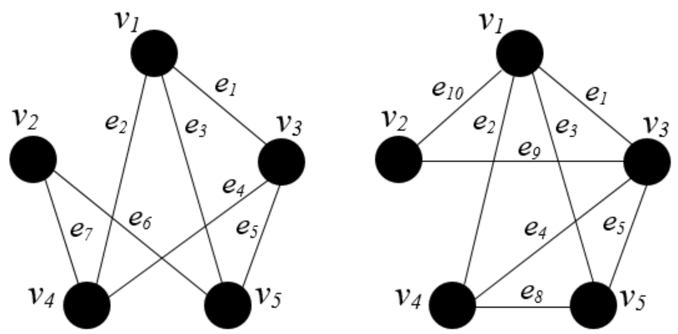


Рисунок 1. Пронумеровані графи G_1 і G_2 відповідно

2. Визначимо матриць суміжності для графів G_1 та G_2 відповідно:

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

3. Складаємо матрицю суміжності для об'єднання $G_1 \cup G_2$ та будуємо новий граф (рис. 2):

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

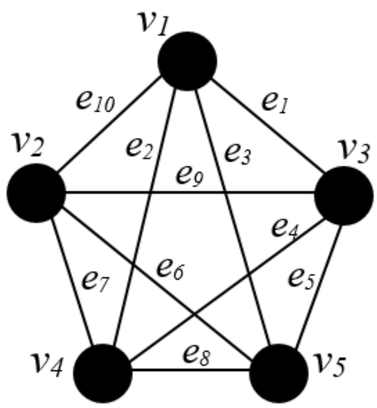


Рисунок 2 Граф об'єднання

a. Перетин $G_1 \cap G_2$:

Складаємо матрицю для перетину та будуємо за нею граф (рис. 3):

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{pmatrix}$$

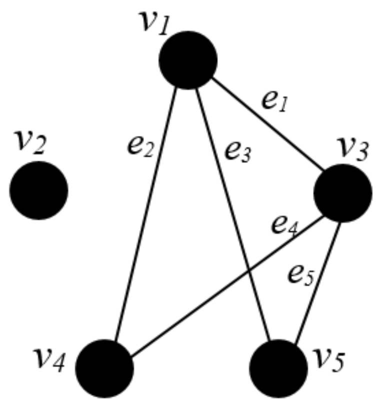


Рисунок 3. Граф перетину

b. Сума по модулю $G_1 \oplus G_2$:

Складаємо відповідну матрицю та будуємо за нею граф (рис. 4):

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix}$$

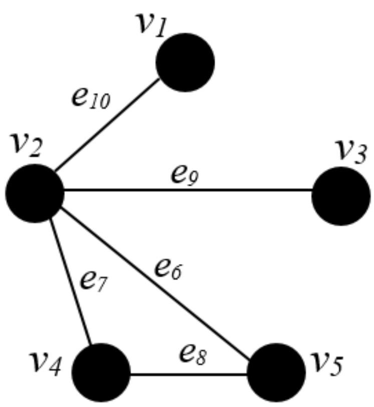


Рисунок 4 Граф після виконання операції суми по модулю

Варіанти

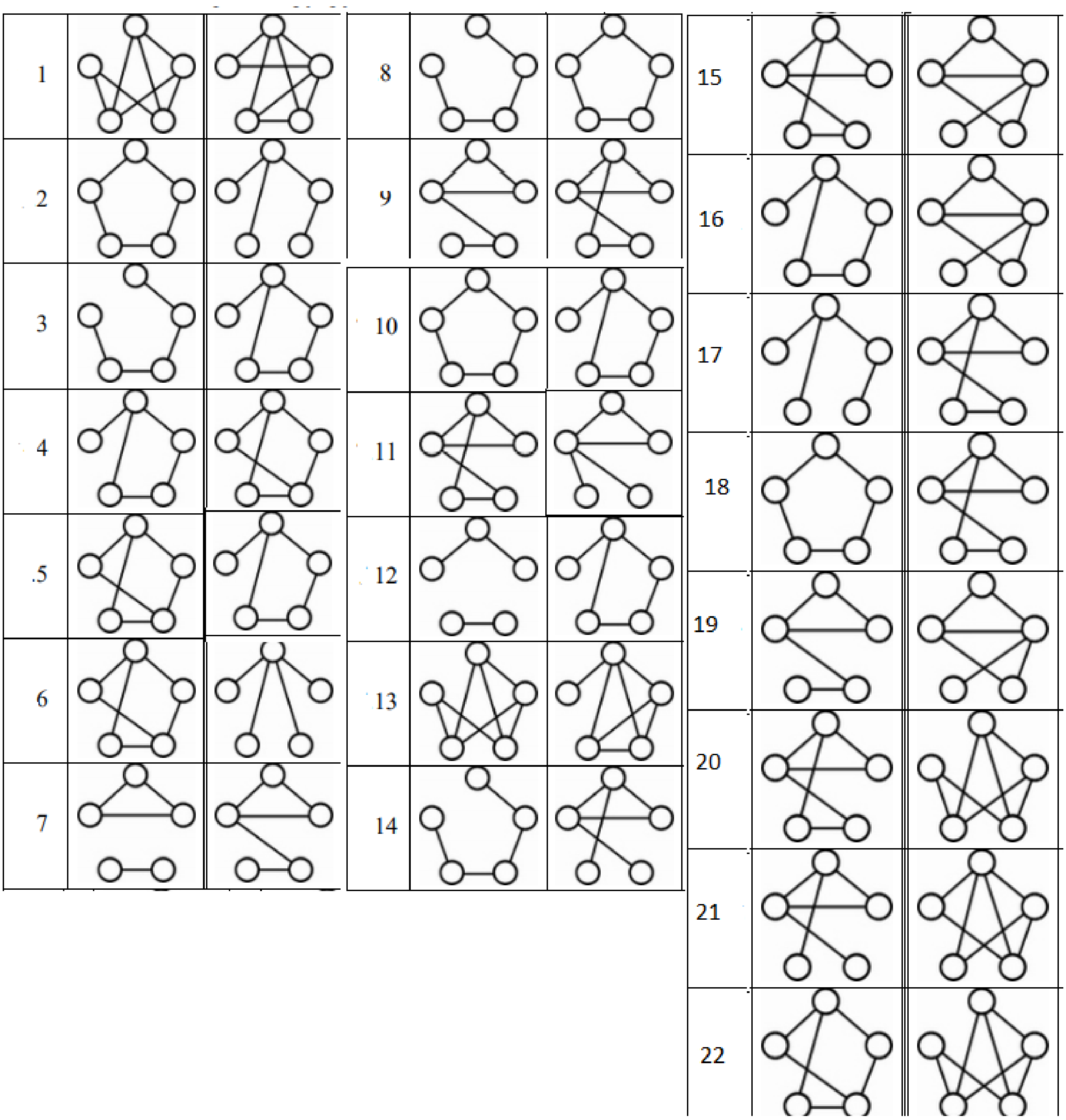


Рисунок 1. графи G_1 и G_2

Контрольні запитання

1. Який ланцюг (цикл) називають простим, або елементарним?
2. Який граф називають зв'язним (незв'язним)?
3. Що розуміють під компонентами зв'язності незв'язного графа?
4. Як формулюють постановку задачі про знаходження найкоротшого ланцюга між двома вершинами графа?
5. Як розв'язують задачу про знаходження найкоротшого ланцюга
6. методом індексації (зважування) вершин графа?

Лабораторний практикум №3

Розробка програми для опису об'єкту типу Ейлерові та гамільтонові графи.

Мета роботи: здобути навики опису складних об'єктів типу Ейлерові та Гамільтонові графи мовою програмування.

Завдання: розробити клас «граф», методи якого для заданого графу визначатимуть характеристики 1-5.

1. Ейлеровий цикл графу (якщо є)
2. Гамільтоновий цикл графу (якщо є).
3. Остов (каркас).
4. Матрицю С фундаментальних циклів.
5. Матрицю К фундаментальних перерізів(коциклів).

Приклад:

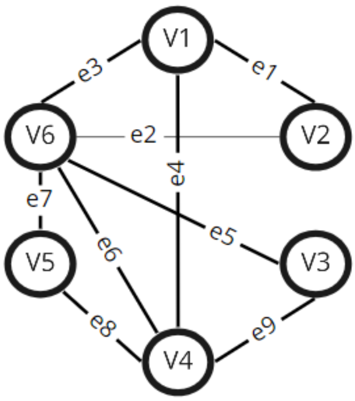


Рисунок 1. Вихідний граф

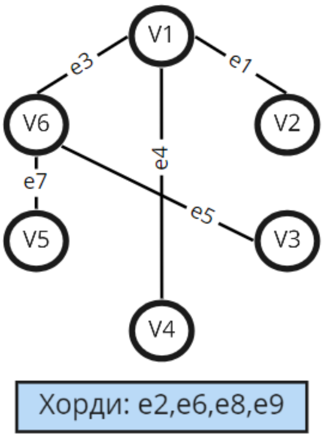
В даному графі немає Ейлерового циклу, так як степені вершин 1 і 6 – непарні (дорівнюють 3 та 5 відповідно).

Гамільтонового циклу в графі також немає.

За теоремою Дірака – гамільтоновим графом є такий граф, у якого виконуються наступні умови: $n \geq 3$, де n – кількість вершин графа; $\delta \geq n/2$, де δ – мінімальна степінь вершини графа.

Параметри для заданого графа G такі: $n=6$; $\delta=2$, а підставивши їх в нерівність отримаємо $2 \geq 3$. Оскільки умова теореми Дірака не виконується, то граф G не є гамільтоновим.

Для заданого графу зобразимо остов (каркас):



1. Знайдемо матрицю C фундаментальних циклів графа G:

Складаємо матрицю, що містить 4 цикли C (за кількістю хорд):

$$\begin{array}{c} \\ C_1 \\ C_2 \\ C_3 \\ C_4 \end{array} \begin{array}{c} \begin{array}{cccc|ccccc} e_2 & e_6 & e_8 & e_9 & e_1 & e_3 & e_4 & e_5 & e_7 \end{array} \\ \left(\begin{array}{cccc|ccccc} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \end{array}\right) \end{array}$$

2. Знайдемо матрицю К фундаментальних перерізів (коциклів) графа G:

Матриця заповнюється таким чином: першій частині (К1) ставиться у відповідність транспонована друга частина (С2) матриці C, а друга частина (К2) заповнюється додаванням рівно однієї гілки дерева до кожного фундаментального перерізу. Матриця фундаментальних перерізів має вигляд:

$$e_2 \quad e_6 \quad e_8 \quad e_9 \quad e_1 \quad e_3 \quad e_4 \quad e_5 \quad e_7$$

$$
\begin{array}{c}
K_1 \\ K_2 \\ K_3 \\ K_4 \\ K_5
\end{array}
\left(\begin{array}{cccc|ccccc}
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right)
$$

3. Перевірка умови ортогональності матриці інцидентності до матриці фундаментальних циклів $B \cdot C^T = 0$:

Щоб перевірити виконання умови ортогональності необхідно скласти матрицю інцидентності *B* та транспонувати матрицю фундаментальних циклів *C*.

$$
B = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0
\end{pmatrix}
\quad
C = \begin{pmatrix}
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}
$$

$$
B \times C = \begin{pmatrix}
1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0
\end{pmatrix}
\times
\begin{pmatrix}
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 \\
0 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}
=
\begin{pmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
$$

Оскільки отримана матриця повністю складається з нулів, то умова ортогональності матриці інцидентності до матриці фундаментальних циклів виконується.

Контрольні запитання

1. Чи може бути Еейлерів граф ациклічним?
2. Чи може бути степінь вершини непарною у Ейлерового графа?
3. Чи може степінь вершини бути непарною у напів- Ейлеровому графі?
4. У чому відмінність між Ейлеровим і Гамільтоновим графами?
5. Які умови існування Гамільтонового графа знаєте?
6. Чи обов'язково Гамільтонів граф має бути зв'язним? Чому?
7. Чи має бути граф Оре Гамільтоновим графом?
8. Чи має бути Гамільтонів граф графом Оре?
9. Напишить умову ортогональності матриці фундаментальних циклів до матриці перерізів.

Лабораторний практикум №4

Розробка програми для опису складного об'єкту типу орієнтовний граф.

Мета роботи: здобути навики опису складних об'єктів типу орієнтовний граф.

Завдання: розробити клас «граф», методи якого для заданих орієнтовних графів (відповідно варіанту) визначатимуть пів-ступінь виходу $d^+(v_i)$ та пів-ступінь входу $d^-(v_i)$. Будуватимуть новий граф ізоморфний даному. Дозволять розфарбувати вершини та визначити хроматичне число. Користувацькій інтерфейс має підтримувати можливість задати вихідний орієнтовний граф матрицею суміжності.

Приклад:

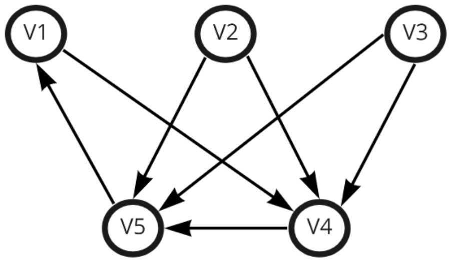


Рисунок 1. Вихідний граф

1. Визначення для кожної вершини пів-ступеней виходу $d^+(V_i)$ та пів-ступеней входу $d^-(V_i)$:

$d^+(V_1) = 1$ $d^-(V_1) = 1$

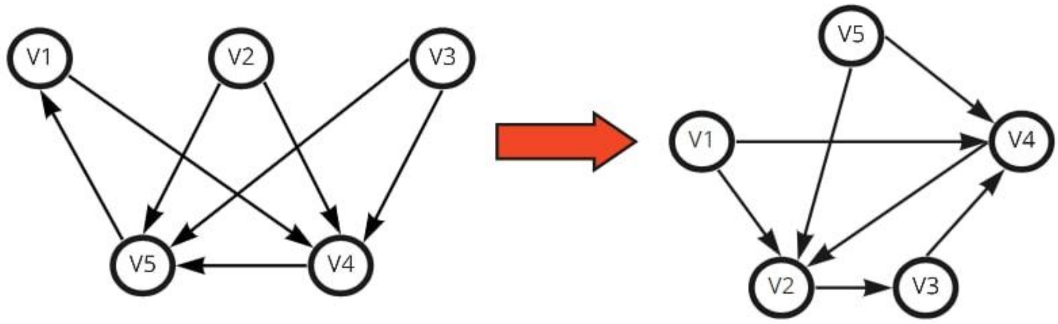
$d^+(V_2) = 2$ $d^-(V_2) = 0$

$d^+(V_3) = 2$ $d^-(V_3) = 0$

$d^+(V_4) = 1$ $d^-(V_4) = 3$

$d^+(V_5) = 1$ $d^-(V_5) = 3$

2. Побудова графу та доведення його ізоморфності даному:



Щоб довести, що перетворення відповідає критеріям ізоморфності, побудуємо матрицю суміжності для початкового графу та спробуємо отримати з неї матрицю другого графу.

Матриця суміжності початкового графу:

	1	**2**	**3**	**4**	**5**
1	0	0	0	1	0
2	0	0	0	1	1
3	0	0	0	1	1
4	0	0	0	0	1
5	1	0	0	0	0

Крок 1:

	1	**2**	**3**	**4**	**5**
1	0	0	0	1	1
2	1	0	0	0	0
3	0	0	0	1	0
4	0	0	0	0	1
5	0	0	0	1	1

Крок 2:

	1	**2**	**3**	**4**	**5**
1	0	1	0	1	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	1	0	0	0
5	0	1	0	1	0

Матриця другого графу:

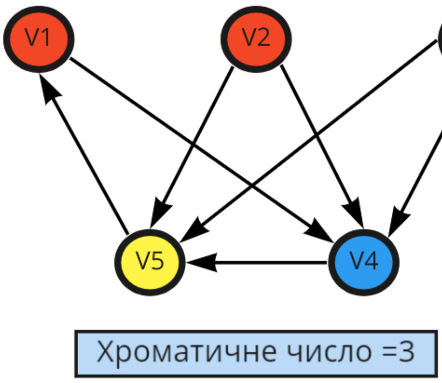
	1	**2**	**3**	**4**	**5**
1	0	1	0	1	0
2	0	0	1	0	0
3	0	0	0	1	0

4	0	1	0	0	0
5	0	1	0	1	0

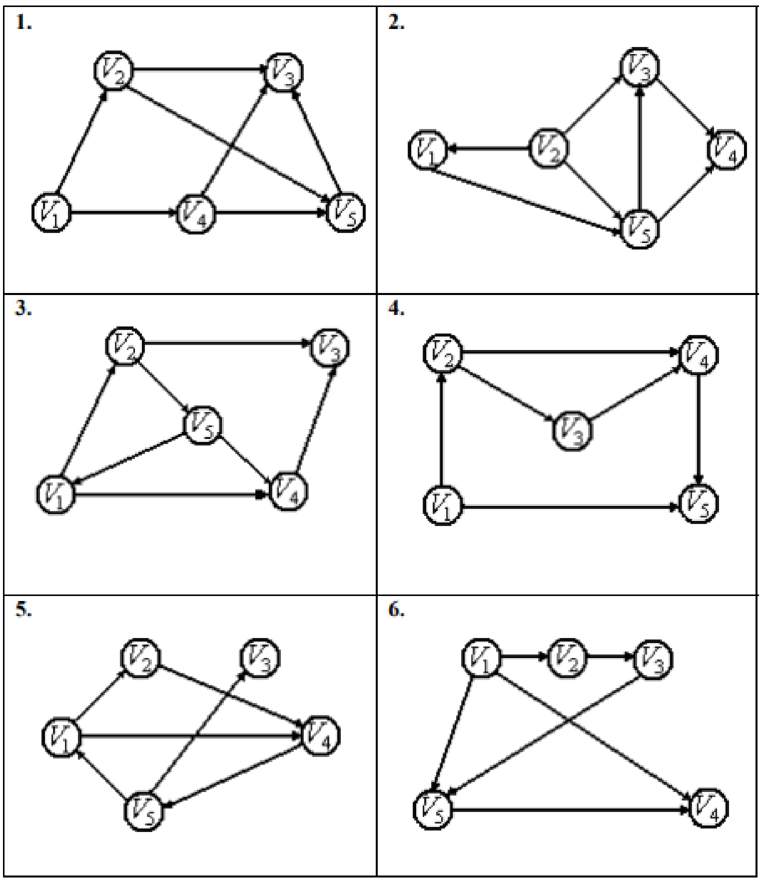
Ізоморфність графів доведено.

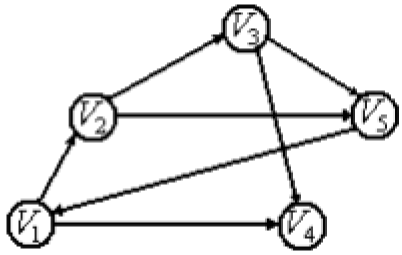
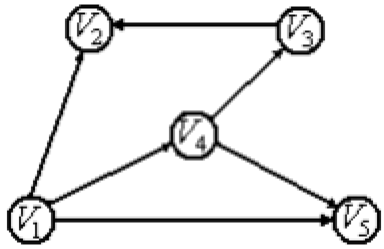
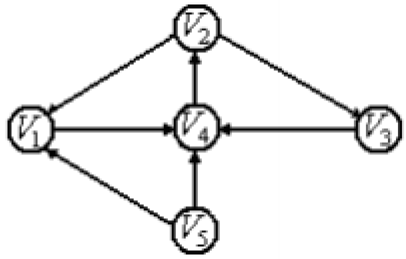
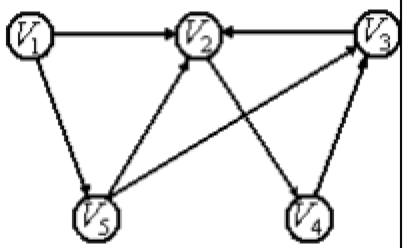
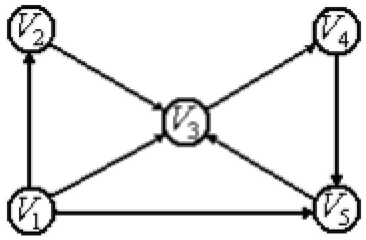
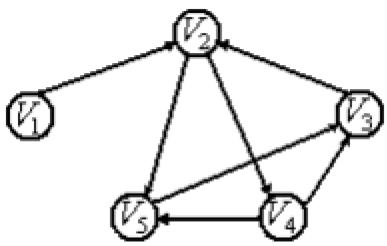
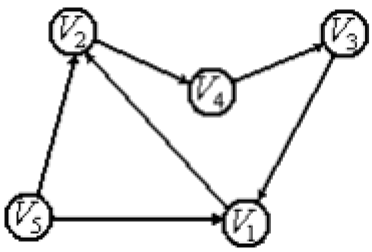
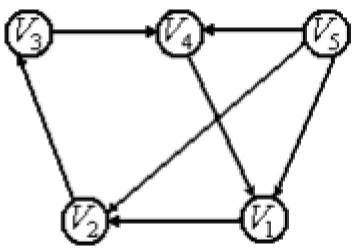
3. Розфарбування вершин та визначення хроматичного числа:

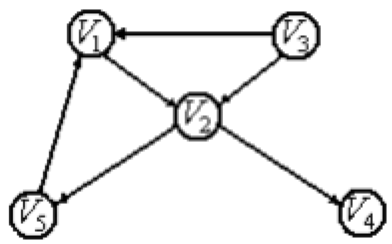
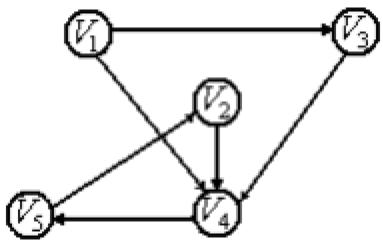
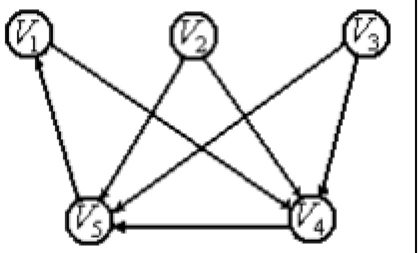
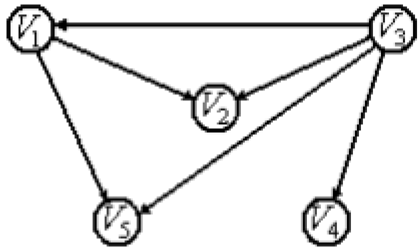
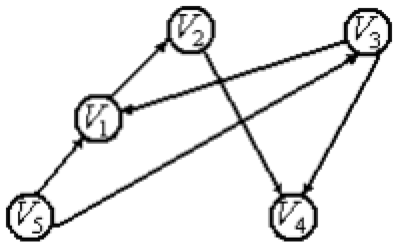
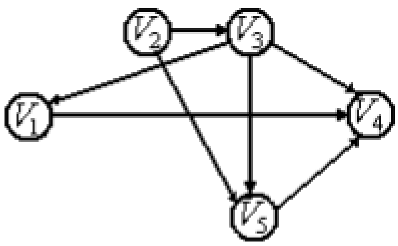
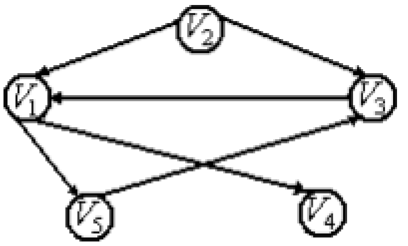
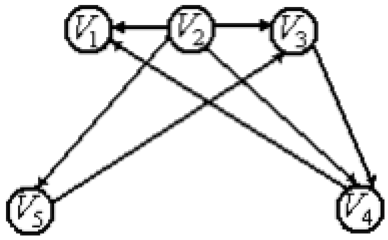
Найменша кількість кольорів, якою можна розфарбувати вершини даного графа – 3:



Варіанти



7. 	**8.** 
9. 	**10.** 
11. 	**12.** 
13. 	**14.** 

15. 	**16.** 
17. 	**18.** 
19. 	**20.** 
21. 	**22.** 

Контрольні запитання

1. Який граф називають ізоморфний даному?
2. Який граф називають планирним?
3. Які є способи задання ОГ і в чому вони полягають?
4. Що таке "основа" ОГ і який ОГ називають сильнозв'язним (слабкозв'язним, односторонньо зв'язним, порожнім, тривіальним)?
5. Як визначають матрицю інцидентностей (суміжності вершин) ОГ?
6. Що називають матрицею досяжності та контрдосяжності ОГ?
7. Що таке «правильне» розфарбування графа?

Лабораторний практикум №5

Розробка програми для опису складного об’єкту типу сітковий граф та розв’язку на ньому задачі складання розкладу.

Мета роботи: здобути навики опису складних об’єктів типу сітковий граф та розв’язку на ньому задачі складання розкладу.

Завдання:

У навчальному центрі необхідно провести заняття з математики, фізики, хімії, біології в групах A, B, C. Заняття проводяться викладачами K, L, M. Кожне заняття проводиться протягом двох годин, включаючи перерви. В центрі є три аудиторії, які вміщають лише одну з групу для занять з фізики і хімії обладнана одна з цих аудиторій. Чи можна провести всі необхідні заняття протягом 6 годин? Якщо можна, скласти розклад. Якщо не можна, то скласти розклад, як провести ці заняття за мінімальну кількість годин.

Вихідна умова відповідно варіанту.

Приклад:

Таблиця 1. Вихідна умова

Предмет	Група			Викладач		
	A	B	C	K	L	M
математика	+	+		+		
фізика	+	+			+	
хімія		+	+		+	
біологія	+		+			+

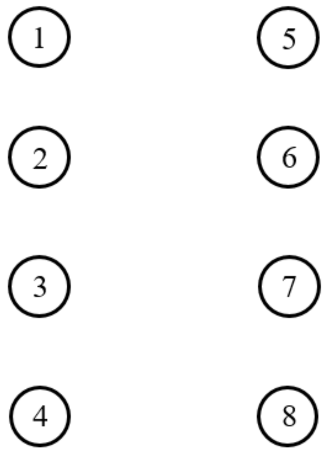
Хід виконання

Скласти оптимальний розклад, з урахуванням всіх критеріїв, можна застосовуючи алгоритм розфарбовування графу. Для цього побудуємо граф, де вершинами є окремі заняття, що з’єднані ребрами, коли їх неможливо провести одночасно. Кількість вершин це загальна кількість занять (всього 8).

Спочатку необхідно присвоїти кожній вершині заняття, якому вона відповідає. Вважатимемо, що аудиторії 1, 2 для загального призначення, а аудиторія 3 – для занять з хімії та фізики, згідно з умовою задачі. Отже:

Вершина	Предмет	Група	Викладач	Аудиторія
v_1	математика	A	K	1
v_2	фізика	A	L	3
v_3	біологія	A	M	2
v_4	математика	B	K	1
v_5	фізика	B	L	3
v_6	хімія	B	L	3
v_7	хімія	C	L	3
v_8	біологія	C	M	2

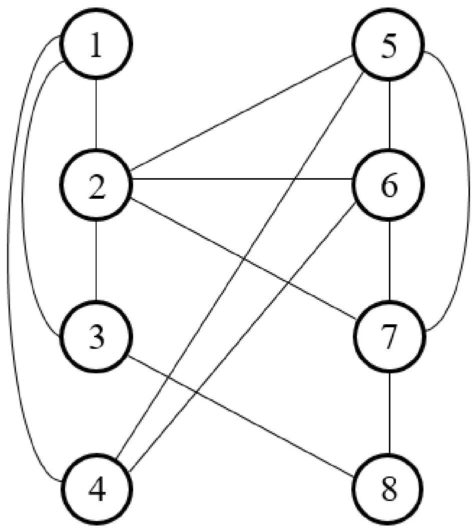
Розмістимо вершини на графі в зручному порядку:



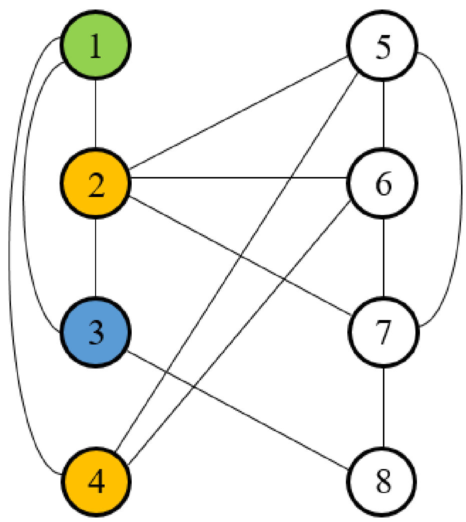
З'єднаємо їх лініями відповідно до умови: заняття які не можуть бути проведені одночасно. Такими заняттями є:

- Предмети, що викладає один викладач (наприклад, одночасно хімію і фізику провести неможливо, оскільки їх викладає 1 людина);
- Один предмет, що проводиться у декількох груп (наприклад, є два заняття біології у двох груп);
- Заняття однієї групи – одночасно кожна група може мати тільки одне заняття, так само як і викладач.

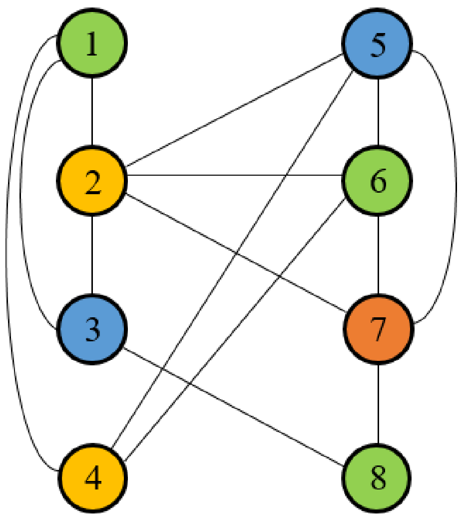
Отже, побудований граф має такий вигляд:



Переходимо до його розфарбування. Нехай v_1 отримає перший колір, тоді v_2 та v_4 – другий, а v_3 може бути лише 3 кольору, оскільки вона з'єднана з v_1 і v_2.



Вершина v_8 та v_6 також можна зафарбувати першим кольором. Тоді v_5 отримає третій колір, а v_7 – четвертий:



Оскільки в даному графі мінімальна кількість кольорів – 4, це означає, що неможливо провести всі заняття за 6 годин (3 заняття), а лише за 8 годин (що дорівнює 4 заняттям).

Складемо оптимальний розклад для 4 занять:

		Урок 1	Урок 2	Урок 3	Урок 4
Групи	**A**	Математика, викладач K, аудиторія 1	Біологія, викладач M, аудиторія 2	Фізика, викладач L, аудиторія 3	-
	B	Фізика, викладач L, аудиторія 3	Математика, викладач K, аудиторія 1	-	Хімія, викладач L, аудиторія 3
	C	Біологія, викладач M, аудиторія 2	Хімія, викладач L, аудиторія 3	-	-

Або розклад за графом:

		Урок 1	Урок 2	Урок 3	Урок 4
Групи	**A**	Математика, викладач K, аудиторія 1	Фізика, викладач L, аудиторія 3	Біологія, викладач M, аудиторія 2	-

	B	Хімія, викладач L, аудиторія 3	Математика, викладач K, аудиторія 1	Фізика, викладач L, аудиторія 3	-
	C	Біологія, викладач M, аудиторія 2	-	-	Хімія, викладач L, аудиторія 3

Варіанти

Варіант1

Предмет	Група			Викладач		
	A	B	C	K	L	M
математика		+	+			+
фізика	+	+			+	
хімія		+	+		+	
біологія	+		+			+

Варіант2

Предмет	Група			Викладач		
	A	B	C	K	L	M
математика	+		+	+		
фізика	+	+	+		+	
хімія		+			+	
біологія	+		+			+

Варіант3

Предмет	Група			Викладач		
	A	B	C	K	L	M
математика		+	+	+		
фізика	+	+			+	
хімія			+		+	
біологія	+	+	+			+

Варіант4

Предмет	Група	Викладач

	A	B	C	K	L	M
математика		+	+	+		
фізика	+			+		
хімія	+	+	+		+	
біологія	+		+			+

Варіант5

Предмет	Група			Викладач	
	A	B	C	K	L
математика	+		+	+	
фізика	+	+		+	
хімія		+			+
біологія	+		+		+

Варіант6

Предмет	Група			Викладач	
	A	B	C	K	L
математика		+	+	+	
фізика	+		+	+	
хімія		+			+
біологія	+	+	+		+

Варіант7

Предмет	Група			Викладач		
	A	B	C	K	L	M
математика		+	+	+		
фізика	+				+	
хімія	+		+		+	
біологія	+	+	+			+

Варіант8

Предмет	Група			Викладач	
	A	B	C	K	L
математика	+	+		+	
фізика	+		+	+	
хімія		+	+		+
біологія	+				+

Варіант9

Предмет	Група			Викладач		
	A	B	C	K	L	M
математика		+	+	+		
фізика	+		+		+	
хімія		+			+	
біологія	+		+			+

Варіант10

Предмет	Група			Викладач	
	A	B	C	K	L
математика		+		+	
фізика	+		+	+	
хімія		+	+		+
біологія	+	+			+

Контрольні запитання:

1. Що називають сітковим графіком комплексу робіт (проєкту)?
2. Що називають раннім (пізнім) строком звершення події, резервом часу, критичним часом, критичним шляхом на СГ?
3. Що має передувати (щодо відомостей) побудові СГ?
4. Як формулюють постановку задачі про знаходження критичного часу та критичного шляху на СГ?
5. Із яких етапів складається розв'язання задачі про критичний час і критичний шлях на СГ?

Лабораторний практикум №6

Розробка програми для розв'язку «комівояжера» з матрицею C відстаней заданої в табл.1 методом гілок і границь. Відповідь має включати повне дерево розгалужень та знайдений цикл.

Мета роботи: здобути навики розв'язку задач на графах.

Приклад:

Таблиця 1. Задана матриця відстаней

	1	**2**	**3**	**4**	**5**
1	∞	18	16	16	10
2	18	∞	20	23	28
3	21	25	∞	10	18
4	13	19	29	∞	12
5	24	13	15	17	∞

Знаходимо мінімальні значення кожного рядка та записуємо їх в колонку d_i. Виконуємо редукцію по рядках матриці – віднімаємо від кожного значення рядка мінімальне зі значень даного рядка.

	1	**2**	**3**	**4**	**5**	**d(i)**
1	∞	8	6	6	0	**10**
2	0	∞	2	5	10	**18**
3	11	15	∞	0	8	**10**
4	1	7	17	∞	0	**12**
5	11	0	2	4	∞	**13**

Після чого в рядок d_j виписуємо мінімальні значення стовпців й також проводимо по ним редукцію.

	1	**2**	**3**	**4**	**5**	**d(i)**
1	∞	8	4	6	0	**10**
2	0	∞	0	5	10	**18**
3	11	15	∞	0	8	**10**
4	1	7	15	∞	0	**12**
5	11	0	0	4	∞	**13**
d(j)	**0**	**0**	**2**	**0**	**0**	**65**

Отже, H_0=65.

Переходимо до 1 кроку розв'язання. Оцінюємо нулі матриці, просумувавши мінімальні значення стовпця та рядка даного елементу.

	1	**2**	**3**	**4**	**5**
1	∞	8	4	6	0(4)
2	0(1)	∞	0(0)	5	10
3	11	15	∞	0(12)	8
4	1	7	15	∞	0(1)
5	11	0(7)	0(0)	4	∞

Отже, на даному етапі розглядатиметься відрізок (3, 4) за такими варіантами:

1) відрізок (3, 4) не включається в маршрут:

Комірку даного відрізку приймаємо за нескінченність і обраховуємо суму мінімальних елементів.

	1	**2**	**3**	**4**	**5**	**d(i)**
1	∞	8	4	6	0	**0**
2	0	∞	0	5	10	**0**
3	11	15	∞	∞	8	**8**
4	1	7	15	∞	0	**0**
5	11	0	0	4	∞	**0**
d(j)	**0**	**0**	**0**	**4**	**0**	**12**

Виконуємо розрахунок H за формулою:

$$H(a^*, b^*) = H_0 + \sum d$$

$$H(3^*, 4^*) = 65 + 12 = 77$$

2) відрізок (3, 4) включається в маршрут:

Видаляємо з матриці відповідні рядок та стовпець, а також позначаємо зворотний елемент нескінченністю, щоб виключити цей шлях з маршруту. Виконавши, запишемо нову матрицю, що складається з усіх елементів, що залишились невидаленими. Також знаходимо мінімальні елементи, а їх суму додаємо до значення поточного маршруту.

	1	**2**	**3**	**5**	**d(i)**
1	∞	8	4	0	**0**
2	0	∞	0	10	**0**
4	1	7	∞	0	**0**
5	11	0	0	∞	**0**
d(j)	**0**	**0**	**0**	**0**	0

$H(3,4) = 65$

Переходимо до 2 кроку. Для матриці, отриманої на попередньому етапі виконуємо оцінку нулів:

	1	**2**	**3**	**5**
1	∞	8	4	0(4)
2	0(1)	∞	0(0)	10
4	1	7	∞	0(1)
5	11	0(7)	0(0)	∞

Досліджуваним відрізком на даному кроці є (5, 2):

1) відрізок (5, 2) не включається в маршрут:

	1	**2**	**3**	**5**	**d(i)**
1	∞	8	4	0	**0**
2	0	∞	0	10	**0**
4	1	7	∞	0	**0**
5	11	∞	0	∞	**0**
d(j)	**0**	**7**	**0**	**0**	**7**

Суму мінімальних елементів додаємо до маршруту:

$$H(5^*, 2^*) = 65 + 7 = 72$$

2) відрізок (5, 2) включається в маршрут, то:

Обраховуємо мінімальні значення для отриманої матриці.

	1	**3**	**5**	**d(i)**
1	∞	4	0	**0**
2	0	0	10	**0**
4	1	∞	0	**0**
d(j)	**0**	**0**	**0**	**0**

$H(5,2) = 65$

На третьому кроці так само проводимо операції над отриманою на попередньому етапі матрицею. Спочатку виконуємо оцінку нулів.

	1	**3**	**5**
1	∞	4	0(4)
2	0(1)	0(4)	10
4	1	∞	0(1)

Розглядаємо відрізок (1, 5):

1) відрізок (1, 5) не включається в маршрут:

Обраховуємо мінімальні значення матриці:

	1	**3**	**5**	**d(i)**
1	∞	4	∞	**4**
2	0	0	10	**0**
4	1	∞	0	**0**
d(j)	**0**	**0**	**0**	**4**

$H(1^*, 5^*) = 65 + 4 = 69$

2) відрізок (1, 5) включається в маршрут:

	1	**3**	**d(i)**
2	0	0	**0**
4	1	∞	**1**
d(j)	**0**	**0**	**1**

$H = 65 + 1 = 66$

Виконаємо наступний крок розв’язку, оцінивши останні два нулі матриці:

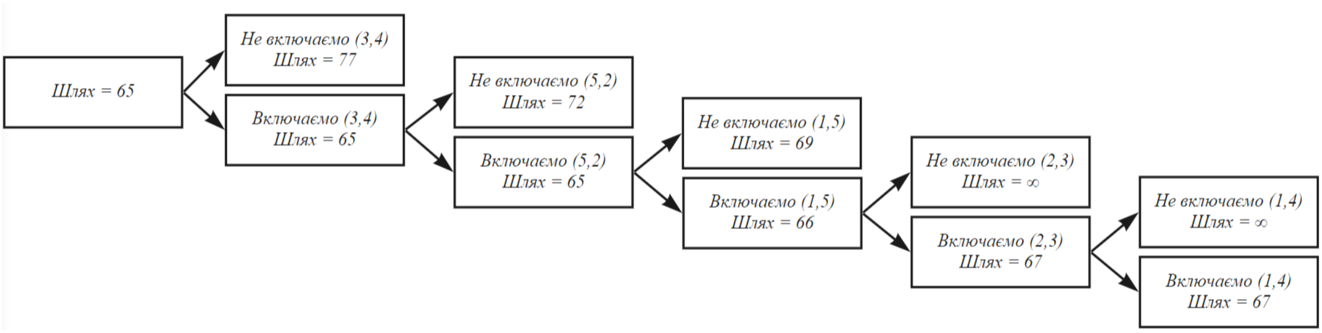
	1	**3**
2	0(1)	0(∞)
4	1	∞

Нескінченність є більшою за 1, тож викреслюємо з матриці відповідний рядок і стовпець і додаємо до маршруту отриманий відрізок (2, 3). Шлях H збільшується на 1, і дорівнює 67.

	1
4	1

Останній відрізок (4, 1), що залишився, також додаємо до маршруту.

Дерево розв’язків має наступний вигляд:



Отже, знайдений цикл має вигляд: 1 - 5 - 2 - 3 – 4 - 1. Таким чином розв’язок задачі складається з найменших можливих відстаней між містами.

Варіанти

Номер варіанта						Номер варіанта					
1	∞	27	26	23	25	2	∞	42	49	33	45
	29	∞	31	32	28		39	∞	38	38	41
	22	34	∞	22	31		46	54	∞	42	38
	29	20	26	∞	23		58	53	45	∞	55
	28	43	32	26	∞		45	54	61	60	∞
3	∞	34	33	25	28	4	∞	29	29	33	40
	28	∞	25	27	22		32	∞	30	40	37
	37	28	∞	30	31		44	28	∞	32	26
	30	29	28	∞	34		47	38	44	∞	47
	39	38	41	40	∞		44	37	33	32	∞
5	∞	52	46	52	43	6	∞	19	22	18	21
	48	∞	44	53	43		22	∞	22	22	21
	53	59	∞	54	48		24	19	∞	24	27
	52	44	55	∞	42		23	27	23	∞	22
	54	57	57	51	∞		23	30	25	29	∞
7	∞	57	38	38	50	8	∞	43	49	32	31
	55	∞	45	52	32		31	∞	41	52	42
	35	43	∞	48	57		42	50	∞	28	51
	37	47	56	∞	37		32	39	30	∞	39
	46	35	61	61	∞		36	54	42	53	∞
9	∞	32	32	36	38	10	∞	30	24	27	21
	45	∞	29	38	39		28	∞	26	20	29
	34	31	∞	35	36		23	32	∞	25	38
	32	34	42	∞	28		24	22	24	∞	34
	37	48	41	34	∞		31	28	35	36	∞
11	∞	14	23	24	16	12	∞	48	44	38	46
	22	∞	12	19	22		46	∞	51	43	30
	16	24	∞	17	21		40	39	∞	45	34
	23	15	29	∞	15		56	50	53	∞	49

13	∞	26	25	33	32	14	∞	16	18	23	19
	36	∞	36	24	29		14	∞	28	22	16
	42	33	∞	37	39		15	20	∞	24	25
	24	41	42	∞	36		26	19	25	∞	17
	37	27	38	41	∞		27	29	21	20	∞
15	∞	19	10	26	29	16	∞	24	18	12	19
	18	∞	14	21	27		23	∞	38	22	26
	14	19	∞	10	12		35	42	∞	24	15
	13	21	25	∞	12		24	24	28	∞	27
	13	13	15	17	∞		26	27	21	20	∞
17	∞	18	16	16	10	18	∞	30	27	13	16
	18	∞	20	23	28		14	∞	18	22	17
	21	25	∞	10	18		25	42	∞	34	25
	13	19	29	∞	12		20	21	26	∞	37
	24	13	15	17	∞		21	27	21	19	∞

Контрольні запитання

1. Яка мета алгоритму Дейкстри?
2. Які приклади зважених графів знаєте?
3. Яка послідовність кроків в алгоритму пошуку в ширину? Наведіть приклади використання алгоритму пошуку.
4. Яка послідовність кроків в алгоритму пошуку в глибину? Наведіть приклади використання алгоритму пошуку в глибину.

Лабораторний практикум №7

Розробка програми для визначення максимальний потоку та його розподіл за дугами за алгоритмом Форда-Фалкерсона.

Мета роботи: здобути навики розв’язку задач на графах.

Приклад:

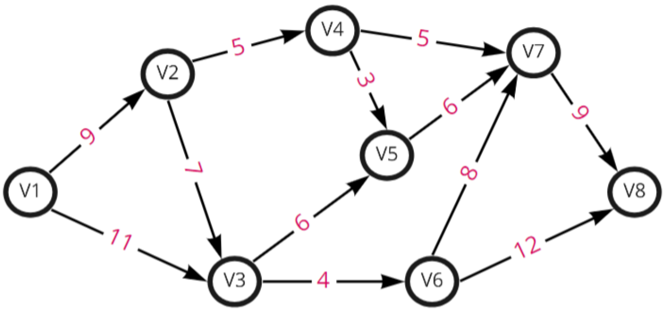
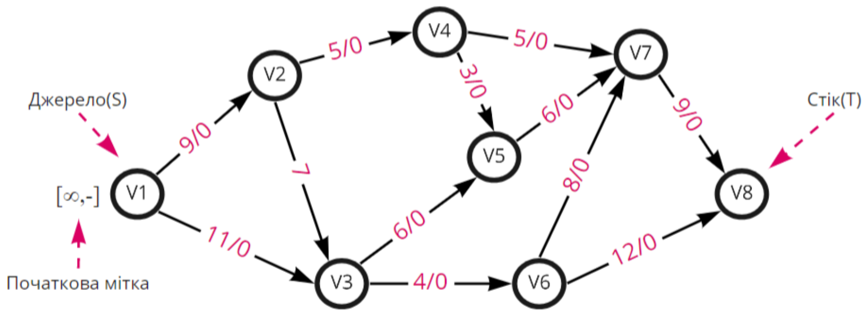


Рисунок 1. Вихідний граф

Для початку підготуємо граф, тобто виконаємо його ініціалізацію. Для цього першому вузлу v_0 додаємо мітку початку. Для всіх ребер позначимо залишкову пропускну спроможність рівній максимальній пропускній спроможності.



Алгоритм Форда-Фалкерсона складається з декількох кроків, більшість з яких будуть виконуватись на кожній ітерації в процесі пошуку:

Крок 1 – Приймаємо $i = 0$ та переходимо до другого кроку.

Крок 2 – Приймаємо множину S_i, як множину вузлів, в які можна перейти з вузла i по ребру, яке має залишкову пропускну знатність вище 0. Якщо ця множина пуста, то переходимо до четвертого кроку, а інакше до третього.

Крок 3 – В множині S_i обираємо такий вузол (k), ребро якого має найбільшу залишкову пропускну здатність (c) з усіх варіантів. Тоді надаємо даному вузлу мітку [a_i, i], де a_i це залишкова пропускна здатність ребра, а i – номер вузла, від якого йде ребро до даного вузла. Якщо міткою позначається кінцевий вузол (n) то переходимо до кроку 5, інакше приймаємо $i = k$ і переходимо до другого кроку.

Крок 4 – Якщо i дорівнює 1 то прокласти шлях неможливо і необхідно перейти до кроку 6, інакше, якщо ні, то знаходимо вузол r, що передує вузлу i, й видаляємо вузол i з множини вузлів, суміжних з даним вузлом r. Приймаємо $i = r$ та переходимо до кроку 2.

Крок 5 – Для знайденого шляху визначаємо максимальний потік, що проходить по даному шляху. Максимальний потік f – рівний найменшому зі значень залишкових пропускних здатностей ребер, що входять до даного шляху. Залишкові пропускні здатності всіх ребер даного шляху зменшуються на величину максимального потоку, а значення використаних пропускних здатностей цих ребер збільшуються на ту ж величину. Після цього прибираємо всі мітки біля вузлів, окрім першої та переходимо на крок 1 для пошуку нового шляху.

Крок 6 – При m знайдених шляхах, обчислюємо значення максимального потоку за формулою:

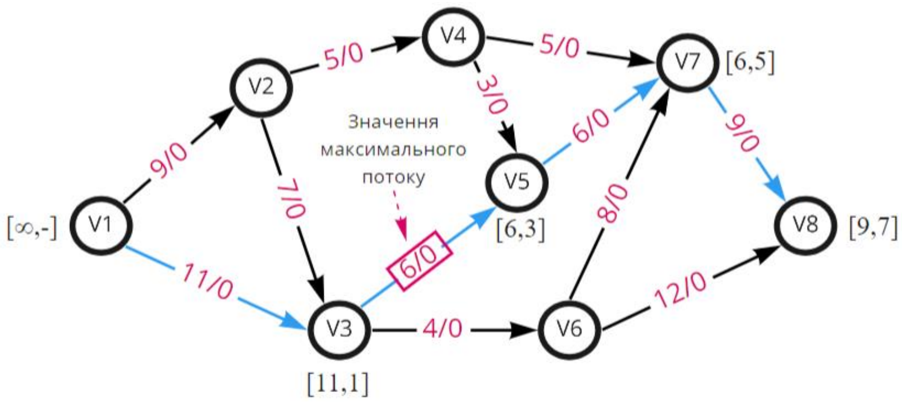
$$F = f_1 + f_2 + \ldots + f_m .$$

Визначивши кроки алгоритму, яким ми будемо слідувати, можна переходити до виконання розв'язку. Будемо записувати значення, отримані згідно кроків алгоритму, супроводжуючи їх графічним відображенням.

Для зручності кожен побудований шлях відокремлюється від інших ітерацією, тобто кожна ітерація відповідає одному конкретному шляху в графі.

Ітерація 1

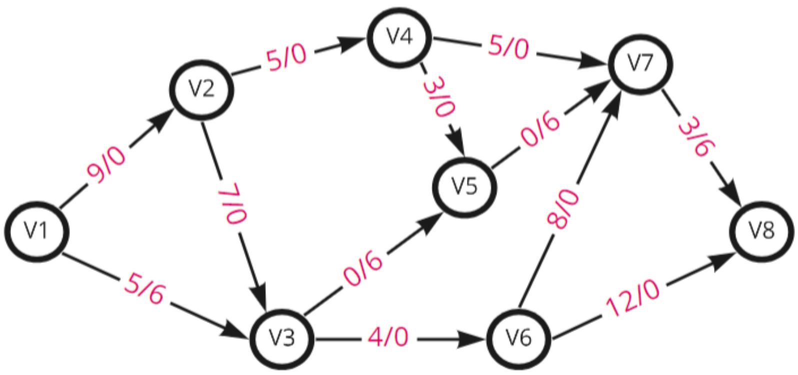
Зобразимо шлях на графі:



Значення максимального потоку для даного шляху – 6.

Віднімаємо дане значення від значень *c* задіяних ребер. Після цього прибираємо всі мітки біля вузлів, окрім першого.

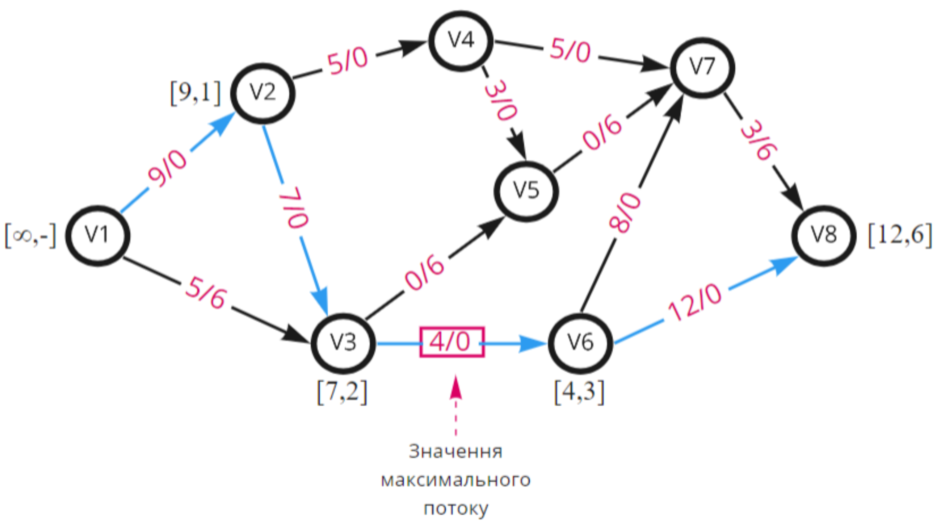
Перерахунок №1



Можемо переходити до наступної ітерації.

Ітерація 2

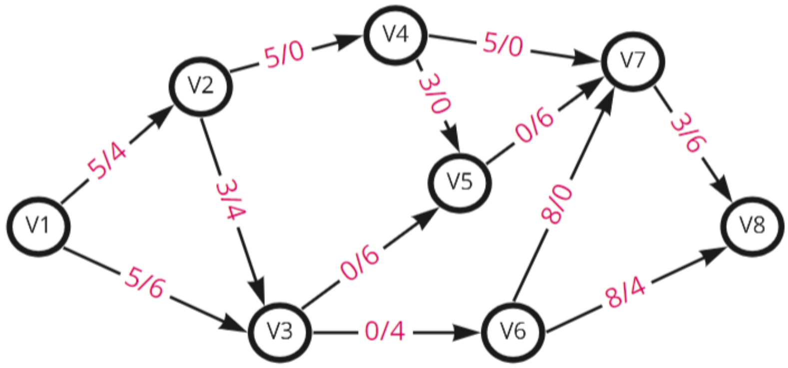
Зобразимо шлях на графі:



Значення максимального потоку для даного шляху – 4.

Віднімаємо дане значення від значень *с* задіяних ребер. Після цього прибираємо всі мітки біля вузлів, окрім першого.

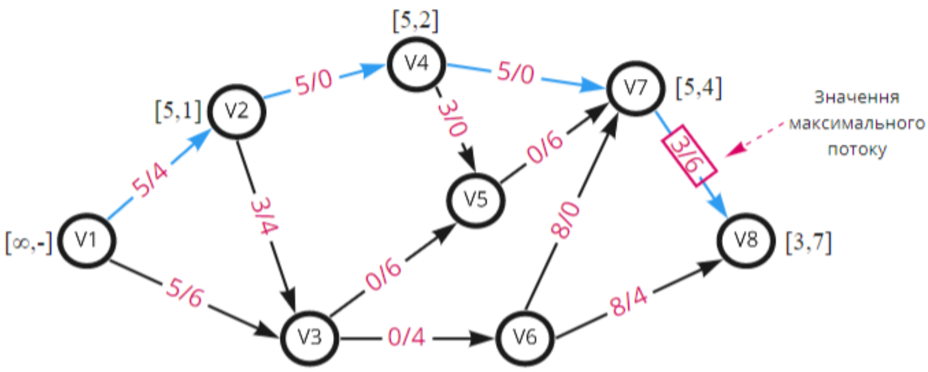
Перерахунок №2



Можемо переходити до наступної ітерації.

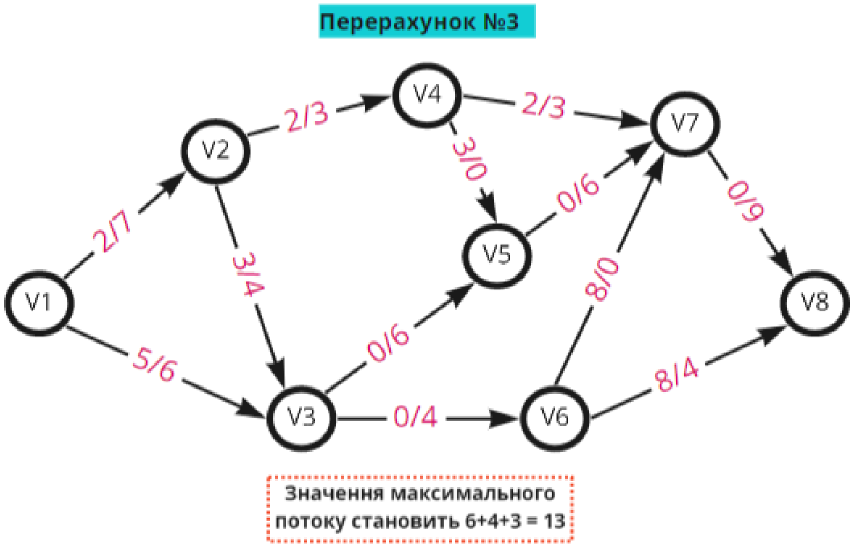
Ітерація 3

Зобразимо шлях на графі:



Значення максимального потоку для даного шляху – 3.

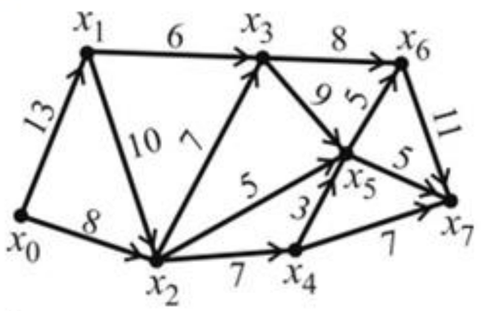
Віднімаємо дане значення від значень *с* задіяних ребер. Після цього прибираємо всі мітки біля вузлів, окрім першого.



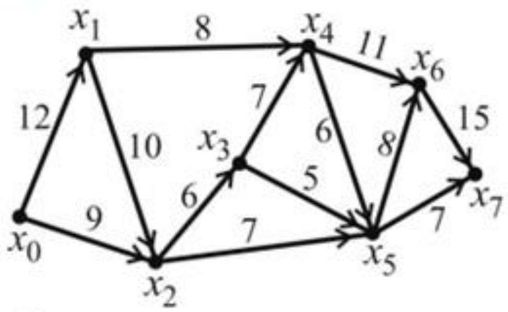
Оскільки з вузла v_1 більше неможливо потрапити в кінцевий вузол, то виконання ітерацій на цьому завершується. Значення максимального потоку дорівнює 13.

Варіанти

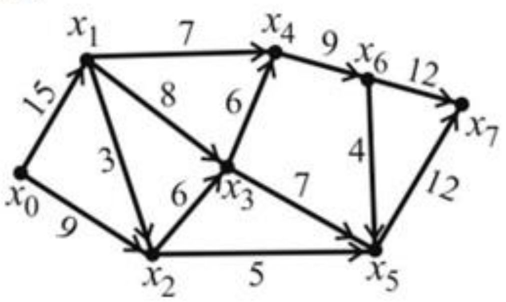
1.



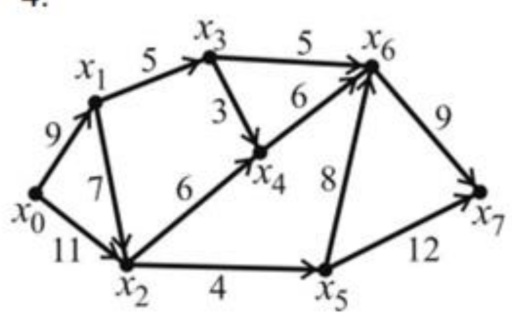
2.



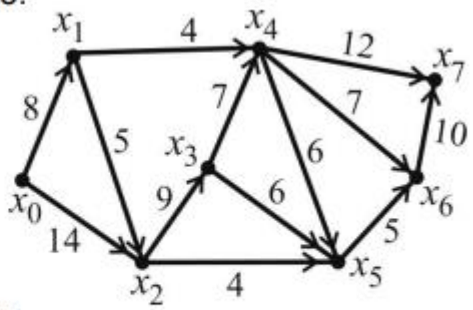
3.



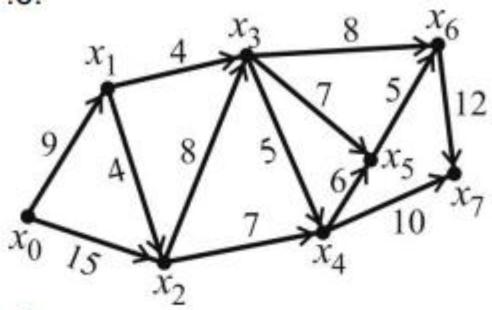
4.



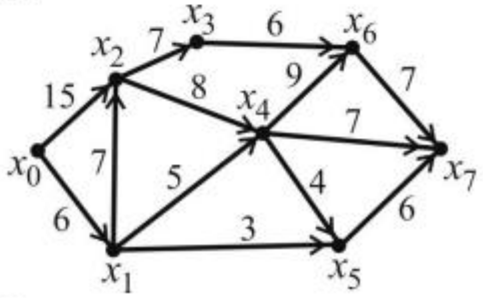
.5.



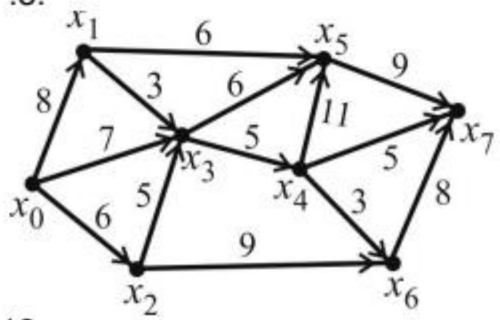
.6.



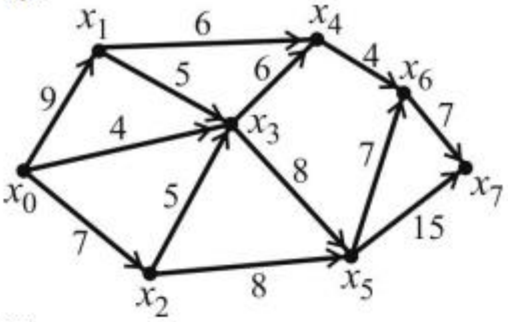
.7.



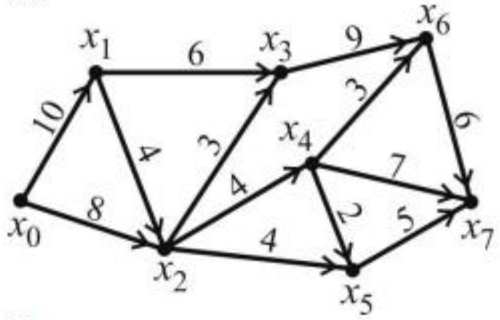
.8.



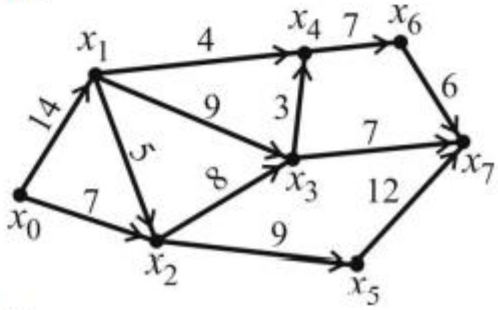
.9.



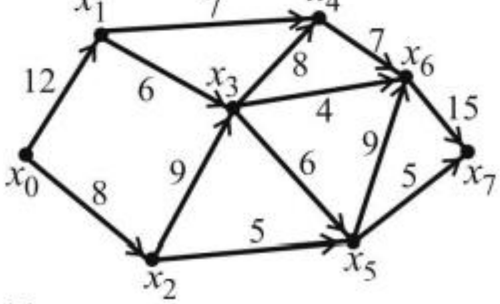
10.



11.



12.



Контрольні запитання

1. Що називають транспортною мережею?
2. Які умови (згідно з означенням) мають задовольняти дуговий потік, насичені (ненасичені, навантажені, ненавантажені) дуги на ТМ?
3. Що називають потоком і повним потоком на ТМ та у якому разі кажуть, що здійснено їхній розподіл за дугами?
4. Що розуміють під індексацією (зважуванням) вершин ТМ і як її виконують?

5. Який висновок слід зробити в разі, коли вихід ТМ виявився непроіндексованим?
6. Як здійснюють зміну потоку (у бік його збільшення) на ТМ зі зваженими вершинами?