МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ І ПРИКЛАДНИХ ТЕХНОЛОГІЙ

Р. М. Бабаков

ДИСКРЕТНА МАТЕМАТИКА ДЛЯ КОМП'ЮТЕРНИХ НАУК

Методичні рекомендації до виконання індивідуальних робіт
для здобувачів вищої освіти ОС «Бакалавр» денної форми здобуття освіти
спеціальності F3 Комп'ютерні науки

Вінниця
2025

УДК 519.1:004](072)
Б 12

Рекомендовано до друку вченою радою
факультету інформаційних і прикладних технологій
Донецького національного університету імені Василя Стуса
(протокол № 4 від 24 жовтня 2025 р.)

Автор:

Бабаков Р. М., д-р техн. наук, доцент, професор кафедри інформаційних технологій Донецького національного університету імені Василя Стуса.

Рецензенти:

Веселовська Н. Р., д-р техн. наук, в. о. завідувача кафедри інформаційних технологій Донецького національного університету імені Василя Стуса;

Фриз І. В., канд. фіз.-мат. наук, старший викладач кафедри інформаційних технологій Донецького національного університету імені Василя Стуса.

Р. М. Бабаков

Б 12 Дискретна математика для комп'ютерних наук: методичні рекомендації до виконання індивідуальних робіт для здобувачів вищої освіти ОС «Бакалавр» денної форми здобуття освіти спеціальності F3 Комп'ютерні науки. Вінниця: ДонНУ імені Василя Стуса, 2025. 63 с.

Методичні рекомендації є навчально-методичним документом, який містить рекомендації для отримання практичних навичок розв'язання задач під час виконання практикуму з дисципліни «Дискретна математика для комп'ютерних наук».

Для здобувачів освіти ОС «Бакалавр» спеціальності F3 Комп'ютерні науки факультету інформаційних і прикладних технологій ДонНУ імені Василя Стуса.

УДК 519.1:004](072)

© Бабаков Р. М., 2025
© ДонНУ імені Василя Стуса, 2025

ЗМІСТ

ВСТУП.....4
Індивідуальна робота № 1. Алгебра множин.....6
Приклад виконання індивідуальної роботи № 1.....9
Індивідуальна робота № 2. Системи числення і комп'ютерна арифметика.....11
Приклад виконання індивідуальної роботи № 2.....13
Індивідуальна робота № 3. Загальний аналіз графу.....18
Приклад виконання індивідуальної роботи № 3.....20
Індивідуальна робота № 4. Пошук найкоротших шляхів на графі.....31
Приклад виконання індивідуальної роботи № 4.....33
Індивідуальна робота № 5. Остови й цикли у графах.....36
Приклад виконання індивідуальної роботи № 5.....38
Індивідуальна робота № 6. Потоки в мережах.....51
Приклад виконання індивідуальної роботи № 6.....52
Індивідуальне творче завдання.....59
СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ.....62

ВСТУП

Дискретна математика лежить в основі будь-якого сучасного вивчення інформатики. Слово «дискретний» означає «такий, що складається з окремих частин», а дискретна математика стосується сукупностей об'єктів, що називаються множинами, і визначеними на них структурами. Більшість таких множин кінцеві і можуть ототожнюватися з реальними об'єктами навколишнього світу. Сучасний цифровий комп'ютер – по суті, кінцева дискретна система. Розуміння того, як працює комп'ютер, можна досягти, якщо розглядати його як дискретну математичну систему.

В Україні підготовка фахівців з комп'ютерних наук ступеня освіти «бакалавр» здійснюється за спеціальністю F3 Комп'ютерні науки в межах галузі знань F Інформаційні технології відповідно до затвердженого стандарту вищої освіти. Серед освітніх компонент професійного спрямування, що формують загальні й фахові компетентності майбутнього фахівця з комп'ютерних наук, можна зазначити такі навчальні дисципліни: основи програмування, алгоритми і структури даних, теорія ймовірностей та математична статистика в програмних середовищах, комп'ютерні мережі та основи вебтехнологій, системний аналіз та моделювання систем, комп'ютерно-математичне моделювання, а також комплекс дисциплін, пов'язаних із обробкою, зберіганням та аналізом даних.

Запорукою успішного опанування цими дисциплінами є базова підготовка, роль якої значною мірою належить дисципліні «Дискретна математика для комп'ютерних наук». Ця дисципліна традиційно представляє сукупність розділів, що присвячені дискретним математичним структурам (зазвичай скінченим), які у широкому сенсі можуть охоплювати значні частини алгебри, теорії чисел, математичної логіки. Серед багатьох розділів дискретної математики, які мають прикладне застосування у подальшій підготовці та професійній діяльності фахівця з комп'ютерних наук, найважливішими є такі: «Теорія множин», «Системи числення та комп'ютерна арифметика», «Теорія графів».

Розділ «Теорія множин» має передувати усім іншим розділам дискретної математики, оскільки дає понятійну і термінологічну базу у контексті математичного апарату теорії множин.

У розділі «Системи числення та комп'ютерна арифметика» здобувачі вищої освіти ознайомлюються з двійковою системою числення, опановують формати представлення числових та інших даних у пам'яті комп'ютера, встановлюють аналогії між алгоритмами десяткової та двійкової обробки даних. Прикладний аспект цього розділу полягає у його безпосередньому інтегруванні у процеси, що пов'язані з розробкою алгоритмів і написанням комп'ютерних програм, які використовують обробку даних на рівні окремих двійкових розрядів та бітових структур даних.

Розділ «Теорія графів» переважно присвячений вивченню методів і алгоритмів на графах, до яких належать методи пошуку найкоротших шляхів на графі, побудови мінімальних остовних дерев, пошуку ейлерових та гамільтонових циклів, а також пошуку максимального потоку в мережі. Ці теми знаходять практичне застосування у різних сферах людського життя і орієнтовані на розуміння відповідних алгоритмічних підходів з метою їх майбутньої програмної реалізації.

У Донецькому національному університеті імені Василя Стуса під час підготовки здобувачів освіти за спеціальністю F3 Комп'ютерні науки зазначені розділи розглядаються в межах навчальної дисципліни «Дискретна математика для комп'ютерних наук». Методичні рекомендації містять детальні пояснення щодо виконання індивідуальних робіт та творчого завдання за цією дисципліною, забезпечують отримання програмних результатів навчання, загальних і спеціальних компетентностей для відповідної освітньої компоненти.

Індивідуальна робота № 1
Алгебра множин

Вибір варіанта завдання

У цій індивідуальній роботі варіантом завдання є деяка предметна область, яку здобувач вибирає з таблиці 1 відповідно до свого номера в журналі групи.

Оформлення звіту з індивідуальної роботи

Звіт з індивідуальної роботи оформлюється від руки або набирається на комп'ютері. У будь-якому разі мають використовуватися аркуші формату А4, заповнені з однієї сторони.

Звіт повинен містити такі обов'язкові елементи:

– титульний аркуш, оформлений за вимогами Університету;
– варіант завдання (рядок з таблиці 1, відповідно до номера варіанта);
– результати роботи (за пунктами завдання);
– висновки до індивідуальної роботи (2–3 рядки у довільній формі).

Завдання до індивідуальної роботи

Завдання складається з декількох пунктів, які треба виконувати послідовно один за іншим. Під час оформлення звіту процес виконання завдання описувати необов'язково – достатньо вказати номер пункту, навести початкові умови завдання та кінцевий результат.

1. Сформувати множину U, елементи якої належать заданій предметній області. Потужність множини U повинна бути не менше 20 і не більше 30. Задати множину U перерахуванням і у вигляді характеристичного предиката. Додатково зробити окремий список, у якому елементи множини U пронумерувати (у будь-якому порядку).

2. Вважаючи надалі множину U універсумом, задати на множині U чотири множини $A, B, C, D \subset U$. Кожна із множин має поєднувати свої елементи за деякою загальною ознакою, причому для різних множин ці ознаки ***повинні бути не зв'язаними між собою*** (див. нижче приклад виконання роботи). Потужність кожної із множин повинна бути не менше 7 і не більше 10. Задати кожну із множин A, B, C, D перерахуванням і за допомогою характеристичного предиката. Визначити потужність кожної із множин.

Також вміст множин A, B, C, D зручно розмістити на окремому аркуші, представивши елементи кожної множини у вигляді окремого стовпця, у якому до кожного елементу проставлений номер з раніше складеного списку. Такий підхід повинен допомогти здобувачу освіти під час виконання наступних завдань.

Перед виконанням наступних пунктів завдання рекомендується показати результат перших двох пунктів викладачу. Це дасть змогу викладачу відразу вказати можливі неточності, а здобувачу освіти – уникнути надалі перероблення всієї індивідуальної роботи.

3. Сформувати множини-доповнення для множин A, B, C, D.

4. Виконати такі операції:

$A\cup B$, $A\cup C$, $A\cup D$, $B\cup C$, $B\cup D$, $C\cup D$, $D\cup B$, $D\cup A$, $C\cup B$, $B\cup A$, $C\cup A$.

5. Виконати такі операції:

$A\cup B\cup C$, $D\cup A\cup C$, $B\cup C\cup D$, $A\cup B\cup C\cup D$.

6. Виконати такі операції:

$D\cap A$, $D\cap B$, $D\cap C$, $C\cap B$, $C\cap A$, $B\cap A$, $B\cap D$, $A\cap C$, $A\cap D$, $A\cap B$, $B\cap C$.

7. Виконати такі операції:

$C\cap B\cap A$, $D\cap A\cap B$, $A\cap C\cap D$, $A\cap B\cap D\cap C$.

8. Виконати такі операції з урахуванням пріоритетів:

$A\cap\overline{B}$, $B\cap\overline{C}$, $C\cap\overline{D}$, $\overline{A}\cap\overline{B}\cap\overline{C}$, $\overline{B}\cap\overline{C}\cap\overline{D}$, $\overline{A}\cap\overline{B}\cap\overline{C}\cap\overline{D}$.

9. Виконати такі операції з урахуванням пріоритетів:

$A\setminus\overline{B}$, $C\setminus\overline{D}$, $D\setminus\overline{A}$, $B\setminus A$, $\overline{D}\setminus C$.

10. Виконати такі операції:

$A\Delta B$, $B\Delta C$, $C\Delta D$, $A\Delta B\Delta C$, $B\Delta C\Delta D$, $A\Delta B\Delta C\Delta D$.

11. Виконати такі операції з урахуванням пріоритетів:

$A\cup B\cap C\cup D$, $A\cap B\cup C\cap D$, $\overline{A}\cup B\cap\overline{C}\cup D$, $A\cap\overline{B}\cup C\cap\overline{D}$.

12. Виконати такі операції з урахуванням пріоритетів:

$A\cap(B\cap C\cup D)\cup(A\cup B\cap C)\cap D$;

$((D\cap(B\cap C\cup D)\cup B)\cup A\cap C)\cap A$;

$(C\cup A\cap B)\cup(D\cup B\cup D\cap A)\cap C$.

13. Виконати такі операції з урахуванням пріоритетів:

$A\setminus B\cap D\setminus C\cup A\cap C\setminus D\cup B\setminus C\cap A$;

$D\setminus A\cup C\setminus B\cap A\cap D\setminus B\cap C\setminus D\cup A$.

14. Виконати такі операції з урахуванням пріоритетів:

$A\setminus B\cap(D\setminus C\cup A\cap C)\setminus D\cup(B\setminus C)\cap A$;

$(A\setminus B\cap D\setminus C)\Delta A\cap(C\setminus D\cup B)\setminus C\cap A$.

15. Виконати такі операції з урахуванням пріоритетів:

$\overline{A\setminus B\cap D}\setminus C\cup\overline{A\cap\overline{C\setminus D\cup B}\setminus C}\cap A$;

$\overline{D\setminus A}\cup\overline{C\setminus B\Delta A}\cap D\setminus\overline{B\cap\overline{C\setminus D\cup A}}$;

$\overline{\overline{A \cup B} \cup C \cup D \cap A \cap B \cap C \cap D}$.

16. Виконати таку операцію з урахуванням пріоритетів:

$\overline{\overline{A \setminus B} \cap \overline{C} \cup (D \Delta B \cup A)} \cap \overline{C \setminus (D \setminus B \cap \overline{A} \cap C \Delta B \cap D)} \setminus A \cap C$.

17. На прикладах вмісту множин A, B, C, D перевірити експериментально справедливість таких тотожностей:

$A \cup (B \cup C) = (A \cup B) \cup C$;

$A \cap (B \cap C) = (A \cap B) \cap C$;

$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$;

$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$.

18. На прикладах вмісту множин A, B, C, D перевірити експериментально справедливість таких тотожностей:

$A \cup (\overline{A} \cap B) = A \cup B$;

$A \cap (\overline{A} \cup B) = A \cap B$;

$\overline{A \cup B} = \overline{A} \cap \overline{B}$;

$\overline{A \cap B} = \overline{A} \cup \overline{B}$;

$(A \cap B) \cup (\overline{A} \cap B) = B$;

$(A \cup B) \cap (\overline{A} \cup B) = B$.

19. Перевірити, чи утворюють множини A, B, C і D покриття або розбивку множини U.

20. Виконати завдання 19 для множин $\overline{A}, \overline{B}, \overline{C}, \overline{D}$.

Таблиця 1

№	Предметна область	№	Предметна область
1	Транспортні засоби	16	Літературні твори
2	Тварини	17	Види спорту
3	Музичні інструменти	18	Художні фільми
4	Комп’ютерні ігри	19	Риби
5	Країни	20	Канцелярські товари
6	Напої	21	Магазини Вінниці
7	Інтернет-сайти	22	Комп’ютерні товари
8	Професії	23	Дорогоцінне каміння
9	Взуття	24	Косметичні приналежності
10	Зброя	25	Рослини
11	Суспільні заклади	26	Музичні виконавці
12	Одяг	27	Городні рослини
13	Птахи	28	Деталі комп’ютера
14	Комп’ютерні програми	29	Породи собак

15	Кулінарні страви	30	Телепередачі

Приклад виконання індивідуальної роботи № 1

Розглянемо приклад виконання ***перших двох пунктів*** індивідуальної роботи. Задамо таку предметну область: «Робочий інструмент».

1. Сформуємо множину-універсум U, яку задамо перерахуванням його елементів:

U = {молоток, лопата, викрутка, плоскогубці, сапка, граблі, сокира, секатор, напилок, ніж, гайковий ключ, пензлик, шпатель, дриль, свердел, перфоратор, рівень, рубанок, пилка, цебро, ножівка, кліщі, зубило, паяльник, домкрат, валик, шліфмашина, ліхтарик, стамеска, рулетка}.

Задати множину U у вигляді характеристичного предиката можна в такий спосіб:

$U = \{u \in U \mid u$ – робочий інструмент$\}$.

Пронумеруємо елементи множини так, як показано в таблиці.

№	Елемент множини	№	Елемент множини
1	молоток	16	перфоратор
2	лопата	17	рівень
3	викрутка	18	рубанок
4	плоскогубці	19	пила
5	сапка	20	цебро
6	граблі	21	ножівка
7	сокира	22	кліщі
8	секатор	23	зубило
9	напилок	24	паяльник
10	ніж	25	домкрат
11	гайковий ключ	26	валик
12	пензлик	27	шліфмашина
13	шпатель	28	ліхтарик
14	дриль	29	стамеска
15	свердел	30	рулетка

2. Сформуємо на множині U чотири множини. Ознаки, за якими елементи будуть поєднуватися в множини, виберемо в такий спосіб, щоб одна ознака не залежала від іншої. Кожну множину задамо у вигляді перерахування та у вигляді характеристичного предиката.

1) Множина інструментів, використовуваних маляром:

A = {плоскогубці, ніж, пензлик, шпатель, рівень, цебро, кліщі, валик, стамеска, рулетка}.

$A = \{a \in A \mid a$ – використовується маляром$\}$.

$|A| = 10$.

2) Множина інструментів, застосовуваних під час ремонту автомобіля:

B = {молоток, викрутка, плоскогубці, ніж, гайковий ключ, цебро, кліщі, паяльник, домкрат, ліхтарик}.

$B = \{b \in B \mid b$ – використовується для ремонту автомобіля$\}$.

$|B| = 10$.

3) Множина інструментів, що вміщаються в кишеню:

C = {викрутка, плоскогубці, ніж, гайковий ключ, пензлик, свердел, зубило, стамеска, ліхтарик, рулетка}.

$C = \{c \in C \mid c$ – вміщається в кишеню$\}$.

$|C| = 10$.

4) Множина інструментів, використовуваних на садовій ділянці:

D = {молоток, лопата, сапка, граблі, сокира, секатор, ніж, пилка, цебро, ножівка}.

$D = \{d \in D \mid d$ – використовується на садовій ділянці$\}$.

$|D| = 10$.

Для зручності використання представимо вміст множин у вигляді стовпців.

A	***B***	***C***	***D***
4 – плоскогубці	1 – молоток	3 – викрутка	1 – молоток
10 – ніж	3 – викрутка	4 – плоскогубці	2 – лопата
12 – пензлик	4 – плоскогубці	10 – ніж	5 – сапка
13 – шпатель	10 – ніж	11 – гайковий ключ	6 – граблі
17 – рівень	11 – гайковий ключ	12 – пензлик	7 – сокира
20 – цебро	20 – цебро	15 – свердел	8 – секатор
22 – кліщі	22 – кліщі	23 – зубило	10 – ніж
26 – валик	24 – паяльник	28 – ліхтарик	19 – пила
29 – стамеска	25 – домкрат	29 – стамеска	20 – цебро
30 – рулетка	28 – ліхтарик	30 – рулетка	21 – ножівка

Інші пункти індивідуальної роботи виконуються за аналогією з прикладами, розглянутими на лекційних та практичних заняттях.

Індивідуальна робота № 2
Системи числення і комп'ютерна арифметика

Варіантом завдання для виконання індивідуальної роботи є прізвище, ім'я та по батькові здобувача освіти, записані українською мовою у називному відмінку. Замінимо у прізвищі, імені та по батькові приголосні букви одиницями, голосні – нулями. Внаслідок цього одержимо три числа A, B і C, записані у двійковому коді. Наприклад:

Іванов	$A_2 = 010101_2$;
Володимир	$B_2 = 101010101_2$;
Дмитрович	$C_2 = 110110101_2$.

Розмістивши біти в числах A, B і C у зворотному порядку, одержимо ще три числа D, E і F, записані у двійковому коді:

$D_2 = 101010_2$;
$E_2 = 101010101_2$;
$F_2 = 101011011_2$.

Отримані шість чисел є вхідними даними для виконання завдань в індивідуальній роботі. Під час роботи треба виконати таке:

1. Перевести числа A_2, B_2, C_2, D_2, E_2, F_2 у десяткову систему числення, одержавши числа A_{10}, B_{10}, C_{10}, D_{10}, E_{10}, F_{10}. Відобразити процес перекладу у звіті з індивідуальної роботи (не на чернетці).
2. Перевести числа A_{10}–F_{10} у трійкову систему числення, одержавши числа A_3–F_3. Відобразити процес перекладу у звіті з індивідуальної роботи.
3. Перевести числа A_{10}–F_{10} у систему числення з основою 7, одержавши числа A_7–F_7. Відобразити процес перекладу у звіті з індивідуальної роботи.
4. Перевести числа A_{10}–F_{10} у вісімкову систему числення, одержавши числа A_8–F_8. Відобразити процес перекладу у звіті з індивідуальної роботи.
5. Перевести числа A_{10}–F_{10} у шістнадцяткову систему числення, одержавши числа A_{16}–F_{16}. Відобразити процес перекладу у звіті з індивідуальної роботи.
6. Знайти $A_2 + F_2$, $C_2 + D_2$, $E_2 + B_2$, $A_2 + E_2$, $D_2 + F_2$.
7. Знайти $A_3 + F_3$, $C_3 + D_3$, $E_3 + B_3$, $A_3 + E_3$, $D_3 + F_3$.
8. Знайти $A_7 + F_7$, $C_7 + D_7$, $E_7 + B_7$, $A_7 + E_7$, $D_7 + F_7$.
9. Знайти $A_8 + F_8$, $C_8 + D_8$, $E_8 + B_8$, $A_8 + E_8$, $D_8 + F_8$.
10. Знайти $A_{16} + F_{16}$, $C_{16} + D_{16}$, $E_{16} + B_{16}$, $A_{16} + E_{16}$, $D_{16} + F_{16}$.
11. Знайти $A_2 – F_2$, $C_2 – D_2$, $E_2 – B_2$, $A_2 – E_2$, $D_2 – F_2$.
12. Знайти $A_3 – F_3$, $C_3 – D_3$, $E_3 – B_3$, $A_3 – E_3$, $D_3 – F_3$.
13. Знайти $A_7 – F_7$, $C_7 – D_7$, $E_7 – B_7$, $A_7 – E_7$, $D_7 – F_7$.
14. Знайти $A_8 – F_8$, $C_8 – D_8$, $E_8 – B_8$, $A_8 – E_8$, $D_8 – F_8$.
15. Знайти $A_{16} – F_{16}$, $C_{16} – D_{16}$, $E_{16} – B_{16}$, $A_{16} – E_{16}$, $D_{16} – F_{16}$.

16. Вважаючи числа A_2–F_2 знаковими числами у прямому коді, знайти їхні десяткові значення $A_{ПК}$, ..., $F_{ПК}$.

17. Вважаючи числа A_2–F_2 знаковими числами в прямому коді, представити їх у 16-розрядному форматі, одержавши числа $A_{ПК16}$, ..., $F_{ПК16}$.

18. Над числами $A_{ПК16}$–$F_{ПК16}$ виконати такі дії:

$A_{ПК16} + B_{ПК16}$, $C_{ПК16} + F_{ПК16}$, $B_{ПК16} + E_{ПК16}$, $D_{ПК16} + A_{ПК16}$, $F_{ПК16} + E_{ПК16}$.

Результат кожної операції перевести в шістнадцяткову й десяткову системи числення.

19. Вважаючи числа A_2–F_2 знаковими числами в додатковому коді, знайти їх десяткові значення $A_{ДК}$, ..., $F_{ДК}$.

20. Вважаючи числа A_2–F_2 знаковими числами в додатковому коді, представити їх у 16-розрядному форматі, одержавши числа $A_{ДК16}$, ..., $F_{ДК16}$.

21. Над числами $A_{ДК16}$–$F_{ДК16}$ виконати такі дії:

$A_{ДК16} + B_{ДК16}$, $C_{ДК16} + F_{ДК16}$, $B_{ДК16} + E_{ДК16}$, $D_{ДК16} + A_{ДК16}$, $F_{ДК16} + E_{ДК16}$.

Результат кожної операції перевести в шістнадцяткову й десяткову системи числення.

22. Для чисел $A_{ПК}$–$F_{ПК}$ знайти протилежні за знаком числа $-A_{ПК}$, ..., $-F_{ПК}$ і записати їх у двійковій системі числення у 16-розрядному форматі.

23. Для чисел $A_{ДК}$–$F_{ДК}$ знайти протилежні за знаком числа $-A_{ДК}$, ..., $-F_{ДК}$ і записати їх у двійковій системі числення у 16-розрядному форматі.

24. Для чисел, отриманих у п. 22, розглянути окремо старший і молодший байти. Вважаючи кожен байт окремим числом, представленим у прямому коді, одержати його десяткове значення.

25. Для чисел, отриманих у п. 23, розглянути окремо старший і молодший байти. Вважаючи кожен байт окремим числом, представленим у додатковому коді, одержати його десяткове значення.

Оформлення звіту з індивідуальної роботи

Звіт з індивідуальної роботи оформлюється від руки або набирається на комп’ютері. В будь-якому разі повинні використовуватися аркуші формату А4, заповнені з однієї сторони.

Звіт повинен містити такі обов’язкові елементи:

– титульний аркуш, оформлений за вимогами Університету;

– варіант завдання (прізвище, ім’я та по батькові здобувача освіти українською мовою в називному відмінку);

– результати роботи (за пунктами завдання);

– висновки до індивідуальної роботи (2–3 рядки у довільній формі).

Приклад виконання індивідуальної роботи № 2

Розглянемо приклад виконання ***деяких*** пунктів індивідуальної роботи на прикладі двох чисел А і В. Сформуємо числа в такий спосіб:

Всеволод $A_2 = 11010101_2$;

Резніков $B_2 = 10110101_2$.

Далі йдемо за пунктами завдання.

1. Переведемо A_2 і B_2 у десяткову СЧ.

$A_{10} = 2^7 + 2^6 + 2^4 + 2^2 + 2^0 = 128 + 64 + 16 + 4 + 1 = 213_{10}$.

$B_{10} = 2^7 + 2^5 + 2^4 + 2^2 + 2^0 = 128 + 32 + 16 + 4 + 1 = 181_{10}$.

2. Переведемо числа A_{10}, B_{10} у трійкову СЧ.

У процесі перекладу будемо використовувати послідовне заповнення розрядної сітки, яка у випадку трійкової СЧ буде мати такий вигляд:

4	3	2	1	0
$3^4=81$	$3^3=27$	$3^2=9$	$3^1=3$	$3^0=1$

Тоді A_{10} буде презентовано у трійковій СЧ в такий спосіб:

$A_3 = 2*3^4 + 1*3^3 + 2*3^2 + 2*3^1 + 0*3^0 = 162 + 27 + 18 + 6 + 0 = 21220_3$.

$B_3 = 2*3^4 + 0*3^3 + 2*3^2 + 0*3^1 + 1*3^0 = 162 + 0 + 18 + 0 + 1 = 20201_3$.

6. Виконаємо $A_2 + B_2$.

1	1	1	1		1		1		переноси
+	1	1	0	1	0	1	0	1	A_2
	1	0	1	1	0	1	0	1	B_2
1	1	0	0	0	1	0	1	0	$A_2 + B_2$

7. Виконаємо $A_3 + B_3$.

1		1				переноси
+	2	1	2	2	0	A_3
	2	0	2	0	1	B_3
1	1	2	1	2	1	$A_3 + B_3$

12. Виконаємо $A_2 – B_2$.

		1							позики
–	1	1	0	1	0	1	0	1	A_2
	1	0	1	1	0	1	0	1	B_2
	0	0	1	0	0	0	0	0	$A_2 – B_2$

13. Виконаємо $A_3 – B_3$.

				1		позики
	2	1	2	2	0	A_3
–	2	0	2	0	1	B_3
	0	1	0	1	2	$A_3 – B_3$

16. Вважаючи числа A_2 і B_2 знаковими числами у прямому коді, знайдемо їх десяткові значення $A_{ПК}$ і $B_{ПК}$.

Старший розряд числа A_2 дорівнює 1, отже, число від'ємне. Розряди, що залишилися, утворюють модуль числа:

$|A| = 1010101_2 = 2^6 + 2^4 + 2^2 + 2^0 = 64 + 16 + 4 + 1 = 85_{10}$.

Отже, $A_{ПК} = 1.1010101_2 = –85_{10}$.

Старший розряд числа B_2 дорівнює 1, отже, число від'ємне. Розряди, що залишилися, утворюють модуль числа:

$|B| = 0110101_2 = 2^5 + 2^4 + 2^2 + 2^0 = 32 + 16 + 4 + 1 = 53_{10}$.

Отже, $B_{ПК} = 1.0110101_2 = –53_{10}$.

17. Вважаючи числа A_2 і B_2 знаковими числами в прямому коді, представити їх у 16-розрядному форматі, одержавши в результаті числа $A_{ПК16}$ та $B_{ПК16}$.

Для того, щоб збільшити розрядність числа, представленого в ПК, зберігши водночас правильний знак, потрібно між знаковим розрядом і модулем числа додати відсутню кількість розрядів, рівних нулю.

У нашому випадку число $A_{ПК}$ має 8 розрядів. Отже, щоб збільшити його довжину до 16 розрядів, потрібно між знаком і модулем додати вісім нульових розрядів. От що ми одержимо:

$A_{ПК16} = 1.00000000\ 1010101_{ПК}$.

Аналогічно діємо із числом $B_{ПК}$. Оскільки в нашому прикладі воно також є восьмирозрядним, то після знака додаємо 8 нульових розрядів:

$B_{ПК16} = 1.00000000\ 0110101_{ПК}$.

18. Виконаємо $A_{ПК16} + B_{ПК16}$, представивши результат у двійковій, шістнадцятковій і десятковій СЧ.

Насамперед під час додавання чисел у ПК перевіряємо, чи рівні між собою знаки доданків. Очевидно, що знаки рівні, отже, складаємо модулі чисел.

								1	1	1	0	1		1		переноси
+	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	$\|A_{ПК16}\|$
	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1	$\|B_{ПК16}\|$
	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	$\|A_{ПК16}\| + \|B_{ПК16}\|$

Підставивши до результату знак, рівний у нашому випадку одиниці, одержимо таке значення:

$A_{ПК16} + B_{ПК16} = 1.000000010001010_2 = – (2^7 + 2^3 + 2^1) = – (128 + 8 + 1) = –137_{10}$.

$A_{ПК16} + B_{ПК16} = 1.000000010001010_2 = 1000\ 0000\ 1000\ 1010 = 808A_{16}$.

19. Вважаючи числа A_2 і B_2 знаковими числами в додатковому коді, знайдемо їх десяткові значення $A_{ДК}$ і $B_{ДК}$.

Старший розряд числа A_2=11010101 рівний 1, отже, число від'ємне і вимагає перекладу у прямий код. Забираємо знак, розряди, що залишилися, інвертуємо й додаємо одиницю, після чого підставляємо знак на місце. Одержимо:

$A_{ДК} = 1.1010101_{ДК} = 1.0101011_{ПК} = – (32 + 8 + 2 + 1) = –43_{10}$.

Аналогічно й із числом B:

$B_{ДК} = 1.0110101_{ДК} = 1.1001011_{ДК} = – (64 + 8 + 2 + 1) = –75_{10}$.

20. Представимо числа A_2 і B_2 в 16-розрядному форматі, вважаючи їх записаними в додатковому коді.

Для того, щоб збільшити розрядність двійкового числа в ДК, не змінивши водночас його значення, потрібно між знаковим і значущими розрядами додати відсутню кількість розрядів, ***значення яких дорівнює знаку числа.***

У нашому випадку $A_{ДК} = 1.1010101_{ДК}$ є восьмирозрядним від'ємним числом. Оскільки знак дорівнює 1, після знакового розряду вставляємо 8 розрядів, рівних одиниці. Одержуємо:

$A_{ДК16} = 1.11111111\ 1010101_{ДК}$.

Те ж саме проводимо з числом B:

$B_{ДК16} = 1.11111111\ 0110101_{ДК}$.

21. Виконаємо $A_{ДК16} + B_{ДК16}$, представивши результат у двійковій, шістнадцятковій і десятковій СЧ.

Під час додавання чисел у ДК попередній аналіз знаків не робимо, а складаємо числа «цілком», відразу одержуючи правильний результат:

1	1	1	1	1	1	1	1	1	1	1	1		1		1		переноси
	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	$A_{ДК16}$
+	1	1	1	1	1	1	1	1	1	0	1	1	0	1	0	1	$B_{ДК16}$
	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1	0	$A_{ДК16} + B_{ДК16}$

Зазначимо, що під час додавання чисел у ДК перенос зі старшого (знакового) розряду ігнорується й відкидається. Отже, результат становить стільки ж розрядів, скільки й числа, що додаються (у нашому випадку 16). Одержуємо:

$A_{ДК16} + B_{ДК16} = 1.1111111110001010_2 = 1111\ 1111\ 1000\ 1010 = FF8A_{16}$.

Щоб перевести результат із ДК у десяткову систему, треба спочатку перевести його у прямий код. Оскільки результат від'ємний (старший розряд дорівнює 1), то відокремлюємо знак; 15 розрядів, що залишилися, інвертуємо й додаємо одиницю, після чого підставляємо знак. Одержуємо:

$A_{ДК16} + B_{ДК16} = 1.1111111110001010_{ДК} = 1.0000000001110110_{ПК} = –(64 + 32 + 16 + 4 + 2) = –118_{10}$.

Якщо глянути на результат, отриманий у п. 19, то можна переконатись у правильності виконаного додавання. Дійсно:

$A_{ДК} + B_{ДК} = –43_{10} + (–75_{10}) = –118_{10}$.

22. Знайдемо для чисел $A_{ПК}$ і $B_{ПК}$ протилежні за знаком числа $–A_{ПК}$ і $–B_{ПК}$ та запишемо їх у двійковій системі числення в 16-розрядному форматі.

Для того, щоб у числа, представленого в прямому коді, змінити знак, достатньо інвертувати знаковий розряд числа:

$A_{ПК} = 1.1010101_2 = –85_{10}$.

$–A_{ПК} = 0.1010101_2 = +85_{10}$.

Оскільки $–A_{ПК}$ – додатне число, то під час розширення його до 16 розрядів після знакового розряду вставляємо вісім розрядів, рівних нулю:

$–A_{ПК16} = 0.00000000\ 1010101_2$.

Аналогічно для числа B:

$B_{ПК} = 1.0110101_2 = –53_{10}$.

$–B_{ПК} = 0.0110101_2 = +53_{10}$.

$–B_{ПК16} = 0.00000000\ 0110101_2$.

23. Знайдемо для чисел $A_{ДК}$ і $B_{ДК}$ протилежні за знаком числа $–A_{ДК}$ і $–B_{ДК}$ та запишемо їх у двійковій системі числення в 16-розрядному форматі.

Для того, щоб у числа, представленого в ДК, змінити знак, треба інвертувати всі розряди числа, після чого до отриманого значення додати одиницю. Ці операції робляться над усіма розрядами числа, включно зі знаковим. Виконаємо ці дії над числами A_2 і B_2, вважаючи їх числами в ДК:

$A_{ДК} = 1.1010101_2 = –43_{10}$.

$–A_{ДК} = 0.0101011_2$.

$–A_{ДК16} = 0.00000000\ 0101011_2$.

$B_{ДК} = 1.0110101_2 = –75_{10}$.

$–B_{ДК} = 0.1001011_2$.

$–B_{ДК16} = 0.0000000010010112$.

24. Для чисел, отриманих у п. 22, проаналізуємо окремо старший і молодший байти, вважаючи їх числами в прямому коді.

Отже, маємо:

$-A_{ПК16} = 0.00000000\ 1010101_2$.

Старший байт числа (старші 8 розрядів) дорівнює:

$00000000_2 = 0.0000000_{ПК} = 0_{10}$.

Молодший байт числа (молодші 8 розрядів) дорівнює:

$01010101_2 = 0.1010101_{ПК} = +(64 + 16 + 4 + 1) = +85_{10}$.

Аналогічно для числа B:

$-B_{ПК16} = 0.00000000\ 0110101_2$.

Старший байт числа (старші 8 розрядів) дорівнює:

$00000000_2 = 0.0000000_{ПК} = 0_{10}$.

Молодший байт числа (молодші 8 розрядів) дорівнює:

$01110101_2 = 0.0110101_{ПК} = +(32 + 16 + 4 + 1) = +53_{10}$.

25. Для чисел, отриманих у п. 23, проаналізуємо окремо старший і молодший байти, вважаючи їх числами в додатковому коді.

Отже, маємо:

$-A_{ДК16} = 0.00000000\ 0101011_2$.

Старший байт числа (старші 8 розрядів) дорівнює:

$00000000_2 = 0.0000000_{ДК} = 0_{10}$.

Молодший байт числа (молодші 8 розрядів) дорівнює:

$00101011_2 = 0.0101011_{ДК} = +(32 + 8 + 2 + 1) = +43_{10}$.

Аналогічно для числа B:

$-B_{ДК16} = 0.0000000010010112$.

Старший байт числа (старші 8 розрядів) дорівнює:

$00000000_2 = 0.0000000_{ДК} = 0_{10}$.

Молодший байт числа (молодші 8 розрядів) дорівнює:

$01001011_2 = 0.1001011_{ПК} = +(64 + 8 + 2 + 1) = +75_{10}$.

Індивідуальна робота № 3
Загальний аналіз графу

Індивідуальні роботи 3–5 присвячені розділу «Теорія графів». Особливістю цих робіт є те, що здобувачу освіти видається не готовий варіант завдання, а лише його тематика. Варіант завдання здобувач освіти розробляє сам, оформлює на окремому аркуші та підписує у викладача. Підписаний варіант завдання використовується для виконання індивідуальних робіт 3–5. Здобувач освіти повинен зберігати його у себе до кінця семестру і пред'являти викладачеві під час захисту кожної з індивідуальних робіт 3–5.

Варіантом завдання є орієнтований граф, відповідний до заданої тематики. Тематики завдання представлені в таблиці 1, де номер варіанта відповідає номеру здобувача освіти в журналі групи.

Здобувач вирішує самостійно, що буде розумітися під вершинами й ребрами графу, і які характеристики можуть мати ребра й вершини. У більшості випадків на момент одержання завдання у здобувача може не виявитися достатньо знань для побудови графу. У такому разі здобувач повинен попередньо вивчити задану область в об'ємі, достатньому для побудови графу.

Згідно з тематикою здобувач повинен побудувати орієнтований граф із заданими характеристиками:

1. Кількість вершин у графі: від 12 до 15.
2. Кількість дуг у графі: не менше 20.
3. Кількість ребер із кратністю більше 1: від 2 до 4.

Якщо побудований граф не відповідає цим характеристикам, необхідно внести в нього мінімально необхідну кількість змін для приведення його до необхідного вигляду.

Інші характеристики графу (наприклад, наявність петель, кінцевих і ізольованих вершин, ступені вершин тощо) вибираються відповідно до тематики варіанта завдання.

Оскільки граф відносно невеликий, буде зручно намалювати його на комп'ютері, наприклад, у програмі Word – це спростить оформлення звітів з індивідуальних робіт.

Таблиця 1

1	Порядок огляду визначних пам'яток Києва
2	Харчовий ланцюг у тваринному світі
3	Метрополітен м. Києва
4	Взаємозв'язок дисциплін, що вивчаються у середній школі
5	План евакуації з навчального корпусу ДонНУ імені Василя Стуса
6	Порядок обходу якої-небудь мережі магазинів м. Вінниці
7	Порядок огляду визначних пам'яток Криму

8	Взаємозв’язок деталей автомобіля та елементів керування
9	Укладання продовольчих товарів у сумку
10	Схема меню в ресторані (що після чого можна подавати)
11	Порядок обходу комп'ютерних фірм у м. Вінниці
12	Порядок проходження медогляду
13	Порядок надягання елементів гардеробу
14	Тролейбусні маршрути м. Вінниці
15	Порядок огляду визначних пам’яток Парижу
16	Розважальні заходи на святі
17	Порядок виконання вправ у тренажерному залі
18	Схема руху між містами України швидких поїздів, що мають денний час відправлення (з 9:00 до 18:00)
19	Переміщення здобувачів освіти між аудиторіями протягом тижня згідно з розкладом занять
20	Родичі
21	Схема обходу ТЦ «Мегамол»
22	Ваші знайомі (хто з ким знайомий або не знайомий)
23	Взаємний вплив лікарських препаратів
24	Трамвайні маршрути м. Вінниці
25	Внутрішні деталі комп’ютера

У цій індивідуальній роботі треба заповнити табл. 2, визначивши перераховані параметри графу і давши необхідні пояснення. Усі параметри повинні визначатися здобувачем «на око», без застосування будь-яких спеціальних методів і алгоритмів.

Таблиця 2

Під час заповнення таблиці треба дотримуватися таких рекомендацій:

1. Потрібно розширювати висоту кожного рядка для того, щоб вмістити необхідний матеріал. Це робиться шляхом додавання порожніх рядків, розтягування границь таблиці або налаштування параметрів потрібного рядка.

2. Таблицю дозволяється розміщувати на сторінці горизонтально, тобто використовувати альбомну орієнтацію сторінки.

3. Пункти 25–26 дозволяється навести окремо за межами таблиці, зазначивши в таблиці лише основні параметри таблиць і підграфів.

Оформлення звіту з індивідуальної роботи

Звіт з індивідуальної роботи оформлюється від руки або набирається на комп’ютері. Для звіту треба використовувати аркуші формату А4, заповнені з однієї сторони. Звіт повинен мати такі обов’язкові елементи:

– титульний аркуш, оформлений за вимогами Університету;

– граф, підписаний викладачем (копія);

– результати роботи (заповнена таблиця з характеристиками графу);

– висновки до індивідуальної роботи (2–3 рядки у довільній формі).

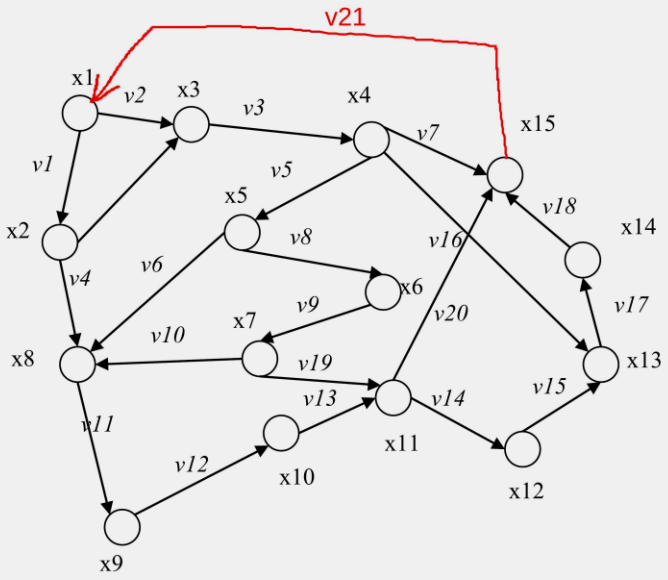
Приклад виконання індивідуальної роботи № 3

Нижче наведені приклад підписаного графу та приклад виконання індивідуальної роботи. Треба зазначити, що цей приклад виконання не є ідеальним, може містити помилки і не відповідати високій оцінці за роботу. Приклад демонструє лише загальний вигляд результатів роботи.

№	Назва параметра	Значення параметра	Пояснення
1	Множини вершин, ребер і дуг; потужності даних множин		
2	Вершина з максимальною кількістю суміжних вершин		
3	Вершини з максимальним і мінімальним ступенями		
4	Кількість ізольованих і кінцевих вершин		
5	Дуга (ребро) з максимальною кратністю		
6	Кількість петель		
7	Чи є граф простим графом, псевдографом або мультиграфом		
8	Три різні шляхи зі вказівкою їх довжин		
9	Три різні контури (якщо такі є) зі вказівкою їх довжин		
10	Три прості шляхи зі вказівкою їх довжин		
11	Три прості контури (якщо такі є) зі вказівкою їх довжин		
12	Три елементарні шляхи зі вказівкою їх довжин		
13	Три елементарні контури (якщо такі є) зі вказівкою їх довжин		
14	Гамільтонів шлях (якщо такий є) зі вказівкою його довжини		
15	Гамільтонів контур (якщо такий є) зі вказівкою його довжини		
16	Ейлерів шлях (якщо такий є) зі вказівкою його довжини		
17	Ейлерів контур (якщо такий є) зі вказівкою його довжини		
18	Чи є граф симетричним, чи антисиметричним		
19	Чи є граф зв’язним (якщо ні, то визначити кількість компонентів зв’язності)		
20	Зв’язність графу		
21	Чи є граф порожнім графом, чи повним графом		
22	Чи є граф однорідним		
23	Чи є граф регулярним, ступінь регулярності графу		
24	Чи є граф двочастковим		
25	Матриця суміжності		
26	Матриця інциденцій		
27	П’ять різних підграфів		

Панченко Д. М.
КН-23-В
Варіант №17
«Порядок виконання вправ у тренажерному залі»

12.02.2024



x1 – розминка
x2 – жим ногами
x3 – присідання зі штангою
x4 – розгинання ніг
x5 – підйоми на носочки стоячи
x6 – румунська тяга
x7 – зведення рук на нижньому блоці кросовера лежачи
x8 – жим штанги лежачи
x9 – жим сидячи
x10 – зведення рук
x11 – тяга верхнього блоку до грудей
x12 – тяга нижнього блоку до живота
x13 – підтягування прямим хватом
x14 – тяга штанги в нахилі
x15 – зворотні махи гантелями

Дуги графа відображають порядок виконання вправ у тренажерному залі підібраний для якоїсь людини. Наприклад, після розминки (x1) можна почати робити вправи на ноги починаючи з (дуга *v2*) присідання зі штангою (x3) потім (дуга *v3*) з розгинання ніг (x4), після йде (дуга *v5*) підйоми на носочки стоячи (x5), далі йдуть вправи на груди (дуга *v8*) румунська тяга (x6), також потім (дуга *v9*) зведення рук на нижньому блоці кросовера лежачи (x7), потім можна перейти на вправи для спини (дуга *v19*) тяга верхнього блоку до грудей (x11), потім (дуга *v20*) зворотні махи гантелями (x15) і на цьому кінець тренування.

Граф містить 15 вершин та 20 ребер. Кратні ребра на графі відсутні (у зв’язку з непотрібністю).

№	Назва параметра	Значення параметра	Пояснення
1	Множини вершин, ребер і дуг; потужність даних множин	X = {x1,x2,x3,x4,x5,x6,x7,x8, x9,x10,x11,x12,x13,x14,x15}	X – вершини графу.
		V = {v1=(x1,x2), v2=(x1,x3), v3=(x3,x4), v4=(x2,x8), v5=(x4,x5), v6=(x5,x8), v7=(x4,x15), v8=(x5,x6), v9=(x6,x7), v10=(x7,x8), v11=(x8,x9), v12=(x9,x10), v13=(x10,x11), v14=(x11,x12), v15=(x12,x13), v16=(x4,x13), v17=(x13,x14), v18=(x14,x15), v19=(x7,x11), v20=(x11,x15), v21=(x15,x1), v22=(x2, x3)}	V – дуги графу.
		X = {x∈X |x – вершини графу} |X| = 15	Кількість вершин графу = {x1-x15}, тому потужність множини вершин = 15.
		V = {v∈V |v – дуги графу} |V| = 22	Кількість дуг графу = {v1-v22}, тому потужність множини дуг = 22
2	Вершина з максимальною кількістю суміжних вершин	X4	Вершина x4 має максимальну кількість суміжних вершин. Адже вона суміжна з вершинами x5, x13, x15
3	Вершини з максимальним і мінімальними ступенями	d(x4) = 4, d(x8) = 4, d(x11) = 4, d(x15) = 4.	Вершина x4 має максимальний ступінь, тому що кількість інцедентних дуг до її вершини = 4. Максимальний вихідний ступінь = 3. Максимальний вхідний ступінь = 1.
		d(x6) = 2, d(x9) = 2, d(x10) = 2, d(x12) = 2, d(x14) = 2.	Вершини x8, x15 мають максимальний ступінь, тому що кількість інцедентних дуг до її вершини = 4. Максимальний вихідний ступінь = 1. Максимальний вхідний ступінь = 3. Вершина x11 має максимальний ступінь, тому що кількість інцедентних дуг до її вершини = 4. Максимальний вихідний ступінь = 2. Максимальний вхідний ступінь = 2. Вершини x6, x9, x10, x12, x14 мають мінімальний ступінь, тому що кількість інцедентних дуг до їх вершин = 2. Максимальний вихідний ступінь = 1. Максимальний вхідний ступінь = 1
4	Кількість ізольованих і кінцевих вершин	Кількість ізольованих – 0 Кількість кінцевих – 0	Ізольованих та кінцевих вершин у даному графі немає, тому що мінімальний ступінь вершин починаєть-

№	Назва параметра	Значення параметра	Пояснення
			ся з d(x6) = 2, d(x9) = 2, d(x10) = 2, d(x12) = 2, d(x14) = 2 та більше
5	Дуга (ребро) з максимальною кратністю	Відсутні	Дуга може бути кратною лише тоді, коли вона з'єднується з парою вершин двома або більше дугами одного напрямку. В даному графі вони відсутні. Максимальна кратність усіх дуг = 1
6	Кількість петель	Кількість петель – 0	Петель у даному графі не існує, тому що немає дуг, які співпадають з початковою та кінцевою вершиною
7	Чи є граф простим графом, псевдографом або мультиграфом	Граф є простим	Граф є простим, тому що має всі дуги кратністю 1. Граф не є псевдографом, тому що він не має петель. Граф не є мультиграфом, тому що він не має петель та не має дуг із кратністю більше 1
8	Три різні шляхи із вказівкою їх довжин	<x1,x3,x4,x5,x8,x9> <v2,v3,v5,v6,v11> Довжина шляху = 5	Шлях від вершини x1 до x9, що включає послідовність дуг <v2,v3,v5,v6,v11>. Довжина шляху дорівнює кількості дуг шляху, тобто 5.
		<x3,x4,x15,x1,x2,x8,x9> <v3,v7,v21,v1,v4,v11> Довжина шляху = 6	Шлях від вершини x3 до x9, що включає послідовність дуг <v3,v7,v21,v1,v4,v11>. Довжина шляху дорівнює кількості дуг шляху, тобто 6.
		<x2,x8,x9,x10,x11,x15,x1,x3,x4> <v4,v11,v12,v13,v20,v21,v2,v3> Довжина шляху = 8	Шлях від вершини x2 до x4, що включає послідовність дуг <v4,v11,v12,v13,v20,v21,v2,v3>. Довжина шляху дорівнює кількості дуг шляху, тобто 8
9	Три різні контури (якщо такі є) із вказівкою їх довжин	<x1,x3,x4,x5,x8,x9,x10,x11,x15,x1> <v2,v3,v5,v6,v11,v12,v13,v20,v21> Довжина контуру = 9	Контур від вершини x1 до x1, що включає послідовність дуг <v2,v3,v5,v6,v11,v12,v13,v20,v21>. Довжина контуру дорівнює кількості дуг контуру, тобто 9.
		<x3,x4,x5,x8,x9,x10,x11,x15,x1,x3> <v3,v5,v6,v11,v12,v13,v20,v21,v2> Довжина контуру = 9	Контур від вершини x3 до x3, що включає послідовність дуг <v3,v5,v6,v11,v12,v13,v20,v21,v2>. Довжина контуру дорівнює кількості дуг контуру, тобто 9.

№	Назва параметра	Значення параметра	Пояснення
		<x2,x8,x9,x10,x11,x15,x1,x2> <v4,v11,v12,v13,v20,v21,v1> Довжина контуру = 7	Контур від вершини x2 до x2, що включає послідовність дуг <v4,v11,v12,v13,v20,v21,v1>. Довжина контуру дорівнює кількості дуг контуру, тобто 7
10	Три прості шляхи із вказівкою їх довжин	<x1,x3,x4,x5,x8,x9> <v2,v3,v5,v6,v11> Довжина шляху = 5	Простий шлях від вершини x1 до x9, що включає послідовність дуг < v2,v3,v5,v6,v11>. Жодна дуга не повторюється. Довжина шляху дорівнює кількості дуг шляху, тобто 5.
		<x3,x4,x15,x1,x2,x8,x9> <v3,v7,v21,v1,v4,v11> Довжина шляху = 6	Простий шлях від вершини x3 до x9, що включає послідовність дуг <v3,v7,v21,v1,v4,v11>. Жодна дуга не повторюється. Довжина шляху дорівнює кількості дуг шляху, тобто 6.
		<x2,x8,x9,x10,x11,x15,x1,x3,x4> <v4,v11,v12,v13,v20,v21,v2,v3> Довжина шляху = 8	Простий шлях від вершини x2 до x4, що включає послідовність дуг <v4,v11,v12,v13,v20,v21,v2,v3>. Жодна дуга не повторюється. Довжина шляху дорівнює кількості дуг шляху, тобто 8
11	Три прості контури (якщо такі є) із вказівкою їх довжин	<x1,x3,x4,x5,x8,x9,x10,x11,x15,x1> <v2,v3,v5,v6,v11,v12,v13,v20,v21> Довжина контуру = 9	Простий контур від вершини x1 до x1, що включає послідовність дуг <v2,v3,v5,v6,v11,v12,v13,v20,v21>. Жодна дуга не повторюється. Довжина контуру дорівнює кількості дуг контуру, тобто 9.
		<x3,x4,x5,x8,x9,x10,x11,x15,x1,x3> <v3,v5,v6,v11,v12,v13,v20,v21,v2> Довжина контуру = 9	Простий контур від вершини x3 до x3, що включає послідовність дуг <v3,v5,v6,v11,v12,v13,v20,v21,v2>. Жодна дуга не повторюється. Довжина контуру дорівнює кількості дуг контуру, тобто 9.
		<x2,x8,x9,x10,x11,x15,x1,x2> <v4,v11,v12,v13,v20,v21,v1> Довжина контуру = 7	Простий контур від вершини x2 до x2, що включає послідовність дуг <v4,v11,v12,v13,v20,v21,v1>. Жодна дуга не повторюється. Довжина контуру дорівнює кількості дуг контуру, тобто 7
12	Три елементарні шляхи із вказівкою їх довжин	<x1,x3,x5,x8,x13,x14> <v2,v5,v20,v12,v19> Довжина шляху = 5	Елементарний шлях від вершини x1 до x9, що включає послідовність дуг < v2,v3,v5,v6,v11>. Жодна вершина не повторюється. Довжина шляху дорівнює кількості дуг шляху, тобто 5.

№	Назва параметра	Значення параметра	Пояснення
		<x3,x6,x12,x15,x11,x10,x14> <v4,v6,v16,v23,v17,v14> Довжина шляху = 6	Елементарний шлях від вершини x3 до x9, що включає послідовність дуг <v3,v7,v21,v1,v4,v11>. Жодна вершина не повторюється. Довжина шляху дорівнює кількості дуг шляху, тобто 6.
		<x2,x4,x7,x13,x14,x12,x1,x3,x5> <v1,v10,v11,v19,v24,v22,v2,v5> Довжина шляху = 8	Елементарний шлях від вершини x2 до x4, що включає послідовність дуг <v4,v11,v12,v13,v20,v21,v2,v3>. Жодна вершина не повторюється. Довжина шляху дорівнює кількості дуг шляху, тобто 8
13	Три елементарні контури (якщо такі є) із вказівкою їх довжин	<x1,x3,x4,x5,x8,x9,x10,x11,x15,x1> <v2,v3,v5,v6,v11,v12,v13,v20,v21> Довжина контуру = 9	Елементарний контур від вершини x1 до x1, що включає послідовність дуг <v2,v3,v5,v6,v11,v12,v13,v20,v21>. Жодна вершина не повторюється. Довжина контуру дорівнює кількості дуг контуру, тобто 9.
		<x3,x4,x5,x8,x9,x10,x11,x15,x1,x3> <v3,v5,v6,v11,v12,v13,v20,v21,v2> Довжина контуру = 9	Елементарний контур від вершини x3 до x3, що включає послідовність дуг <v3,v5,v6,v11,v12,v13,v20,v21,v2>. Жодна вершина не повторюється. Довжина контуру дорівнює кількості дуг контуру, тобто 9.
		<x2,x8,x9,x10,x11,x15,x1,x2> <v4,v11,v12,v13,v20,v21,v1> Довжина контуру = 7	Елементарний контур від вершини x2 до x2, що включає послідовність дуг <v4,v11,v12,v13,v20,v21,v1>. Жодна вершина не повторюється. Довжина контуру дорівнює кількості дуг контуру, тобто 7
14	Гамільтонів шлях (якщо такий є) із вказівкою його довжини	Гамільтонів шлях – присутній. <x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x15>	Гамільтонового шляху в даному графі є. Тому що гамільтонів шлях – це елементарний шлях (в якому не повинні повторюватися вершини двічі), який повинен проходити через усі вершини. У даному випадку це <x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x15>
15	Гамільтонів контур (якщо такий є) із вказівкою його довжини	Гамільтонів контур – присутній. <x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x15,x1>	Гамільтонового контуру в даному графі є. Тому що гамільтонів контур – це елементарний контур (у якому не повинні повторюватися вершини двічі), який повинен проходити через усі вершини. У даному випадку це можливо
16	Ейлерів шлях (якщо такий є) із вказівкою його довжини	Ейлерів шлях – відсутній	Ейлерового шляху в даному графі немає. Тому що ейлерів шлях – це простий шлях (в якому не повинні

№	Назва параметра	Значення параметра	Пояснення
			повторюватися дуги двічі), який повинен містити усі дуги. В даному випадку це неможливо
17	Ейлерів контур (якщо такий є) із вказівкою його довжини	Ейлерів контур – відсутній	Ейлерового контуру в даному графі немає. Тому що ейлерів контур – це простий контур (в якому не повинні повторюватися дуги двічі), який повинен містити усі дуги. У даному випадку це неможливо
18	Чи є граф симетричним або антисиметричним	Граф є антисиметричний	Граф – антисиметричний, тому що для будь-якої дуги не існує протилежно орієнтована дуга
19	Чи є граф зв'язним (якщо ні, то визначити кількість компонентів зв'язності)	Граф є зв'язним	Граф є зв'язним, тому що між будь-якою парою вершин цього графу існує шлях та він містить лише одну компонентну зв'язність. Також він не містить окремих компонентів зв'язності
20	Зв'язність графу	Зв'язність графу – 2	Зв'язність графу – це коли після мінімального видалення дуг граф стає незв'язним. Тому в даному графі, якщо видалити дуги v8, v9, вийде два незв'язні графи
21	Чи є граф порожнім графом або повним графом	Граф не є порожнім і не є повним	Граф не є порожнім, тому що всі вершини з'єднані дугами. Також мінімальний ступінь вершини починається з $d(x6) = 2$, $d(x9) = 2$, $d(x10) = 2$, $d(x12) = 2$, $d(x14) = 2$. Граф не є повним, тому що його ступінь всіх вершин не дорівнює n-1, де n – це кількість вершин
22	Чи є граф однорідним	Граф не є однорідним	Граф не є однорідним, тому що усі вершини графу мають різні ступені. Починаючи з $d(x6) = 2$, $d(x9) = 2$, $d(x10) = 2$, $d(x12) = 2$, $d(x14) = 2$ та закінчуючи $d(x4) = 4$, $d(x8) = 4$, $d(x11) = 4$, $d(x15) = 4$
23	Чи є граф регулярним, ступінь регулярності графу	Граф не є регулярним	Граф не є регулярним, тому що усі вершини мають різні ступені
24	Чи є граф двочастковим	Граф не є двочастковим	Граф не є двочастковим, тому що множина його вершин не може бути розбита на дві підмножини так, що кожне ребро графу буде мати одну вершину з першої підмножини і одну з другої та не перетинаючись між вершинами в своїй множині
25	Матриця суміжності	Таблиця наведена нижче	У таблиці зображено суміжність між двома вершинами. Якщо в таблиці стоїть 1, то ці дві вершини су-

№	Назва параметра	Значення параметра	Пояснення
			міжні. В іншому випадку, якщо в таблиці стоїть 0, то ці дві вершини між собою не суміжні
26	Матриця інциденцій	Таблиця наведена нижче	У таблиці зображено зв’язки між інцидентними елементами графу (дуга і вершина). Стовпці матриці відповідають дугам, рядки – вершинам. Якщо в таблиці стоїть 1, то ця дуга є вихідною з цієї вершини. Якщо в таблиці стоїть –1, то ця дуга є вхідною в цю вершину. В іншому випадку, якщо в таблиця стоїть 0, то ця дуга не має ніякого зв’язку з цією вершиною
27	П’ять різних підграфів	Малюнки наведені нижче	1) Перший підграф утворився після видалення вершини x6 та всіх дуг, які мали зв’язки з цією вершиною (v8, v9). 2) Другий підграф утворився після видалення вершини x1 та всіх дуг, які мали зв’язки з цією вершиною (v1, v2, v21). 3) Третій підграф утворився після видалення вершин x9, x10 та всіх дуг, які мали зв’язки з цими вершинами (v11, v12, v13). 4) Четвертий підграф утворився після видалення вершини x7 та всіх дуг, які мали зв’язки з цією вершиною (v9, v10, v19). 5) П’ятий підграф утворився після видалення вершин x12 і x13 та 14 та всіх дуг, які мали зв’язки з цими вершинами (v14, v15, v16, v17, v18)

Нижче наведений табличний та ілюстративний матеріал для пунктів 25–27.

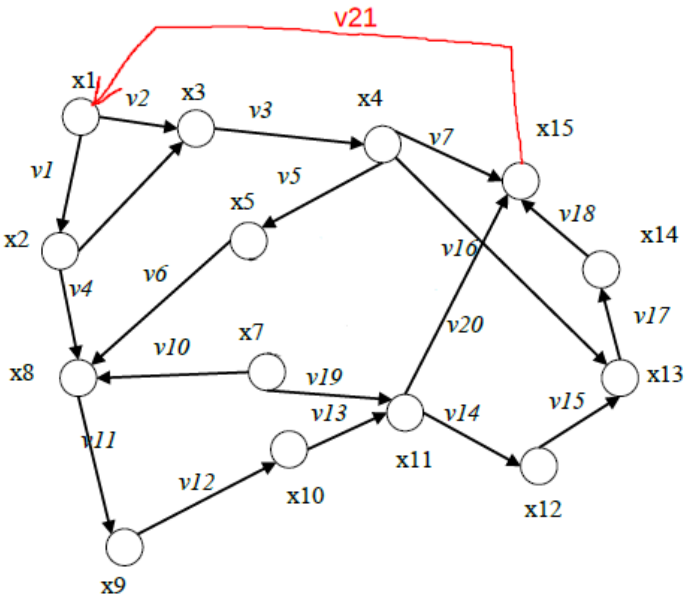
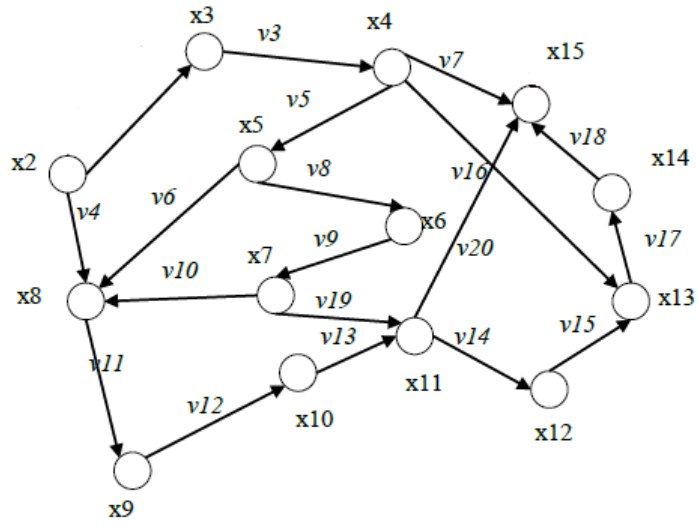
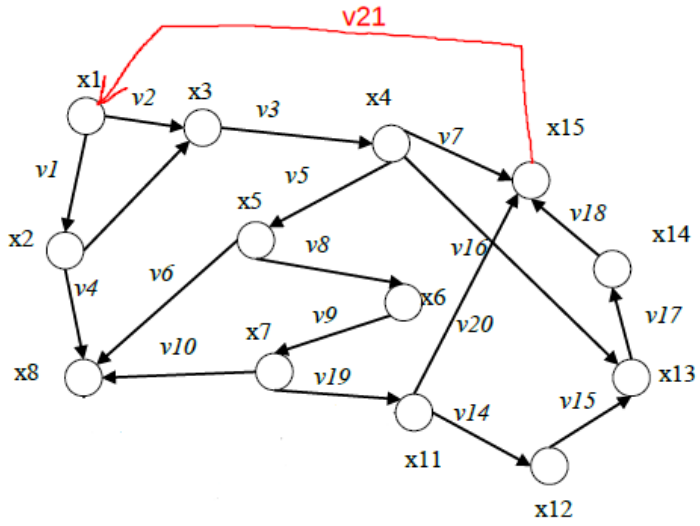
25.

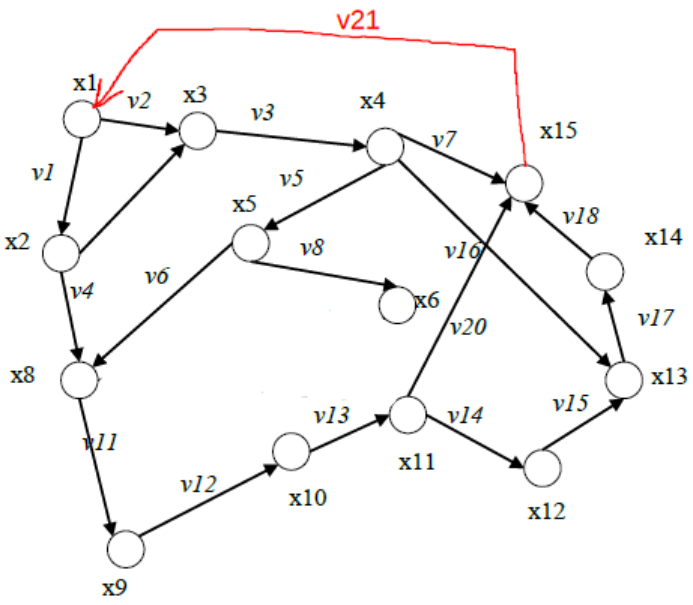
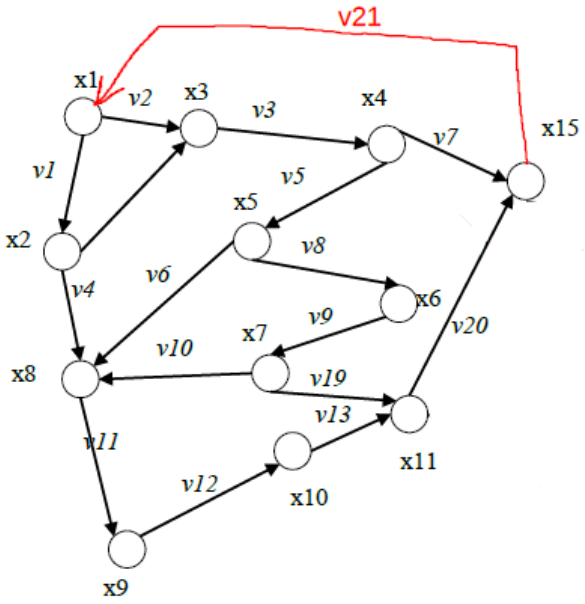
	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15
X1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
X2	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
X3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
X4	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1
X5	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
X6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
X7	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
X8	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
X9	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
X10	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
X11	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
X12	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
X13	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
X14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
X15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

26.

	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20	V21	V22
X1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0
X2	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
X3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1
X4	0	0	-1	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
X5	0	0	0	0	-1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
X6	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
X7	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	1	0	0	0
X8	0	0	0	-1	0	-1	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
X9	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
X10	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0
X11	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	-1	1	0	0
X12	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0	0	0	0
X13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	1	0	0	0	0	0
X14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
X15	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	-1	0	-1	1	0

27.

1	v21 x1 v2 x3 v3 x4 v7 x15 v1 v5 x5 v18 x2 x14 v16 v6 v4 v17 v20 x7 v10 x8 v19 x13 v15 v13 v14 v11 x11 v12 x10 x12 x9
2	x3 v3 x4 v7 x15 v5 x5 v18 x2 v8 v16 x14 v6 v4 x6 v9 v17 v20 x7 v10 x8 v19 x13 v15 v13 v14 v11 x11 v12 x10 x12 x9
3	v21 x1 v2 x3 v3 x4 v7 x15 v1 v5 x5 v18 x2 v8 v16 x14 v6 v4 x6 v9 v17 v20 x7 v10 x8 v19 x13 v15 v14 x11 x12

4	
5	

Висновки до індивідуальної роботи

Під час виконання індивідуальної роботи був проведений загальний аналіз заданого графу за 27 параметрами. Виконання роботи дало змогу закріпити на практиці розуміння основних понять теорії графів.

Індивідуальна робота № 4
Пошук найкоротших шляхів на графі

Для свого графу необхідно виконати таке:

1. Виконати пошук найкоротших шляхів методом пошуку у ширину.

1.1. Вважаючи заданий граф неорієнтованим, задати ваги всіх ребер рівними 1.

1.2. Вибрати вершину з мінімальною кратністю.

1.3. Знайти методом пошуку у ширину найкоротші шляхи (шляхи і їх довжини) до всіх інших вершин графу. Необхідно давати короткі текстові коментарі своїм діям.

1.4. Зобразити кінцевий граф, вказавши для вершин отримані довжини найкоротших шляхів. Окремо привести самі шляхи (послідовності вершин).

1.5. Придумати значеннєве формулювання розв'язаної задачі в контексті тематики заданого графу.

2. Виконати пошук найкоротших шляхів методом Дейкстри.

2.1. Вважаючи заданий граф орієнтованим, задати додатні ваги всіх ребер відповідно до тематики свого графу.

2.2. Вибрати вершину з максимальною кількістю вихідних ребер.

2.3. Знайти методом Дейкстри найкоротші шляхи (шляхи і їх довжини) до всіх інших вершин графу. Необхідно давати короткі покрокові коментарі своїм діям.

2.4. Зобразити кінцевий граф, указавши для вершин отримані довжини найкоротших шляхів. Окремо привести самі шляхи (послідовності вершин).

2.5. Придумати значеннєве формулювання розв'язаної задачі в контексті тематики заданого графу.

3. Виконати пошук найкоротших шляхів методом Белмана–Форда.

3.1. Для графу із завдання № 2 задати двом несуміжним ребрам від'ємні ваги. Значення цих ваг повинні бути або обрані осмислено відповідно до тематики графу (якщо таке можливо), або задані довільно.

3.2. Вибрати вершину з максимальною кількістю вихідних ребер.

3.3. Знайти методом Белмана–Форда найкоротші шляхи (шляхи і їх довжини) до всіх інших вершин графу. Необхідно давати короткі покрокові коментарі своїм діям.

3.4. Зобразити кінцевий граф, вказавши для вершин отримані довжини найкоротших шляхів. Окремо привести шляхи (послідовності вершин). Якщо у про-

цесі роботи алгоритму буде виявлений цикл із від'ємною вагою, показати цей цикл (послідовність вершин і ребер) та його вагу.

3.5. Придумати значеннєве формулювання розв'язаної задачі в контексті тематики заданого графу.

Можливі випадки, коли знаходження шляхів до деяких вершин буде неможливим. У цьому випадку потрібно пояснити причину виникнення такої ситуації, а також придумати їй смислове пояснення в контексті тематики заданого графу.

Оформлення звіту з індивідуальної роботи

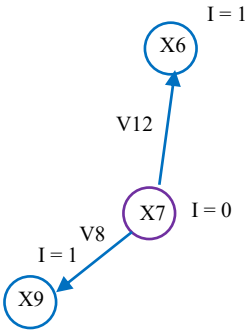
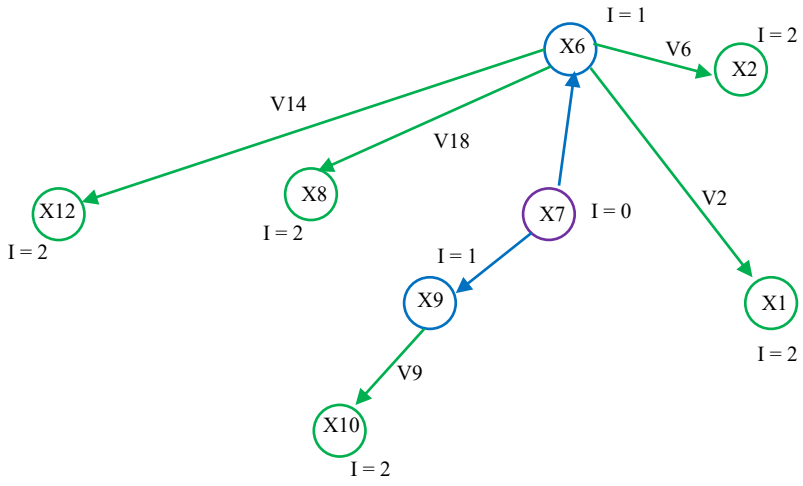
Звіт з індивідуальної роботи оформлюється від руки або набирається на комп'ютері. Для звіту треба використовувати аркуші формату А4, заповнені з однієї сторони.

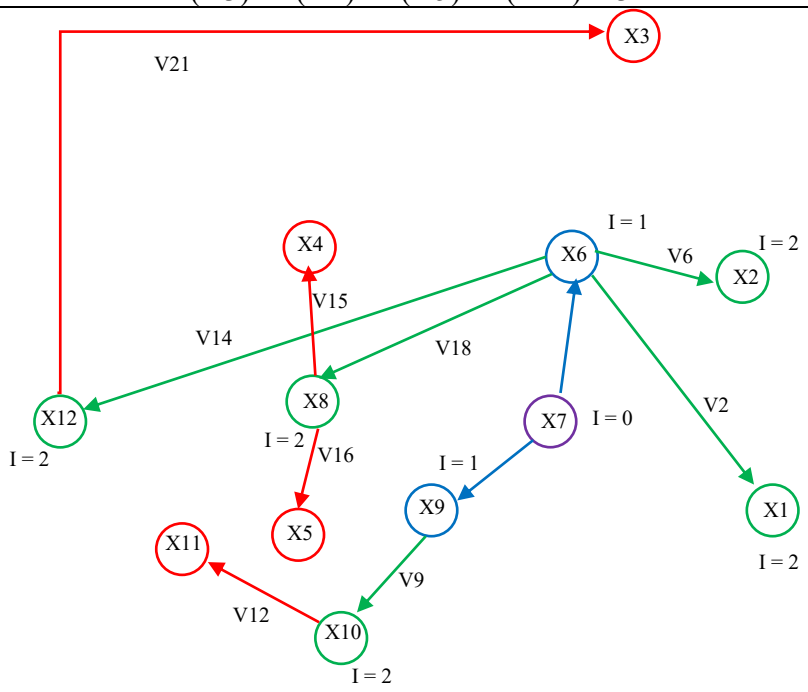
Звіт повинен містити такі обов'язкові елементи:

– титульний аркуш, оформлений за вимогами Університету;
– граф, підписаний викладачем (копія);
– результати роботи (покрокове виконання завдань 1–3);
– висновки до індивідуальної роботи (2–3 рядки у довільній формі).

Приклад виконання індивідуальної роботи № 4

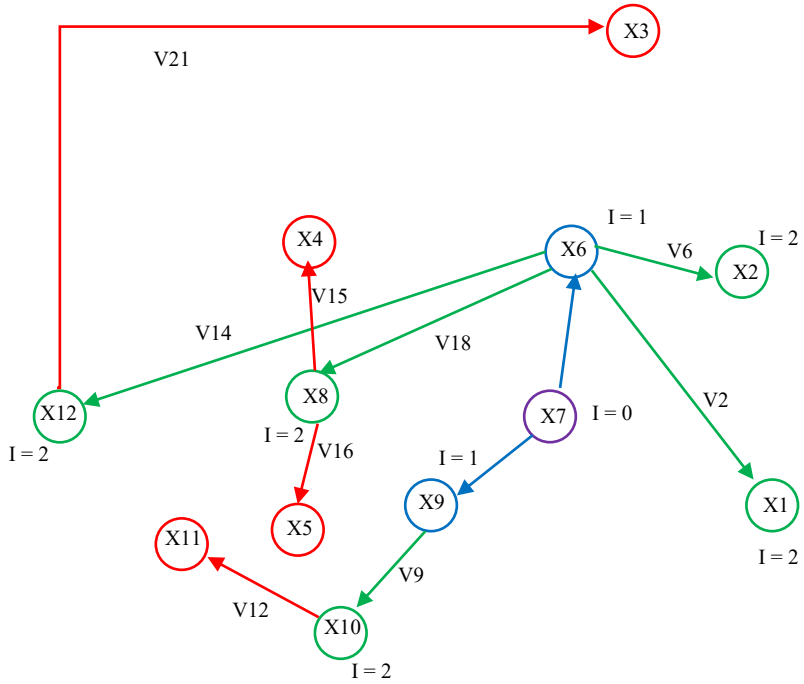
Загальне оформлення завдань має бути схожим із відповідними прикладами в матеріалі лекцій. Нижче наведений приклад оформлення завдання 1. Приклад не є ідеальним, може містити помилки і призначений лише для демонстрації оформлення завдань цієї роботи. Інші завдання треба оформлювати у подібний спосіб.

Етап	**Дії**
	Граф
1	X0 = ∅; X1 = {X7}; X2 = {X1, X2, X3, X4, X5, X6, X8, X9, X10, X11, X12}; I(X7) = 0
	 Починаємо із вершини за найменшою кратністю
2	X0 = {X7}; X1 = {X6, X9}; X2 = {X1, X2, X3, X4, X5, X8, X10, X11, X12}; I(X6) = I(X9) = 1
	 З вершини x7 ідемо до x6 та x9
3	X0 = {X7, X6, X9}; X1 = {X1, X2, X8, X12, X10}; X2 = {X3, X4, X5, X11}; I(X1) = I(X2) = I(X8) = I(X12) = I(X10)= 2
	 З вершини x9 йдемо до x10, а з вершини x6 йдемо до вершин x1, x2, x8, x12
4	X0 = {X7, X6, X9, X1, X2, X8, X12, X10}; X1 = {X3, X4, X5, X11}; X2 =∅;

	I(X3) = I(X4) = I(X5) = I(X11) = 3
	 З вершини x10 йдемо до x11, з x8 до x4 та x5, з x12 до x3

Усі вершини досягнуті, множина X2 порожня, алгоритм завершено.

Кінцевий граф:



Знайдені найкоротші шляхи:

1) {X7, X9, X10, X11}
2) {X7, X6, X8, X5}
3) {X7, X6, X12, X3}
4) {X7, X6, X8, X4}
5) {X7, X6, X2}
6) {X7, X6, X1}

Пояснення: знайдені шляхи показують найкоротші послідовності відвідування магазинів торгового центру «Мегамол», починаючи від магазину «Кава з собою».

Індивідуальна робота № 5
Остови й цикли у графах

Для свого графу необхідно виконати таке:

1. Пошук мінімального остовного дерева.

1.1. Представити заданий граф у зваженому неорієнтованому вигляді.

1.2. Знайти мінімальне остовне дерево методом Прима.

1.3. Знайти мінімальне остовне дерево методом Краскала.

1.4. Пояснити зміст мінімального остовного дерева в контексті індивідуального варіанта завдання.

2. Пошук ейлерового циклу.

2.1. Представити заданий граф у зваженому неорієнтованому вигляді з одиничною кратністю усіх ребер.

2.2. Визначити, чи є граф ейлеровим, напівейлеровим або неейлеровим.

2.3. Якщо граф не є ейлеровим, додати мінімальну кількість ребер, за якої граф стає ейлеровим.

2.4. Знайти ейлерів цикл за допомогою алгоритму Фльорі.

2.5. Представити заданий граф у зваженому орієнтованому вигляді з одиничною кратністю ребер і повторити пункти 2.2–2.4.

2.6. Пояснити зміст ейлерового циклу в контексті індивідуального варіанта завдання.

3. Пошук гамільтонового циклу.

3.1. Із заданого графу сформувати підграф, що складається з 7 вершин. Вершини, якщо можливо, повинні бути обрані в такий спосіб, щоб отриманий підграф ніс якийсь зміст у контексті індивідуального варіанта завдання.

3.2. Вважаючи ребра неорієнтованими, зробити попередній висновок про те, є чи підграф гамільтоновим або напівгамільтоновим.

3.3. Виконати в підграфі пошук гамільтонового циклу за допомогою алгоритму Робертса–Флореса. Достатньо знайти один гамільтонів цикл.

3.4. Якщо жоден гамільтонів цикл не знайдений, виконати в цьому підграфі пошук гамільтонового ланцюга. Якщо цикл знайдений, то гамільтонів ланцюг відповідає знайденому циклу без останнього ребра.

3.5. Зробити остаточний висновок про те, є підграф гамільтоновим, напівгамільтоновим чи негамільтоновим.

3.6. Пояснити зміст гамільтонового циклу й гамільтонового ланцюга в контексті індивідуального варіанта завдання.

Оформлення звіту з індивідуальної роботи

Звіт з індивідуальної роботи оформлюється від руки або набирається на комп'ютері. Для звіту треба використовувати аркуші формату А4, заповнені з однієї сторони.

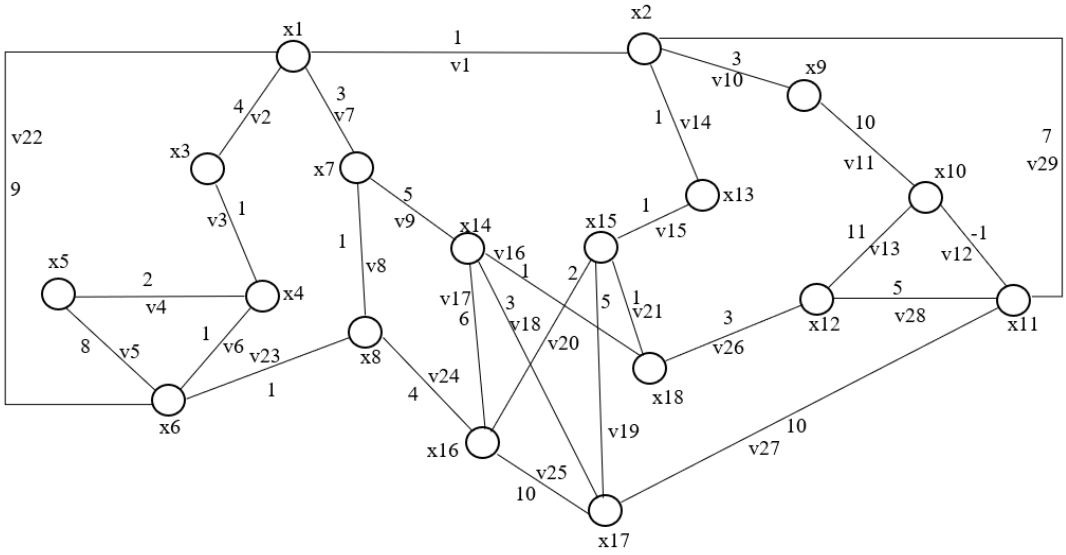
Звіт повинен містити такі обов'язкові елементи:

– титульний аркуш, оформлений за вимогами Університету;
– граф, підписаний викладачем (копія);
– результати роботи (покрокове виконання завдань 1–3);
– висновки до індивідуальної роботи (2–3 рядки у довільній формі).

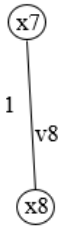
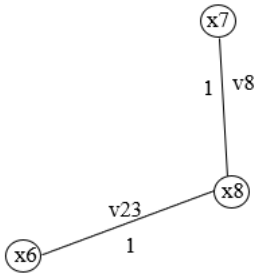
Приклад виконання індивідуальної роботи № 5

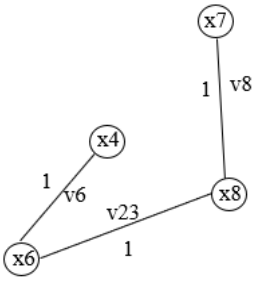
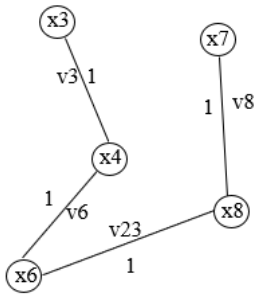
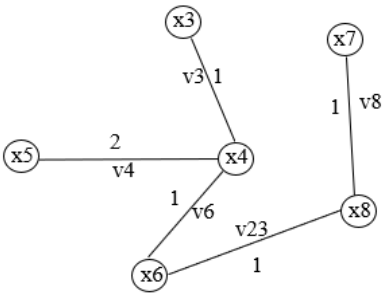
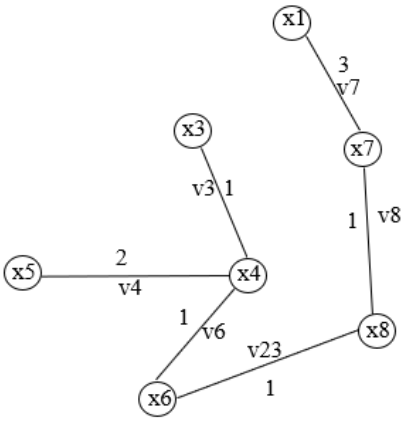
Загальне оформлення завдань має бути схожим із відповідними прикладами в матеріалі лекцій. Нижче наведений приклад оформлення завдання 1. Приклад не є ідеальним, може містити помилки і призначений лише для демонстрації оформлення завдань цієї роботи. Інші завдання треба оформлювати у подібний спосіб.

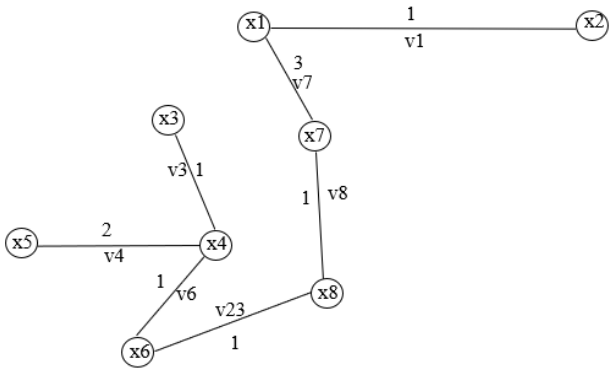
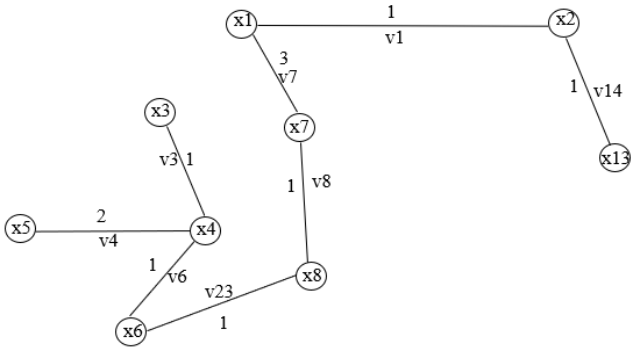
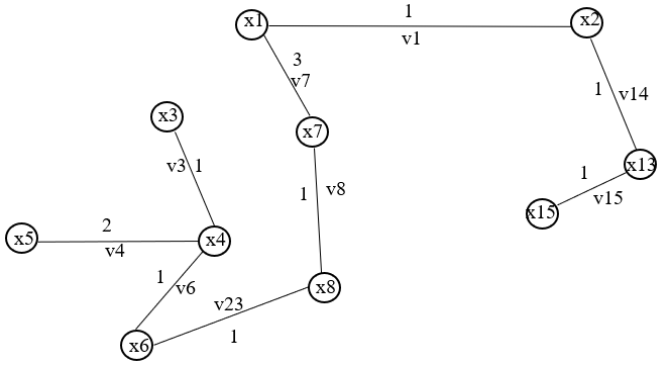
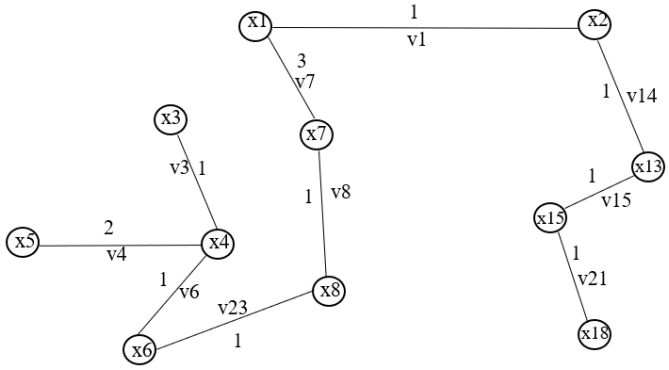
1.1. Початковий граф.

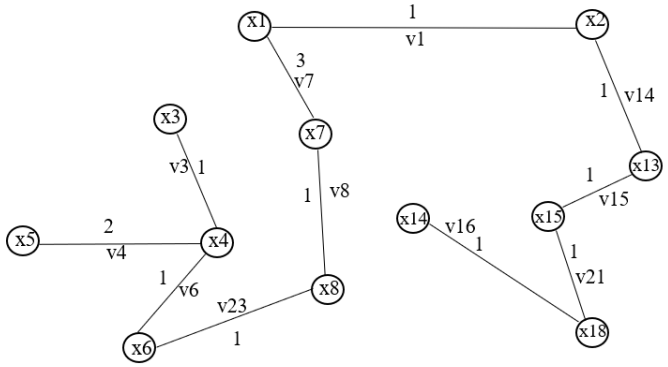
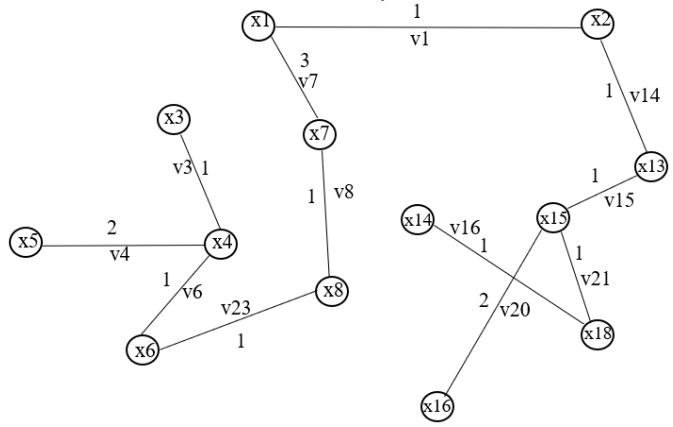
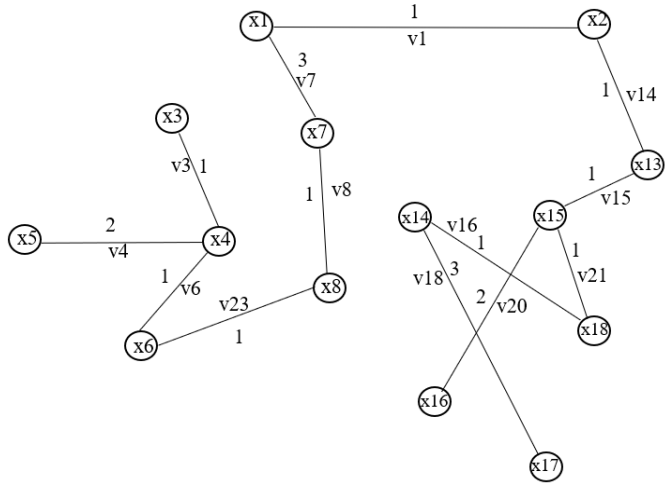


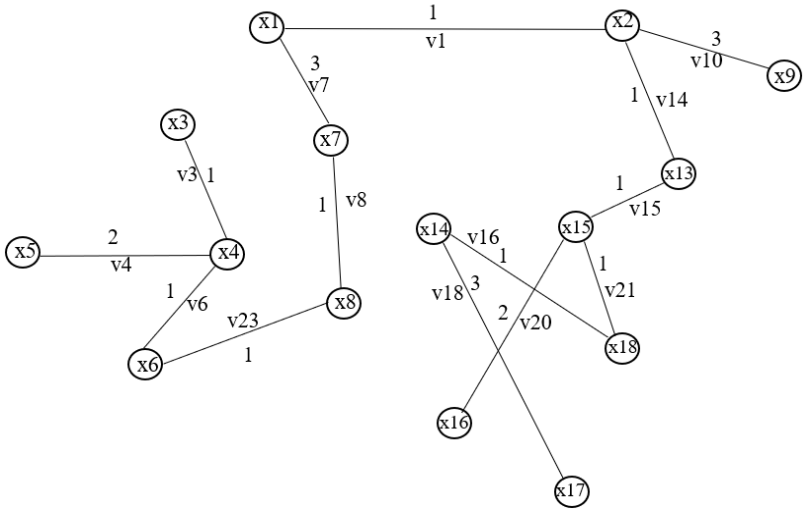
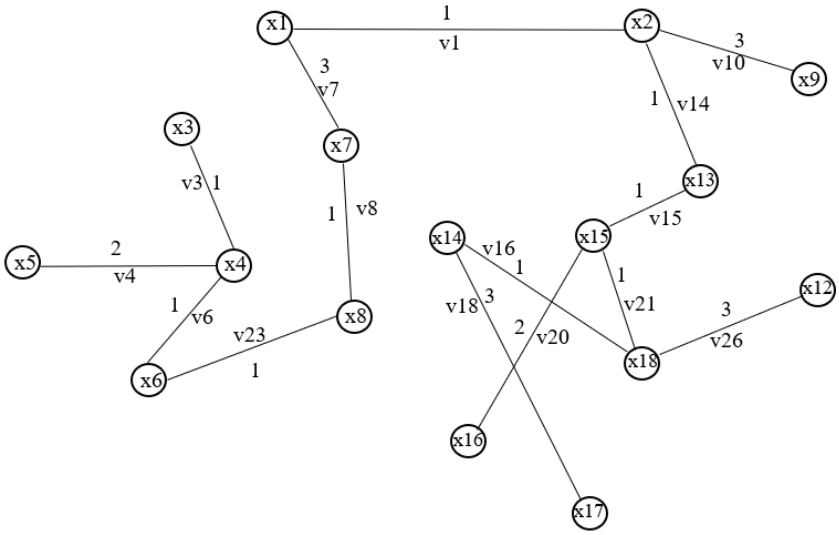
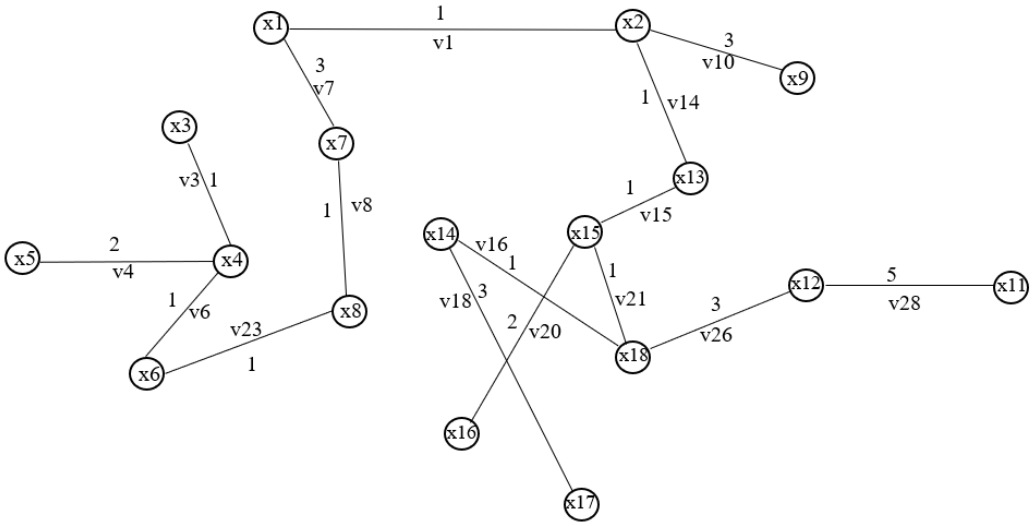
1.2. Алгоритм Прима.

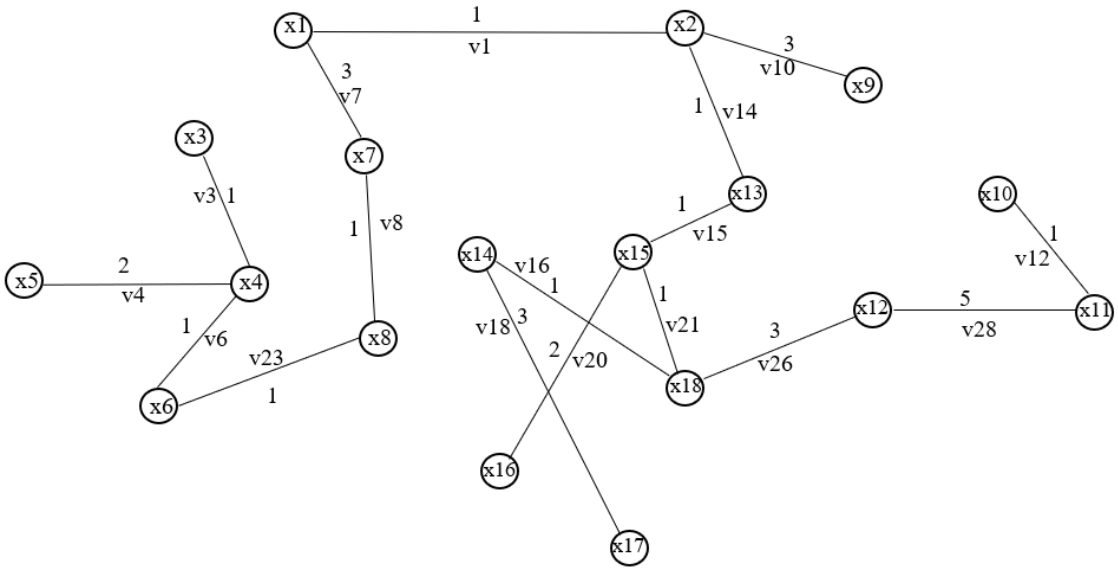
Крок	Дії, граф
0	Виберемо довільну вершину, наприклад, x7. Додаємо x7 у множину X вершин, вже включених до остову: X={x7} 
1	Розглянемо всі ребра, інцидентні вершинам із множини X, і вибираємо ребро з мінімальною вагою, що не приводить до утворення циклів. У нашому випадку це ребро V8. Тоді додаємо вершину x8: X={x7,x8} 
2	Вибираємо ребро V23 і додаємо вершину x6. X={x7,x8,x6} 

Крок	Дії, граф
3	Вибираємо ребро V6 і додаємо вершину x4. X={x7,x8,x6,x4} 
4	Вибираємо ребро V3 і додаємо вершину x3. X={x7,x8,x6,x4,x3} 
5	Вибираємо ребро V4 і додаємо вершину x5. X={x7,x8,x6,x4,x3,x5} 
6	Вибираємо ребро V7 і додаємо вершину x1. X={x7,x8,x6,x4,x3,x5,x1} 

Крок	Дії, граф
7	Вибираємо ребро V1 і додаємо вершину x2. X={x7,x8,x6,x4,x3,x5,x1,x2} 
8	Вибираємо ребро V14 і додаємо вершину x13. X={x7,x8,x6,x4,x3,x5,x1, x2,x13} 
9	Вибираємо ребро V15 і додаємо вершину x15. X={x7,x8,x6,x4,x3,x5,x1,x2,x13,x15} 
10	Вибираємо ребро V21 і додаємо вершину x18. X={x7,x8,x6,x4,x3,x5,x1, x2,x13,x15,x18} 

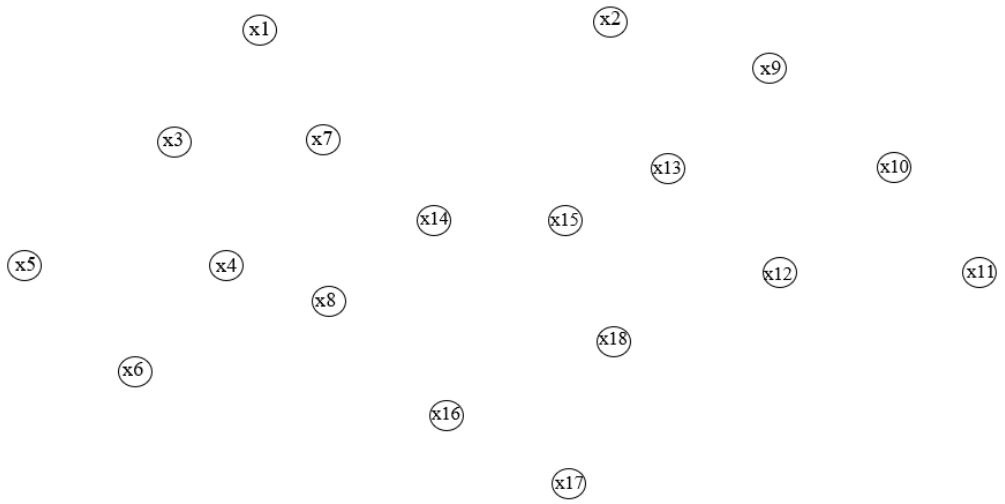
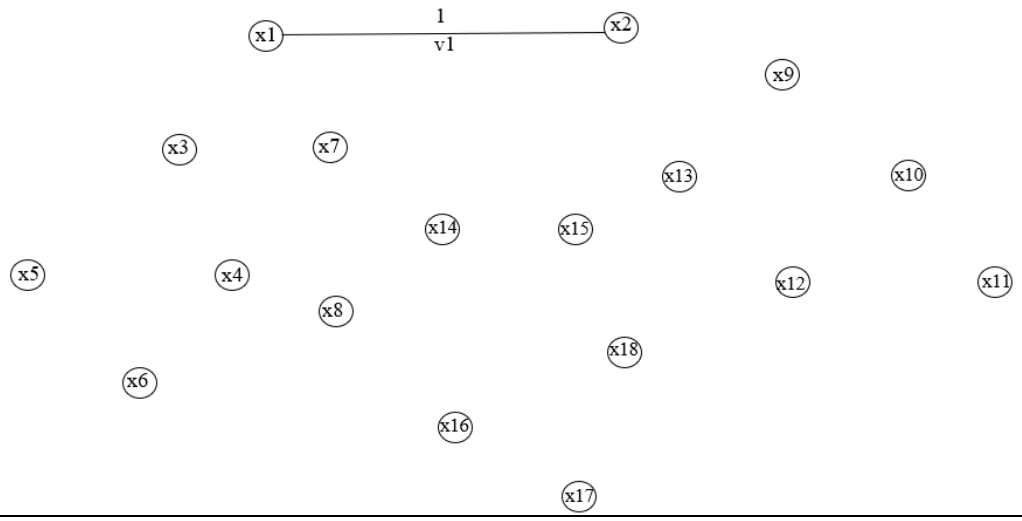
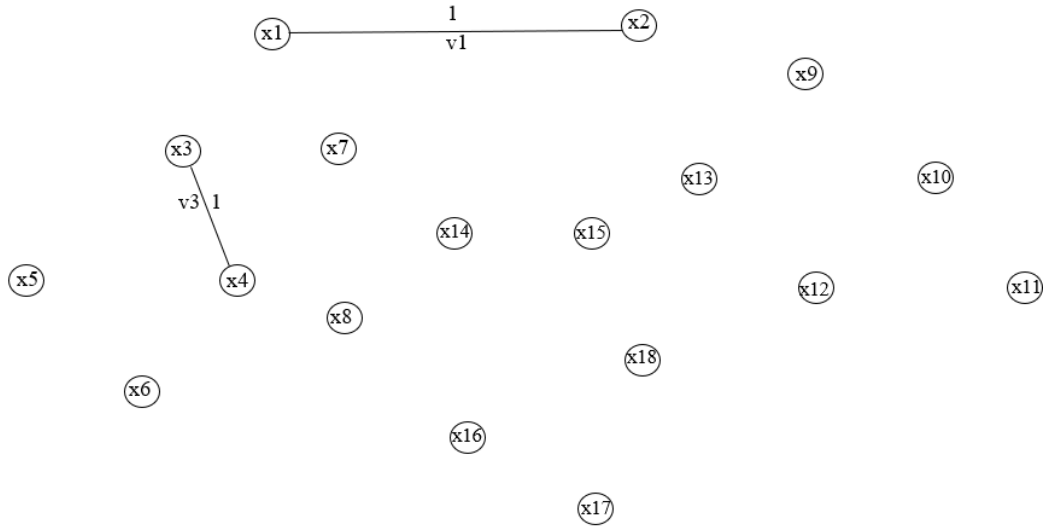
Крок	Дії, граф
11	Вибираємо ребро V16 і додаємо вершину x14. X={x7,x8,x6,x4,x3,x5,x1,x2,x13,x15,x18,x14} 
12	Вибираємо ребро V20 і додаємо вершину x16. X={x7,x8,x6,x4,x3,x5,x1,x2,x13,x15,x18,x14,x16} 
13	Вибираємо ребро V18 і додаємо вершину x17. X={x7,x8,x6,x4,x3,x5,x1,x2,x13,x15,x18,x14,x16,x17} 

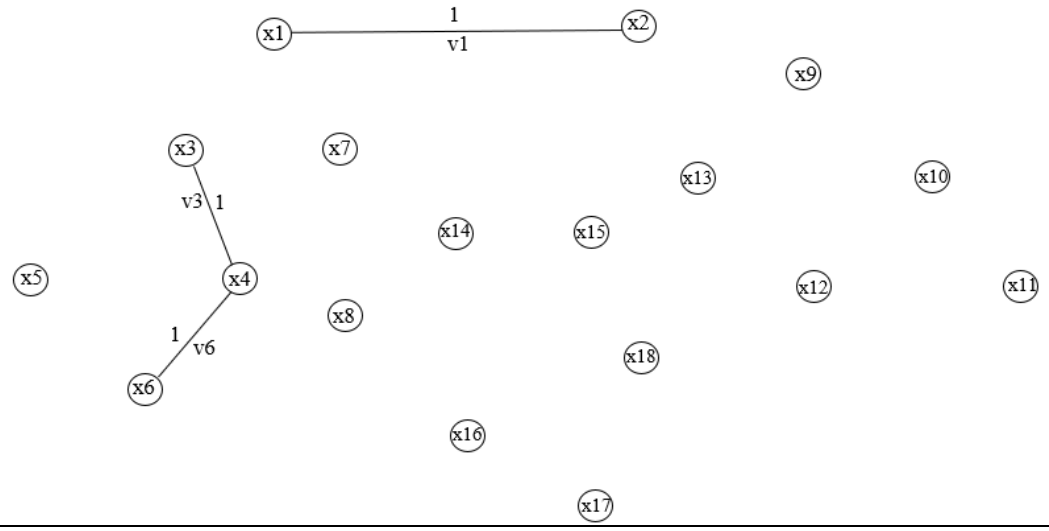
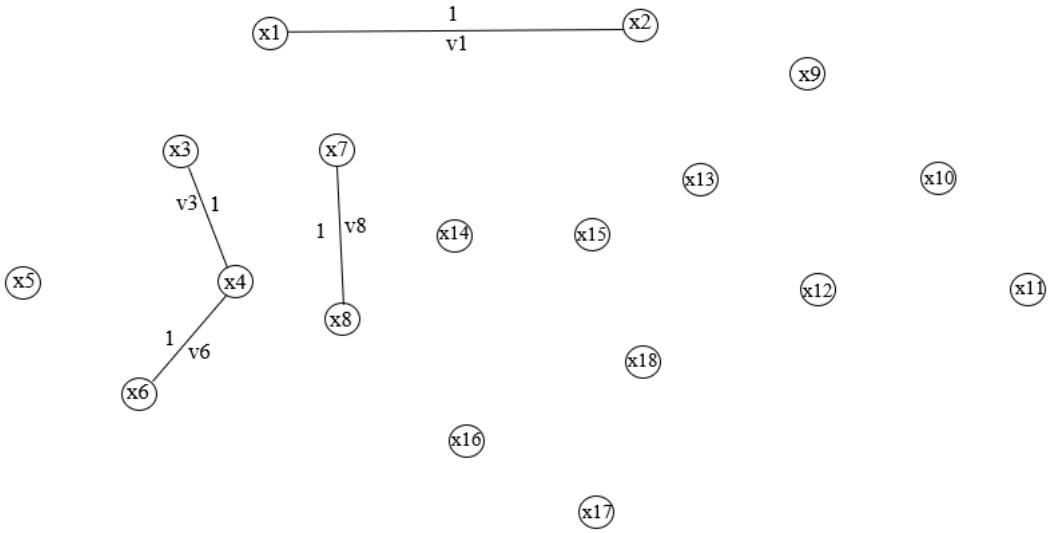
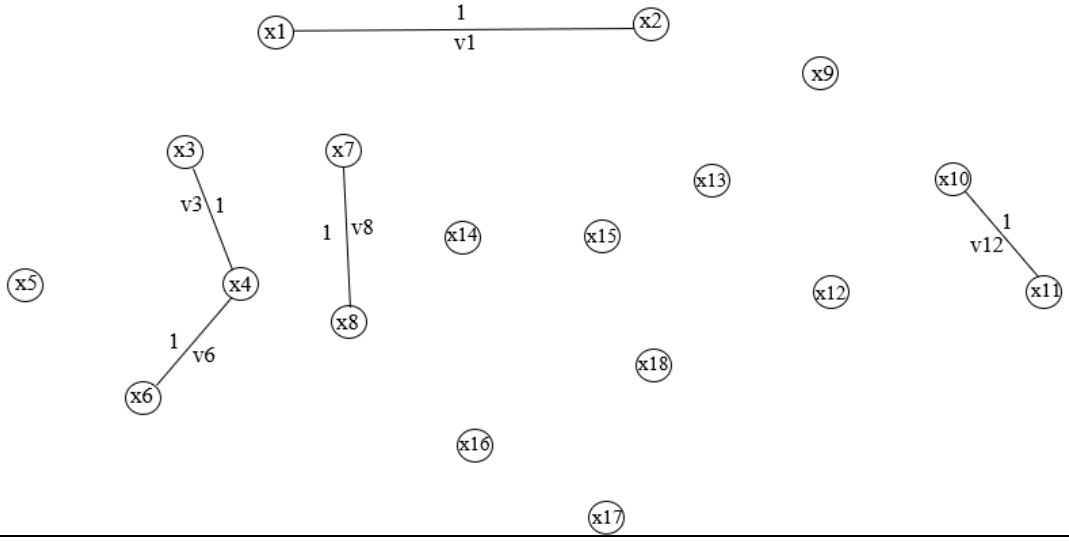
Крок	Дії, граф
14	Вибираємо ребро V10 і додаємо вершину x9. X={x7,x8,x6,x4,x3,x5,x1, x2,x13,x15,x18,x14,x16,x17,x9}. 
15	Вибираємо ребро V26 і додаємо вершину x12. X={x7,x8,x6,x4,x3,x5,x1, x2,x13,x15,x18,x14,x16,x17,x9,x12}. 
16	Вибираємо ребро V28 і додаємо вершину x11. X={x7,x8,x6,x4,x3,x5,x1, x2,x13,x15,x18,x14,x16,x17,x9,x12,x11}. 

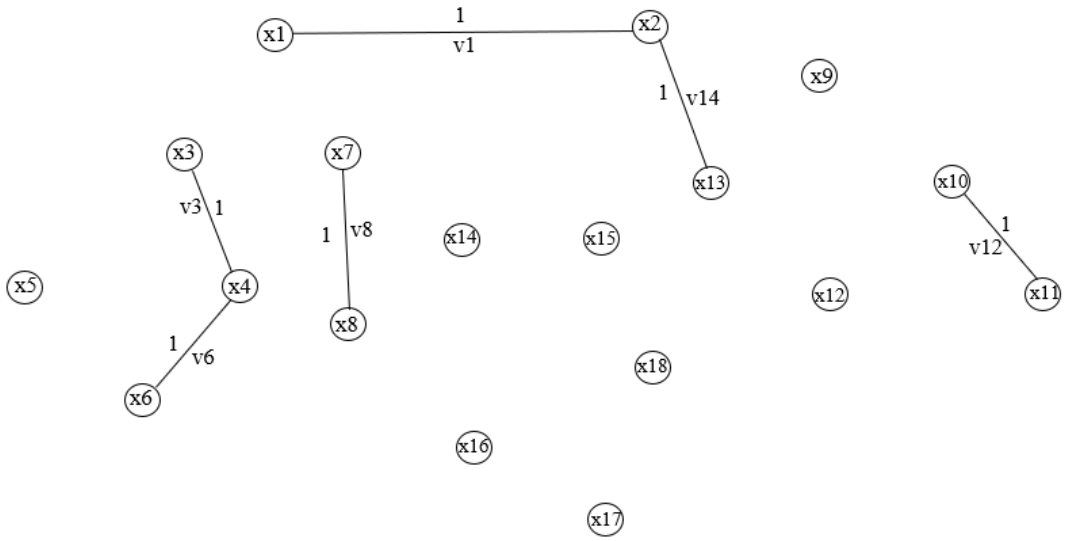
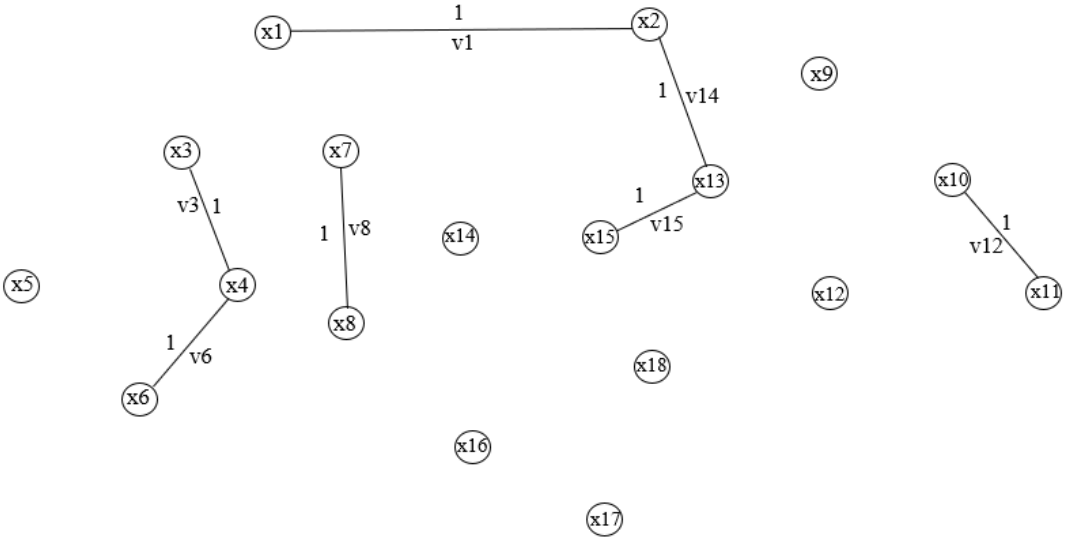
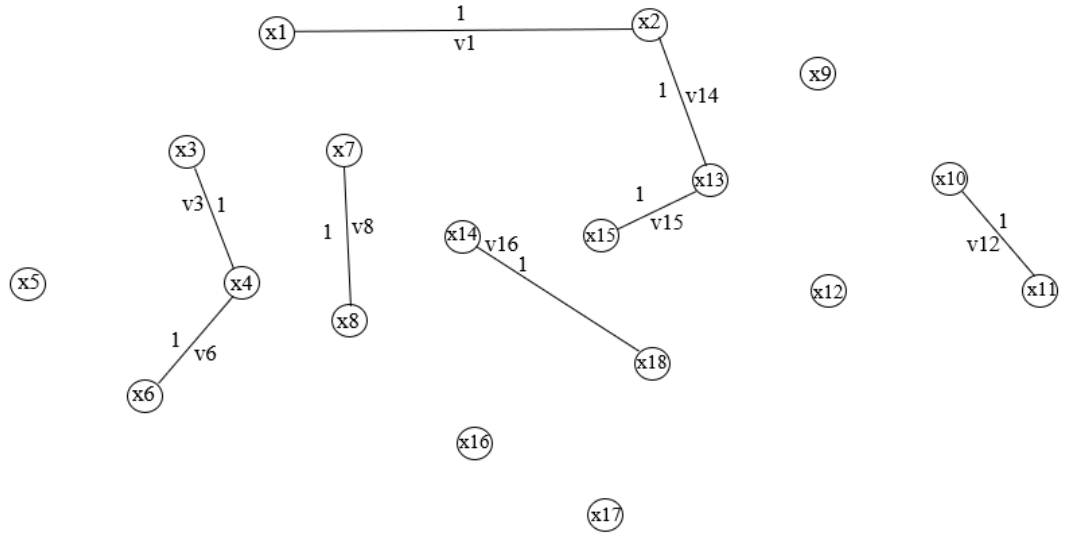
Крок	Дії, граф
17	Вибираємо ребро V12 і додаємо вершину x10. X={x7,x8,x6,x4,x3,x5,x1, x2,x13,x15,x18,x14,x16,x17,x9,x12,x11,x10}. 
18	Остов містить усі вершини графу та не має циклів. Цей остов є мінімальним остовним деревом з вагою (1+1+1+1+2+3+1+1+1+1+1+2+3+3+3+5+1)=31

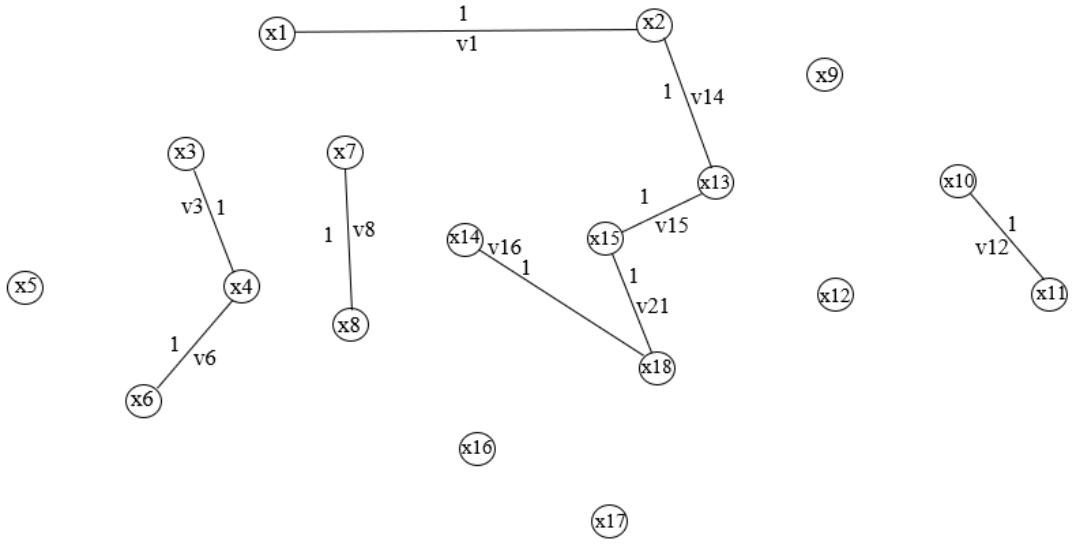
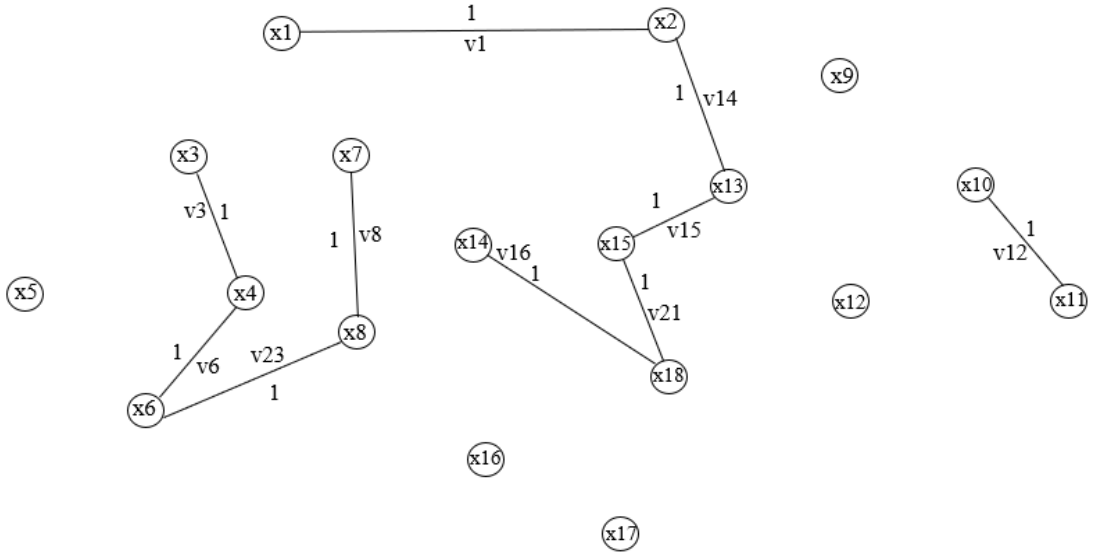
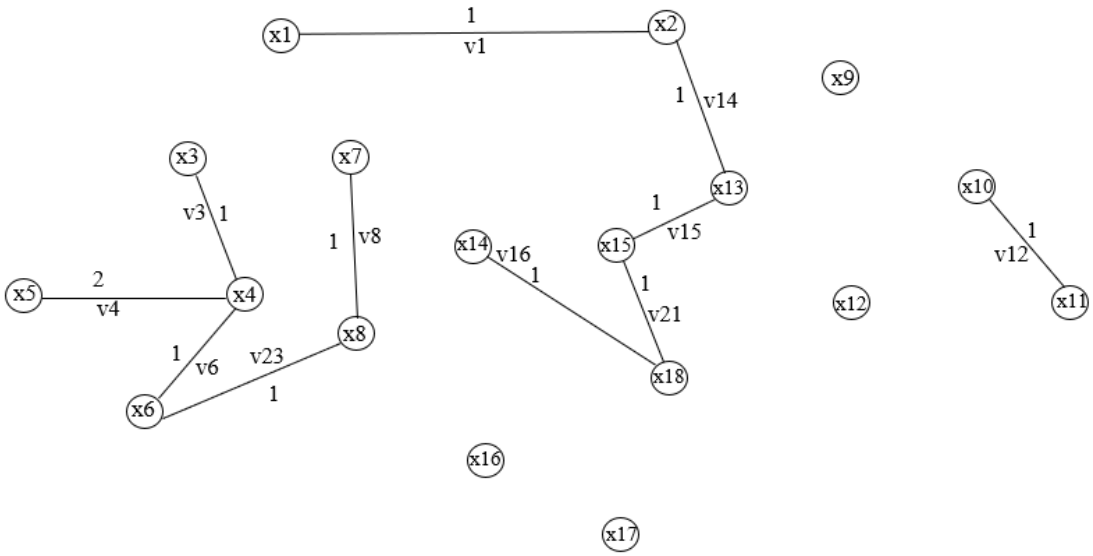
1.3. Алгоритм Краскала.

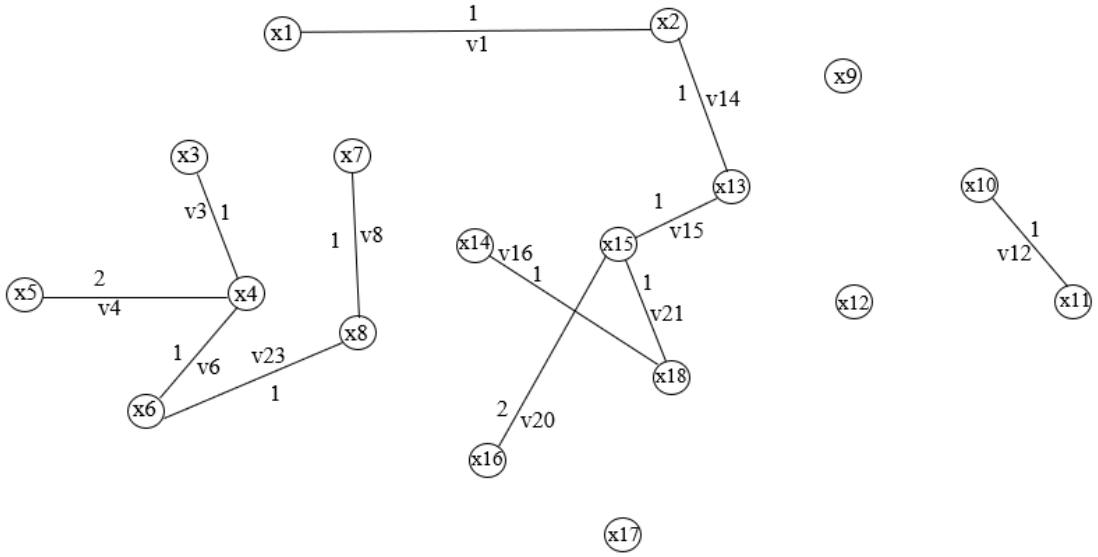
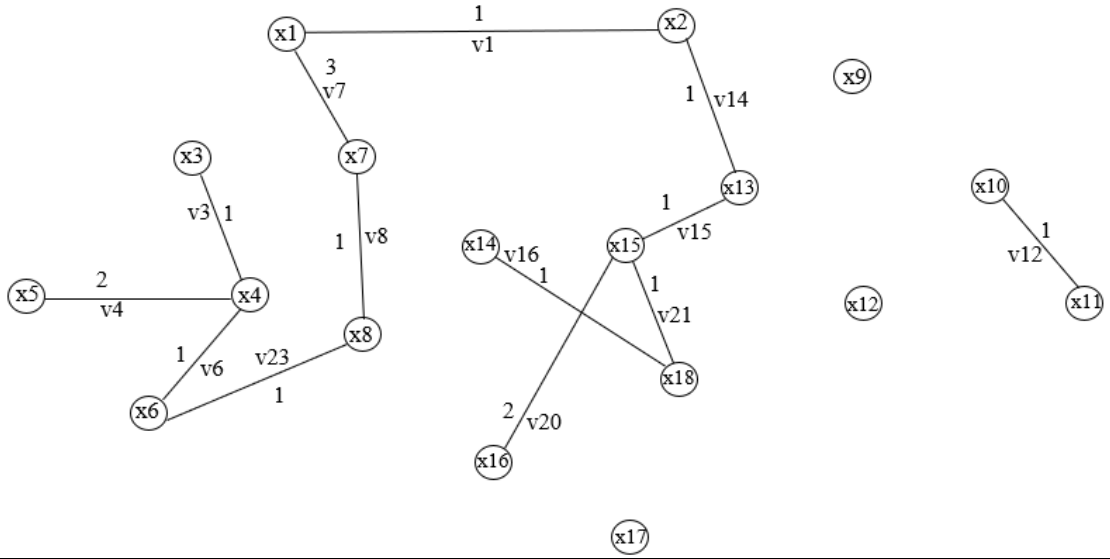
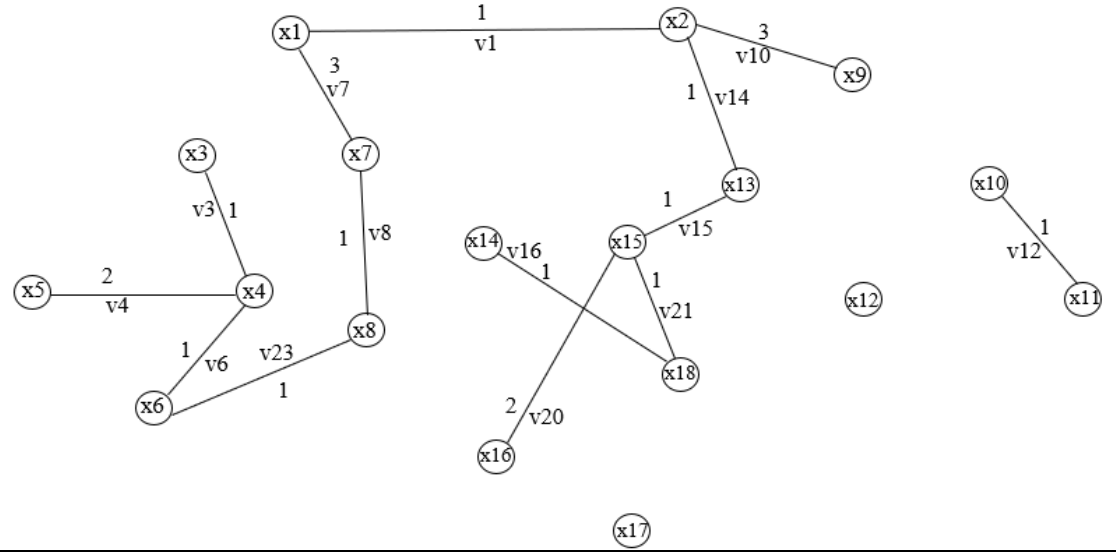
Крок	Дії	
0	Ребра за зростанням ваг:	
	V1 d(v1)=1	V26 d(v26)=3
	V3 d(v3)=1	V2 d(v2)=4
	V6 d(v6)=1	V24 d(v24)=4
	V8 d(v8)=1	V9 d(v9)=5
	V12 d(v12)=1	V19 d(v19)=5
	V14 d(v14)=1	V28 d(v28)=5
	V15 d(v15)=1	V17 d(v17)=6
	V16 d(v16)=1	V29 d(v29)=7
	V21 d(v21)=1	V5 d(v5)=8
	V23 d(v23)=1	V22 d(v22)=9
	V4 d(v4)=2	V11 d(v11)=10
	V20 d(v20)=2	V25 d(v25)=10
	V7 d(v7)=3	V27 d(v27)=10
	V10 d(v10)=3	V13 d(v13)=11
	V18 d(v18)=3	

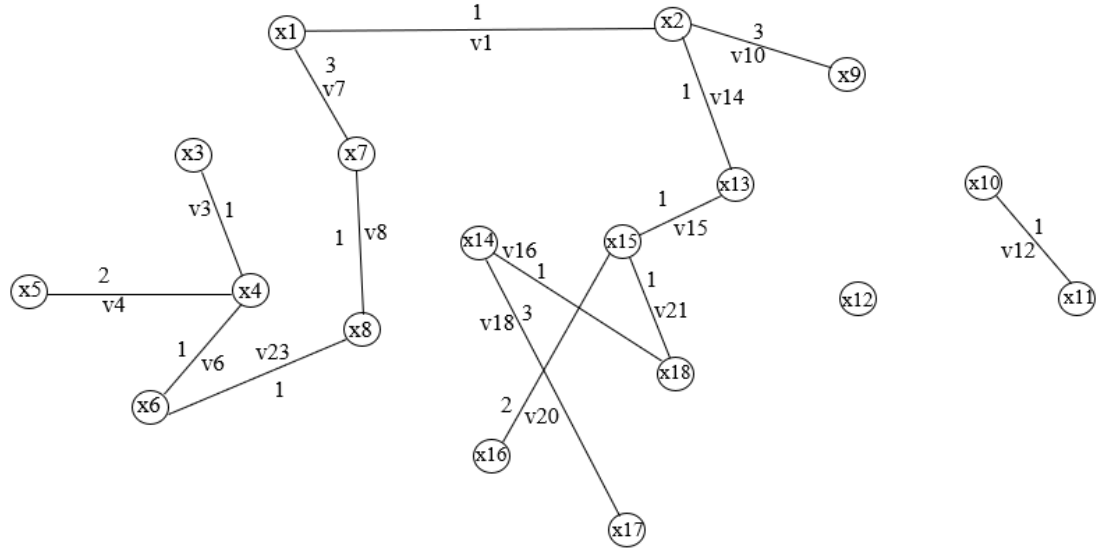
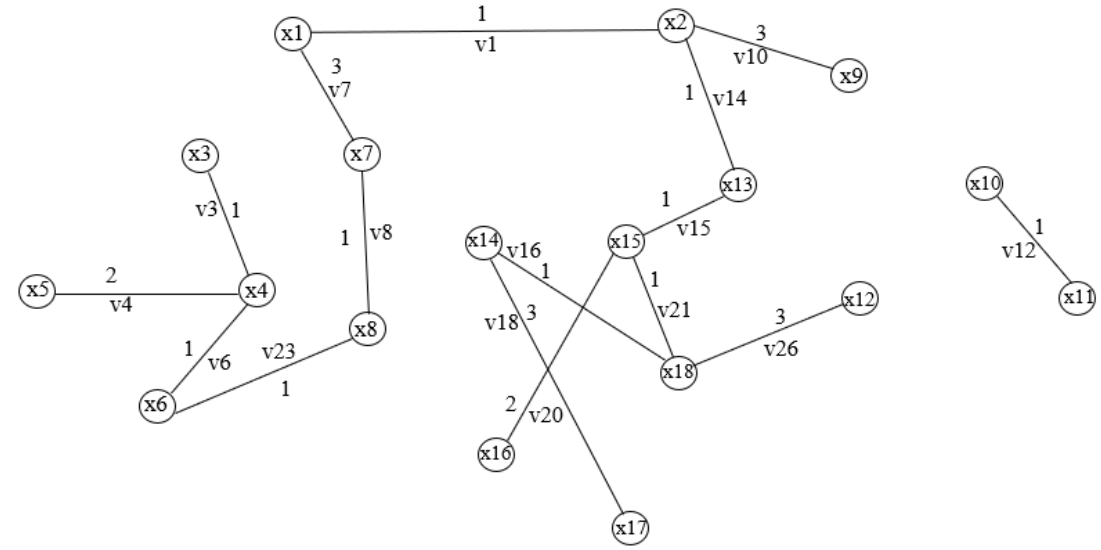
<table>
<tr><td colspan="2">Початковий граф

</td></tr>
<tr><td>1</td><td>Вибираємо ребро v1. Додавання ребра v1 не приводить до утворення циклів, тому додаємо v1

</td></tr>
<tr><td>2</td><td>Вибираємо ребро v3. Додавання ребра v3 не приводить до утворення циклів, тому додаємо v3

</td></tr>
</table>

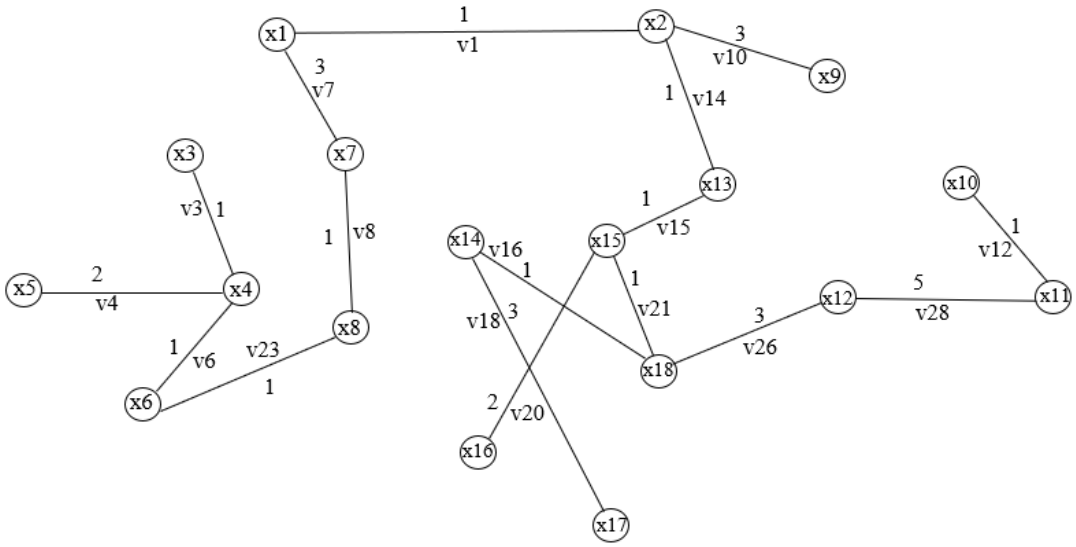
3	Вибираємо ребро v6. Додавання ребра v6 не приводить до утворення циклів, тому додаємо v6 
4	Вибираємо ребро v8. Додавання ребра v8 не приводить до утворення циклів, тому додаємо v8 
5	Вибираємо ребро v12. Додавання ребра v12 не приводить до утворення циклів, тому додаємо v12 

6	Вибираємо ребро v14. Додавання ребра v14 не приводить до утворення циклів, тому додаємо v14 
7	Вибираємо ребро v15. Додавання ребра v15 не приводить до утворення циклів, тому додаємо v15 
8	Вибираємо ребро v16. Додавання ребра v16 не приводить до утворення циклів, тому додаємо v16 

9	Вибираємо ребро v21. Додавання ребра v21 не приводить до утворення циклів, тому додаємо v21 
10	Вибираємо ребро v23. Додавання ребра v23 не приводить до утворення циклів, тому додаємо v23 
11	Вибираємо ребро v4. Додавання ребра v4 не приводить до утворення циклів, тому додаємо v4 

12	Вибираємо ребро v20. Додавання ребра v20 не приводить до утворення циклів, тому додаємо v20 
13	Вибираємо ребро v7. Додавання ребра v7 не приводить до утворення циклів, тому додаємо v7 
14	Вибираємо ребро v10. Додавання ребра v10 не приводить до утворення циклів, тому додаємо v10 

15	Вибираємо ребро v18. Додавання ребра v18 не приводить до утворення циклів, тому додаємо v18 
16	Вибираємо ребро v26. Додавання ребра v26 не приводить до утворення циклів, тому додаємо v26 
17	Вибираємо ребро v2. Додавання ребра v2 приводить до утворення циклу, тому ми не додаємо v2, граф залишається без змін
18	Вибираємо ребро v24. Додавання ребра v24 приводить до утворення циклу, тому ми не додаємо v24, граф залишається без змін
19	Вибираємо ребро v9. Додавання ребра v9 приводить до утворення циклу, тому ми не додаємо v9, граф залишається без змін
20	Вибираємо ребро v19. Додавання ребра v19 приводить до утворення циклу, тому ми не додаємо v19, граф залишається без змін
21	Вибираємо ребро v28. Додавання ребра v28 не приводить до утворення циклів, тому додаємо v28

	
22	Оскільки кількість ребер, включених в остов, досягла величини, на одиницю меншої кількості вершин у графі, алгоритм завершується. Отриманий граф є мінімальним остовним деревом вихідного графу і має вагу 31

1.4. Уявимо, що вершини графу – це елементи гардеробу, виставлені на виставці одягу. Для кожного з елементів гардеробу зроблена вітрина, до якої підведене електричне освітлення. Ваги ребер графу – це довжини проводів, які можна прокласти між відповідними вітринами.

Тоді мінімальне остовне дерево в цьому графі означатиме мінімальну загальну довжину електричного проводу, щоб підвести освітлення до усіх вітрин виставки.

Індивідуальна робота № 6
Потоки в мережах

Для заданого графу G необхідно виконати таке:

1. Вважаючи граф орієнтованим, впорядкувати вершини за зменшенням сум вхідних і вихідних ступенів.

2. Перші десять вершин (з найбільшими сумарними ступенями) залишити, інші вершини та інцидентні їм ребра видалити.

3. В отриманому підграфі G' кожній дузі поставити у відповідність деяке числове значення, що відповідає пропускній здатності цієї дуги.

4. У якості джерела вибрати вершину з максимальною кількістю вихідних ребер, у якості стоку – вершину з максимальною кількістю вхідних ребер.

5. Вважаючи початкові потоки в кожній дузі рівними нулю, знайти в підграфі G' величину максимального потоку від джерела до стоку.

6. Вважаючи початкові потоки в кожній дузі рівними пропускній здатності дуги з мінімальною пропускною здатністю, знайти в підграфі G' величину максимального потоку від джерела до стоку.

7. Пояснити зміст пошуку максимального потоку в контексті індивідуального варіанта завдання.

Підготовка звіту з індивідуальної роботи

У звіті треба навести заданий граф G і результати виконання кожного пункту завдання. Під час виконання пунктів 5 і 6 не обов’язково давати великі текстові пояснення – достатньо лише навести вигляд підграфу після кожного етапу алгоритму (після кожної позначки вершин та після кожної зміни потоку).

Приклад виконання індивідуальної роботи № 6

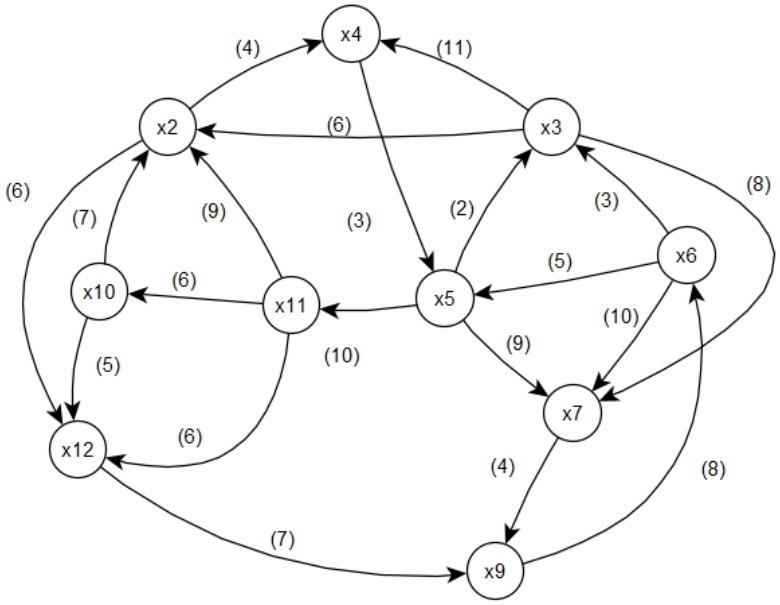
Нижче наведений приклад виконання завдань 1–5. Приклад не є ідеальним, може містити помилки і не відповідати високій кількості балів. Єдине призначення прикладу – демонстрація загального оформлення індивідуальної роботи.

1. Вважаючи граф орієнтованим, упорядкувати вершини за зменшенням сум вхідних і вихідних ступенів.

Вершина	Кількість вхідних ребер	Кількість вихідних ребер	Всього
x3	2	4	6
x2	3	3	6
x7	3	2	5
x5	2	3	5
x12	3	1	4
x11	1	3	4
x9	3	1	4
x6	1	3	4
x4	3	1	4
x10	1	2	3
x1	2	1	3
x8	1	1	2

2. Обираємо перші 10 вершин (з найбільшими ступенями):

{x3, x2, x7, x5, x12, x11, x9, x5, x4, x10}



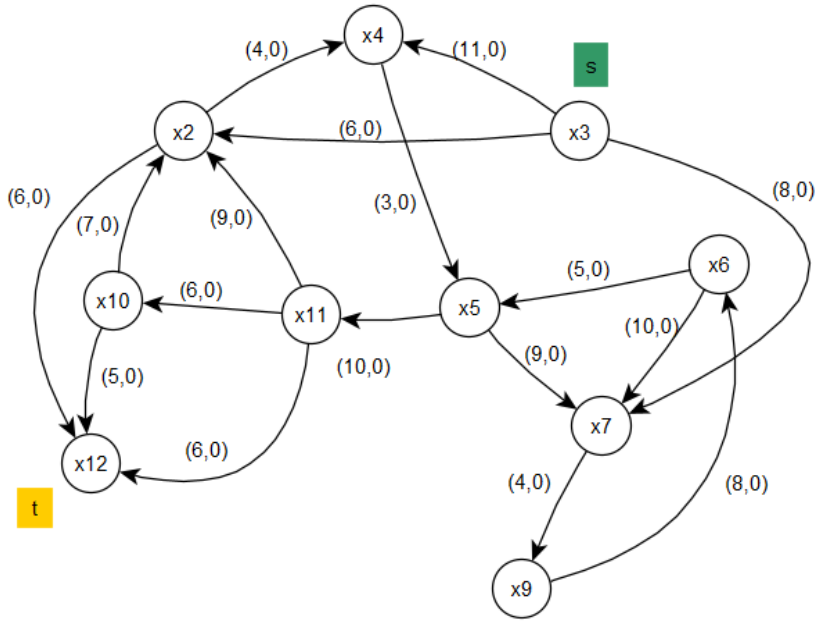
3. Підграф G' (з видаленими вершинами x1 та x8), де кожній дузі поставлене у відповідність числове значення, що відповідає пропускній здатності даної дуги:

4. Джерело – вершина x3, стік – вершина x12.

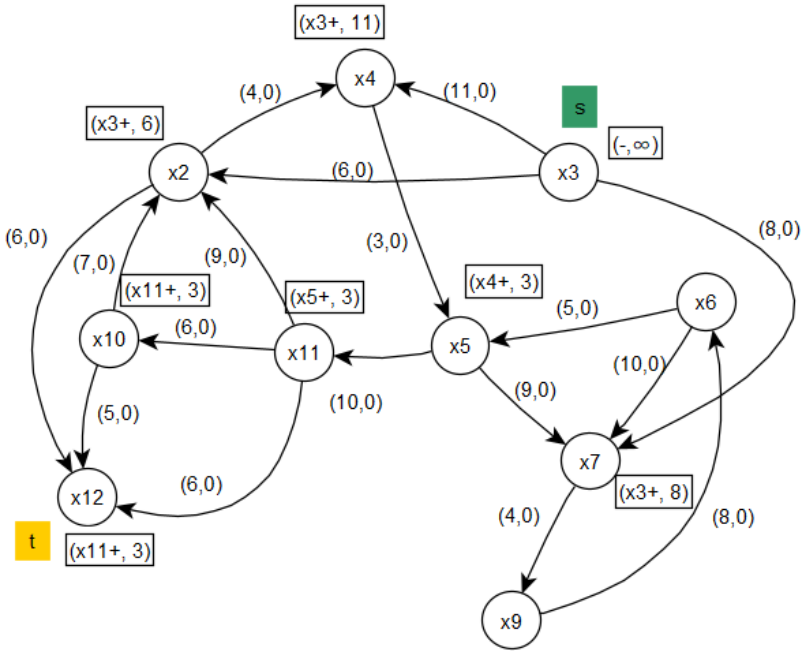
Оскільки джерело не має вхідного потоку, а стік не має містити вихідного, то ребра v6, v7, v24 також можна видалити.

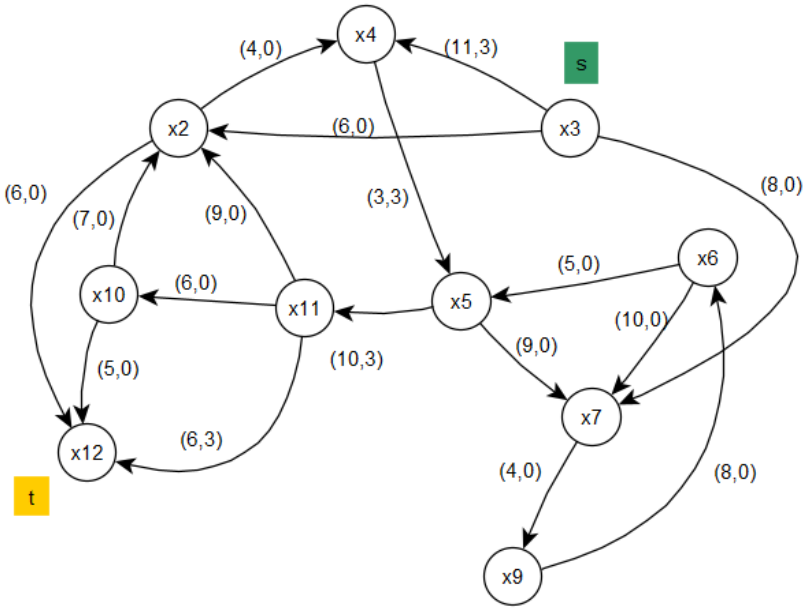
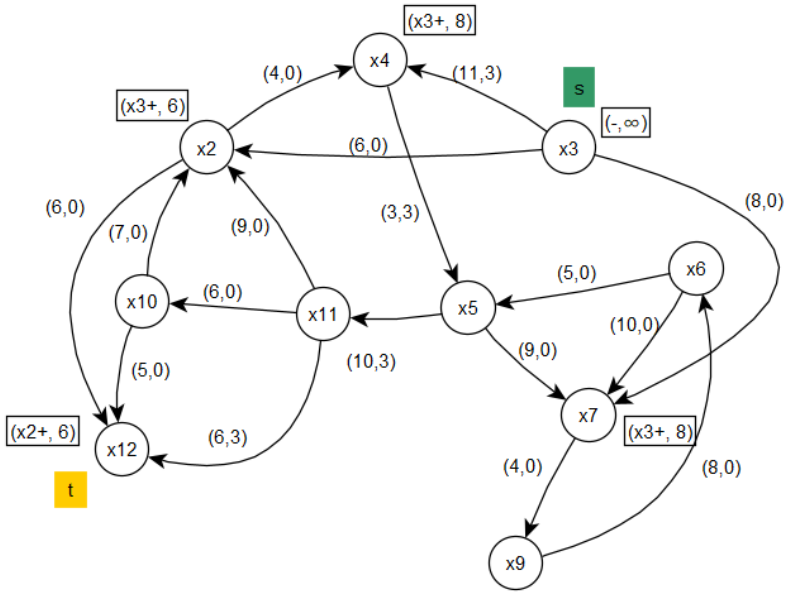
5. Вважаючи початкові потоки в кожній дузі рівним нулю, знайти у підграфі G величину максимального потоку від джерела до стоку.

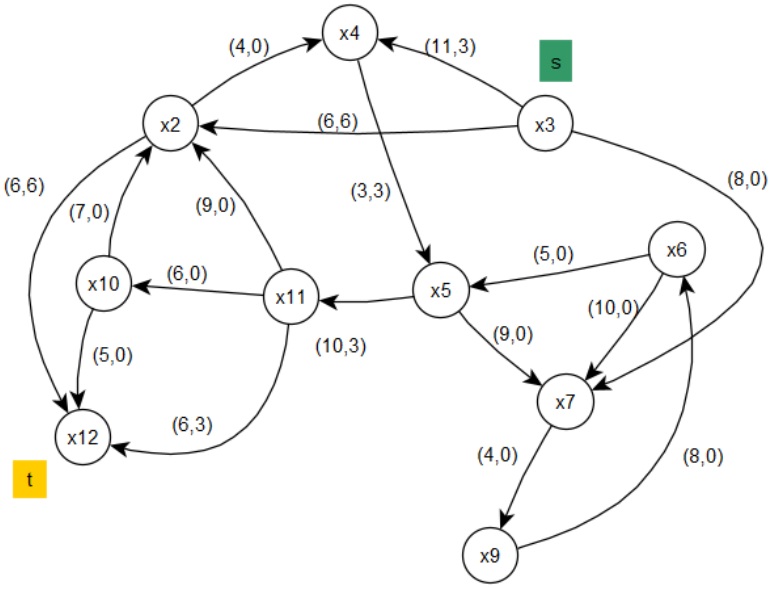
Підготовлений граф:

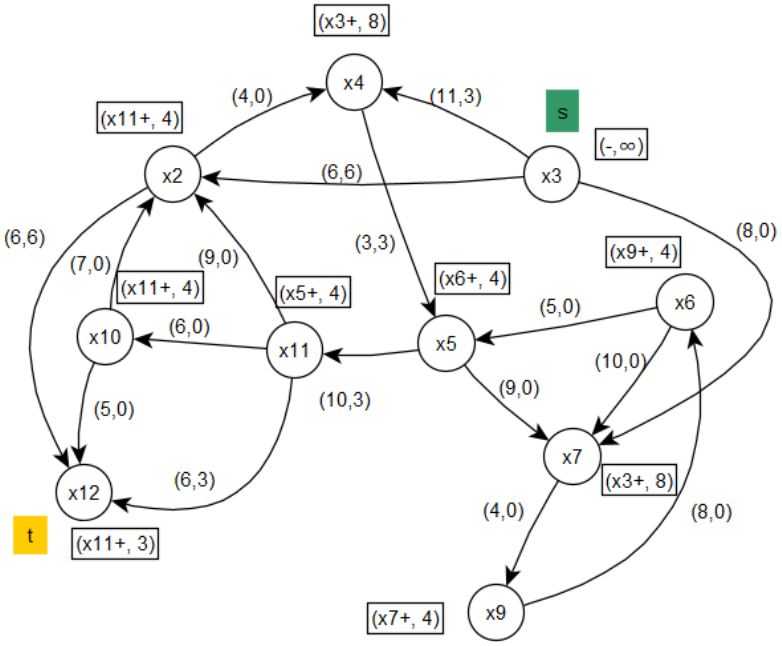


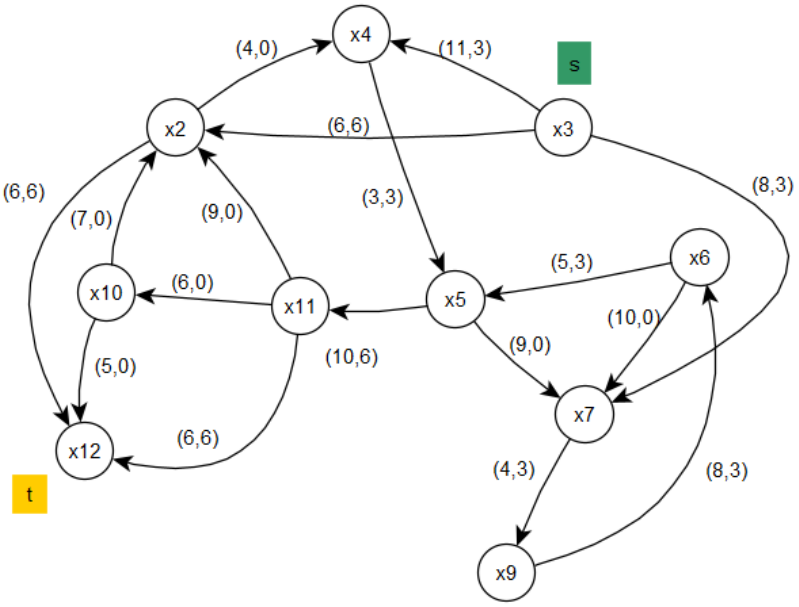
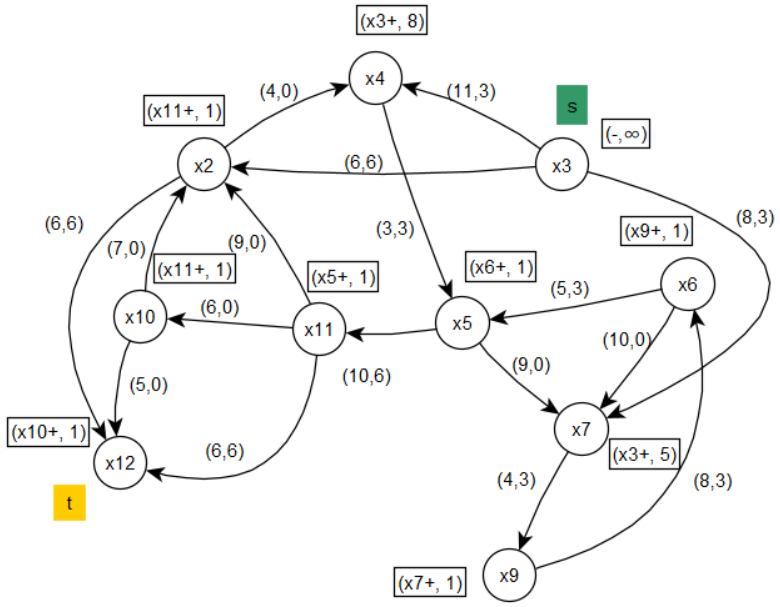
Крок 1	Етап 1. Позначення вершин. Усі вершини вважаємо непозначеними й неопрацьованими. Вершині-джерелу s – x3 ставимо позначку (–, ∞). Вершина x3 стає позначеною й неопрацьованою

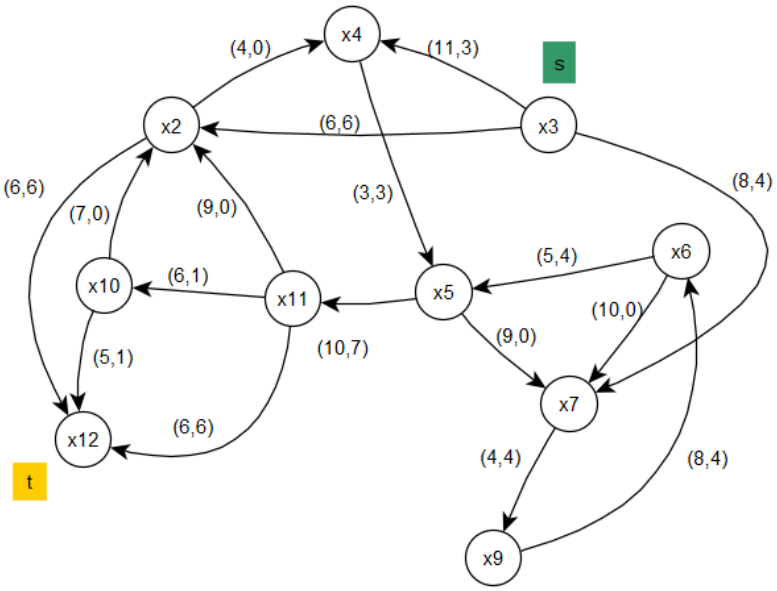
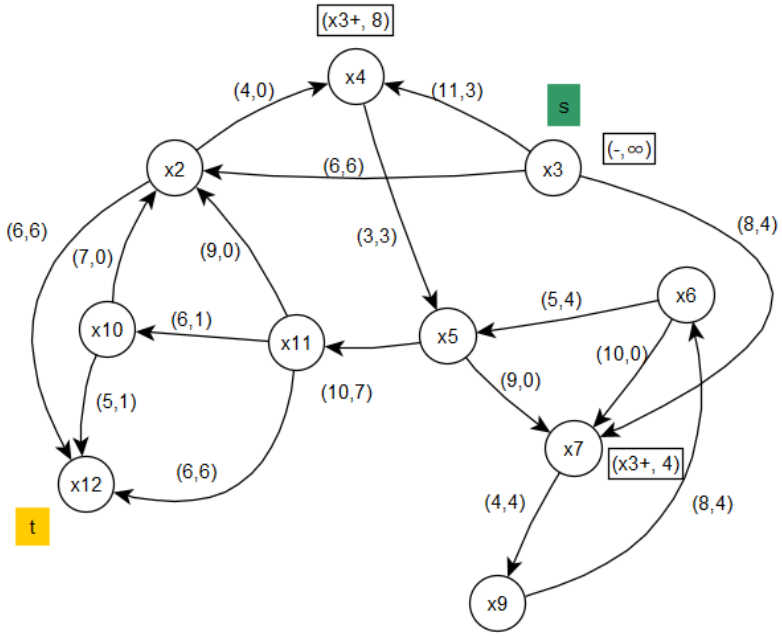


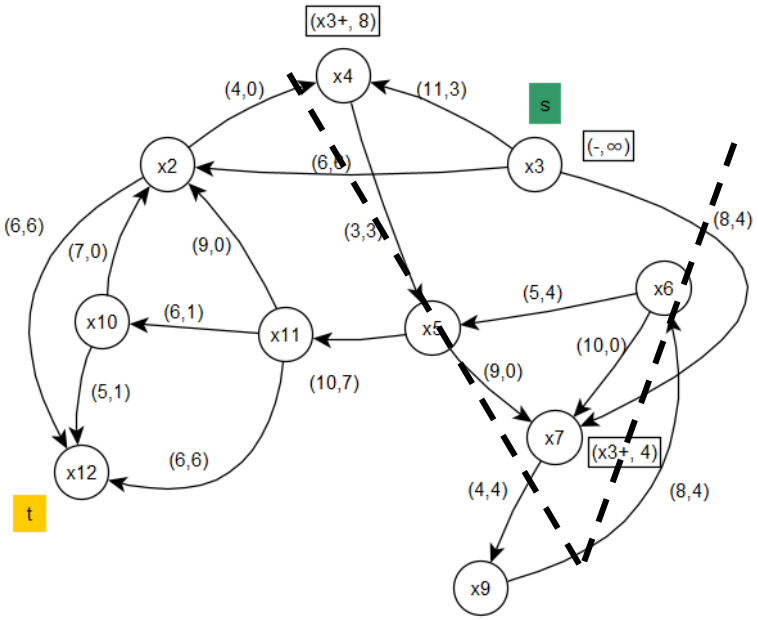
	Етап 2. Зміна потоку. Стік одержав позначку із значенням e = 3, отже, потоки в гілках побудованого ланцюга будуть збільшуватися на величину 3. Зміна потоку завжди починається зі стоку 
Крок 2	Етап 1. Позначення вершин 

	Етап 2. Зміна потоку. Стік одержав позначку із значенням e=6, отже, потоки в гілках побудованого ланцюга будуть збільшуватися на величину 6. Зміна потоку завжди починається зі стоку 

Крок 3	Етап 1. Позначення вершин 

	Етап 2. Зміна потоку. Стік одержав позначку із значенням e=3, отже, потоки в гілках побудованого ланцюга будуть збільшуватися на величину 3. Зміна потоку завжди починається зі стоку 
Крок 4	Етап 1. Позначення вершин 

	Етап 2. Зміна потоку. Стік одержав позначку із значенням e = 1, отже, потоки в гілках побудованого ланцюга будуть збільшуватися на величину 1. Зміна потоку завжди починається зі стоку 
Крок 5	Етап 1. Позначення вершин 
	На цей момент тут немає жодної позначеної неопрацьованої вершини, причому вершина-стік також не була позначена. Це означає, що наявний наразі потік є максимальним. У процесі виконання кроку 9 були позначені вершини x3, x4 і x7. Ці вершини утворюють множину X1 (ті, що залишились – множину X2) і визначають у такий спосіб мінімальний розріз графа G'



До розрізу входять ребра V22 (від x4), V5(від x3) і V11(від x7), причому отримана величина потоку через них збігається із їх пропускною здатністю. Це означає, що розріз дійсно є мінімальним, а сумарний потік через ці ребра – максимальним і дорівнює F = f5+f22+f11 = 3+6+4 = 13. Отже, через мережу, представлену графом G', від джерела s = x3 до стоку t = x12 може проходити потік з максимальним значенням 13. Алгоритм завершений

Індивідуальне творче завдання

Останній тематичний модуль вивчення навчальної дисципліни «Дискретна математика для комп'ютерних наук» присвячений розділу під назвою «Теорія графів». Хоча в індивідуальних роботах здобувачі освіти виконують алгоритми аналізу графів вручну, в реальному житті подібні алгоритми потребують багато зусиль і виконуються за допомогою комп'ютерів. Використання комп'ютерів під час обробки графів можливе у два способи:

1. Якщо людина має знання і досвід у програмуванні, вона може самостійно написати програмну реалізацію того чи іншого алгоритму.

2. Якщо досвід у програмуванні не надто великий, краще скористатись одним із готових програмних продуктів математичного спрямування, що підтримують ті чи інші алгоритми аналізу графів. Цей спосіб буде корисний і тим людям, які пишуть програми власноруч – завжди можна порівняти результат обробки графу за допомогою своєї та сторонньої програм.

Наразі у світі існує багато програм для роботи з графуми, які не вимагають знань програмування або вимагають їх на мінімальному рівні. До таких продуктів належать, наприклад, пакети Maple, MathCAD, Mathematica, Mathlab. Також існує багато сайтів, що надають можливість обробки графів онлайн – англомовні та україномовні. Достатньо задати в пошуку Гуглу назву потрібного алгоритму обробки графів, наприклад «Dijkstra online» або «Дейкстра онлайн». Знайдені браузерні програми дають змогу вручну побудувати невеликий граф та візуалізувати покрокове виконання того чи іншого алгоритму.

Окремо варто виділити клас мобільних додатків, які можна знайти, наприклад, у «Google Play» під операційну систему Android. Не треба очікувати, що ці додатки будуть україномовні, але англомовна термінологія в області теорії графів нескладна, і розібратися з цими програмами зазвичай не надто важко. Використати маленький додаток, написаний для вирішення якоїсь однієї задачі, зазвичай простіше, ніж розбиратись із великими математичними пакетами типу Maple. З іншого боку, маленьким «саморобним» програмам не варто довіряти, коли розв'язуєш задачу, від якої залежать серйозні речі: чи то фінансовий стан, чи то успішність, чи то життя реальних людей.

Суть індивідуального творчого завдання полягає в тому, щоб здобувач освіти самостійно розібрався з якимись готовими програмними продуктами з оброки й аналізу графів, спробував виконати завдання зі своїх індивідуальних робіт за допомогою цих продуктів та порівняв результати роботи програм із результатами, отриманими власноруч під час виконання індивідуальних робіт.

Завдання

Індивідуальне творче завдання не має окремих варіантів завдань. Воно однакове для всіх здобувачів освіти. Завдання полягає в такому:

Пункт 1. Серед завдань, які виконувались в індивідуальних роботах № 4 та № 5, треба обрати будь-які три завдання, одне з яких міститься в індивідуальній роботі № 4 (або хвильовий алгоритм, або алгоритм Дейкстри, або алгоритм Белмана–Форда), друге і третє – в індивідуальній роботі № 5 (будь-які два із наступних: алгоритм Прима, алгоритм Краскала, алгоритм Фльорі, алгоритм Робертса–Флореса).

Отже, треба обрати три якісь завдання. Наприклад, алгоритм Дейкстри з індивідуальної роботи № 4, алгоритм Краскала з індивідуальної роботи № 5, алгоритм Робертса–Флореса з індивідуальної роботи № 5.

Пункт 2. Знайти в мережі Інтернет і обрати програми для розв'язання задач, обраних у пункті 1. Це може бути або одна програма (якщо вона вміє вирішувати усі обрані задачі), або різні програми – стаціонарні, браузерні або мобільні. Важливо, щоб програми були здатні видавати не тільки кінцевий результат розв'язання задачі, але й обов'язково – окремі кроки виконання алгоритму.

Насправді можна спочатку виконати пункт 2, а потім – пункт 1. Спочатку знайти робочі і зрозумілі програми, потім подивитись, що вони вміють, і на основі цього обрати задачі до розв'язання. Тобто можна «підігнати задачі під програми». Хоча це й суперечить порядку дій у реальному житті, з навчальною метою це припустимо.

Під час вибору програм не треба інсталювати їх «піратські» версії. Зазвичай кожна програма або математичний пакет мають «пробний» режим роботи, якого достатньо для розв'язання поставлених задач. Браузерні програми зазвичай безкоштовні, але можуть вимагати стандартної процедури реєстрації за поштовою адресою. У цьому разі бажано про всяк випадок не використовувати власну корпоративну адресу Університету, а завести будь-яку іншу анонімну адресу на безкоштовному поштовому сервісі.

Пункт 3. Взяти свій підписаний граф та виконати кожну із задач, обраних у пункті 1, у програмі, обраній у пункті 2. Переконатись, що результат роботи загалом збігається з результатом відповідної індивідуальної роботи. Якщо, наприклад, розв'язувалась задача пошуку мінімального остовного дерева, знайдений програмою остов може відрізнятись від знайденого вручну, але його результуюча вага має бути однаковою.

Пункт 4. Зробити звіт з індивідуального творчого завдання. Основу звіту мають становити окремі знімки екрана, до кожного з яких додається невеличкий коментар.

Наприклад:

1) Запускаємо програму:

< знімок вікна запущеної програми >

2) Створюємо новий документ:

< знімок вікна програми з новим створеним документом >

3) Створюємо власний граф:

< знімок вікна програми зі вже створеним графом >

4) Запускаємо функцію пошуку найкоротшого шляху методом Дейкстри:

< знімок вікна програми, де показаний запуск функції >

5) Покрокове виконання програми:

< знімок вікна програми після кроку 1 >

< знімок вікна програми після кроку 2 >

...

< знімок вікна програми після кроку N >

6) Кінцевий результат:

< знімок вікна програми з кінцевим результатом >

7) Кінцевий результат, взятий із відповідної індивідуальної роботи:

< рисунок із кінцевим результатом із індивідуальної роботи >

8) Висновки про те, що кінцеві результати загалом збігаються.

Звіт з індивідуального творчого завдання повинен містити такі складники:

1. Титульний аркуш.

2. Лист із підписаним графом (копія).

3. Короткий опис завдань, обраних у пункті 1. Перелік програм або вебсайтів, обраних у пункті 2.

4. Опис процесу покрокового виконання кожного з трьох завдань зі знімками екрана.

5. Висновки щодо правильності виконання завдань.

Отже, під час виконання індивідуального творчого завдання здобувач вищої освіти фактично нічого не повинен робити. Йдеться про те, що не треба розв’язувати ніяких задач із теорії графів – їх буде вирішувати програма. Треба лише правильно скористатись готовими наявними програмами та зробити звіт про порядок використання цих програм на прикладі власного графу.

Виконавши завдання, здобувач освіти отримає такі корисні знання:

– дізнається про наявні програмні продукти, орієнтовані на роботу з графами;

– навчиться використовувати ці програмні продукти на прикладі конкретних задач;

– перевірить власні результати, отримані в індивідуальних роботах.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

Основна література

1. Матвієнко М. П. Дискретна математика: навчальний посібник. Київ: Ліра-К, 2014. 348 с.

2. Базилевич Л. Дискретна математика у прикладах і задачах: підручник. Львів: Видавець І. Е. Чижиков, 2013. 487 с.

3. Борисенко О. А. Дискретна математика: Підручник. Суми: ВТД «Університетська книга», 2007. 255 с.

4. Журавчак Л. М. Дискретна математика для програмістів. Львів: Видавництво Львівської політехніки, 2019. 420 с.

Допоміжна література

1. Кривий С. Л. Збірник задач з дискретної математики. Київ: Букрек, 2018. 456 с.

2. Дискретна математика. Практикум: навчальний посібник для студентів базового напряму 6.020305 – «Філологія» спеціальності «Прикладна лінгвістика» / О. С. Манзій, І. Є. Тесак, І. І. Кавалець, Н. В. Чарковська. Львів: Видавництво Національного університету «Львівська політехніка», 2016. 224 с.

Інформаційні ресурси в мережі Інтернет

1. Кузьменко І. М. Теорія графів: навч. посіб. для здобувачів ступеня бакалавра за освітньою програмою «Комп'ютерний моніторинг та геометричне моделювання процесів і систем» спеціальності 122 Комп'ютерні науки. КПІ ім. Ігоря Сікорського. Електронні текстові дані (1 файл: 1,7 Мбайт). Київ: КПІ ім. Ігоря Сікорського, 2020. 71 с.

2. Матеріали курсу «Дискретна математика». URL: https://github.com/olexiim/dm

3. Discrete Mathematics Tutorial. URL: https://www.tutorialspoint.com/discrete_mathematics/

Навчально-методичне видання

Бабаков Роман Маркович

ДИСКРЕТНА МАТЕМАТИКА ДЛЯ КОМП'ЮТЕРНИХ НАУК

Методичні рекомендації до виконання індивідуальних робіт
для здобувачів вищої освіти ОС «Бакалавр» денної форми здобуття освіти
спеціальності F3 Комп'ютерні науки

Редактор О. А. Солдатова
Технічний редактор О. К. Гомон

Підписано до друку 20.11.2025.
Формат 60×84/16. Папір офсетний.
Друк – цифровий. Умовн. друк. арк. 3,66.
Тираж 30. Зам. 158.

Донецький національний університет імені Василя Стуса
21021, м. Вінниця, 600-річчя, 21.
Свідоцтво про внесення суб'єкта видавничої справи
до Державного реєстру
серія ДК № 5945 від 15.01.2018