

Лабораторна робота №7.

Алгоритми знаходження остовного дерева мінімальної ваги.

Мета роботи: Вивчення алгоритмів побудови остовного дерева мінімальної ваги і реалізація алгоритмів на мові високого рівня.

Задача роботи: Дослідити задачу знаходження остовного дерева мінімальної ваги, використовуючи алгоритм Прима та алгоритм Крускала .

Теоретичні відомості.

Алгоритм Прима.

Для заданого графа G (рис.1) побудуємо остовне дерево мінімальної ваги, використовуючи алгоритм Прима.

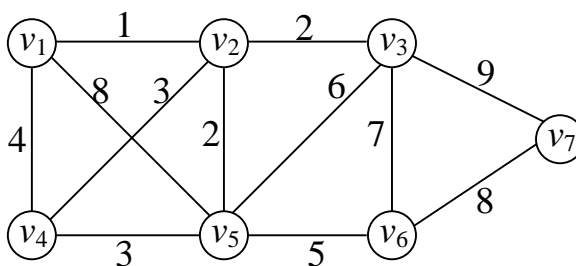


Рис. 1. Неорієнтований граф G

Побудуємо матрицю суміжності ваг:

$$\begin{array}{c} v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6 \quad v_7 \\ \begin{array}{c} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{array} \left(\begin{array}{ccccccc} \infty & 1 & \infty & 4 & 8 & \infty & \infty \\ 1 & \infty & 2 & 3 & 2 & \infty & \infty \\ \infty & 2 & \infty & \infty & 6 & 7 & 9 \\ 4 & 3 & \infty & \infty & 3 & \infty & \infty \\ 8 & 2 & 6 & 3 & \infty & 5 & \infty \\ \infty & \infty & 7 & \infty & 5 & \infty & 8 \\ \infty & \infty & 9 & \infty & \infty & 8 & \infty \end{array} \right)$$

За початкову обираємо довільну вершину, нехай це буде вершина v_1 , і для неї шукаємо найближчого сусіда, тобто вершину відстань до якої найменша:

	v_2	v_3	v_4	v_5	v_6	v_7
v_1	1	∞	4	8	∞	∞

Це буде вершина v_2 . Отримали перший фрагмент (рис.2 а). Вершина v_2 вилучається з рядка і переноситься у стовпчик.

Розширюємо фрагмент, переглядаючи всі можливі відстані від вершин, що знаходяться у стовпчику до ще неприєднаних вершин, які знаходяться в рядочку:

	v_3	v_4	v_5	v_6	v_7
v_1	∞	4	8	∞	∞
v_2	2	3	2	∞	∞

Існує дві вершини відстань до яких найменша і однакова. Це вершини v_3 та v_5 . Обираємо будь-яку з них, нехай це буде вершина v_3 . Отримуємо новий фрагмент (рис. 2 б).

Розширюємо фрагмент:

	v_4	v_5	v_6	v_7
v_1	4	8	∞	∞
v_2	3	2	∞	∞
v_3	∞	6	7	9

Найменша відстань від вершини v_2 до вершини v_5 , тому фрагмент розширюємо на вершину v_5 (рис. 2 в).

Розширюємо фрагмент:

	v_4	v_6	v_7
v_1	4	∞	∞
v_2	3	∞	∞
v_3	∞	7	9
v_5	3	5	∞

Існує дві вершини відстань до яких найменша і однакова. Це вершини v_2 та v_5 . Обираємо будь-яку з них, нехай це буде вершина v_2 . Отримуємо новий фрагмент (рис. 2 г).

Розширюємо фрагмент:

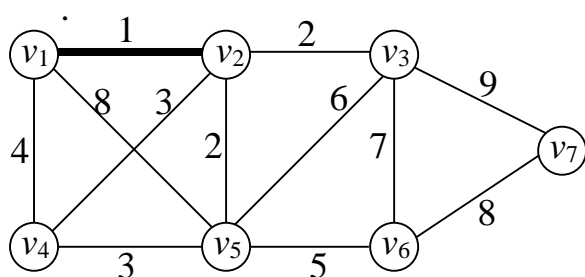
	v_6	v_7
v_1	∞	∞
v_2	∞	∞
v_3	7	9
v_4	∞	∞
v_5	5	∞

Найменша відстань від вершини v_5 до вершини v_6 , тому фрагмент розширюємо на вершину v_6 (рис. 2 д).

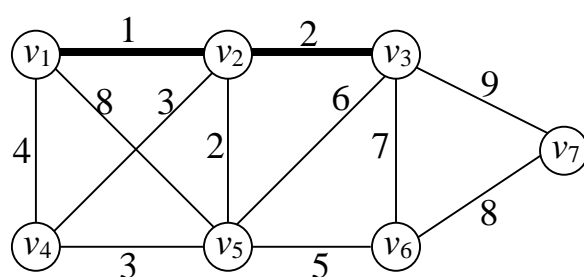
Розширюємо фрагмент:

	v_7
v_1	∞
v_2	∞
v_3	9
v_4	∞
v_5	∞
v_6	8

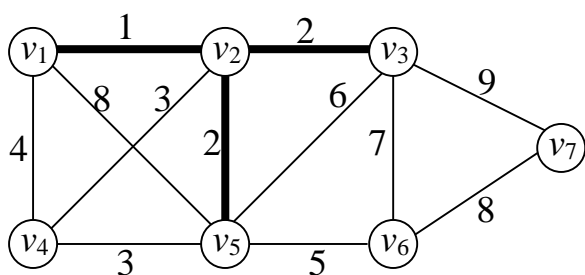
Остання неприєднана вершина v_7 приєднується до вершини v_6 . Оптимізуємо мінімальне остовне дерево T (рис. 2 е) з вагою $w = 1+2+2+3+5+8 = 21$.



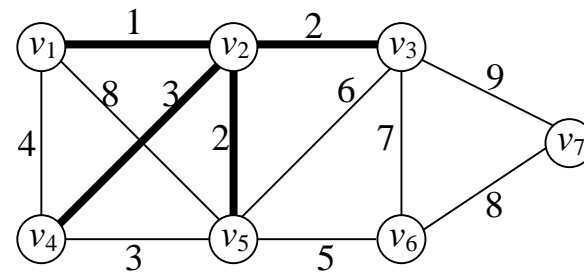
а)



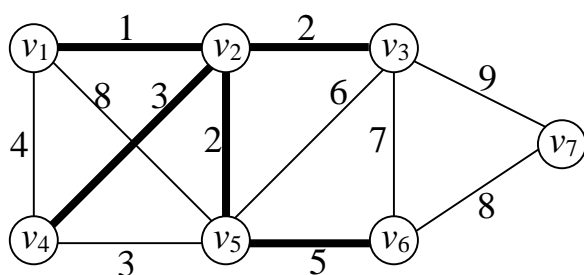
б)



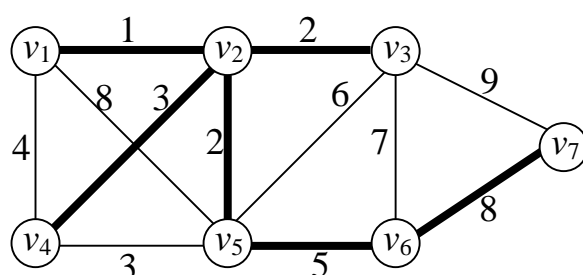
в)



г)



д)



е)

Рис. 2. Реалізація алгоритму Прима

Алгоритм Крускала.

Для заданого графа G (рис. 1) побудуємо остовне дерево мінімальної ваги, використовуючи алгоритм Крускала.

Спочатку впорядковуємо ребра (табл.1, стовпчик 1). Виконуємо розбиття множини вершин $\rho_0 = \{\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\}, \{v_7\}\}$. Переглядаємо кожне ребро в порядку зростання, якщо інцидентні йому вершини знаходяться в різних множинах розбиття, ребро додаємо до остовного дерева мінімальної ваги, а множини, в яких знаходяться ці вершини, об'єднуємо в одну. Процес вибору ребер наведено у табл.1.

Таблиця 1

Ребра впорядковані за зростанням	Розбиття множини вершин	Вибір ребра у мінімальний остов T
$e_1(v_1, v_2) = 1$	$\rho_1 = \{\{v_1, v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\}, \{v_7\}\}$	$E_T = \{e_1\}$
$e_2(v_2, v_3) = 2$	$\rho_2 = \{\{v_1, v_2, v_3\}, \{v_4\}, \{v_5\}, \{v_6\}, \{v_7\}\}$	$E_T = \{e_1, e_2\}$
$e_3(v_5, v_6) = 2$	$\rho_3 = \{\{v_1, v_2, v_3\}, \{v_4\}, \{v_5, v_6\}, \{v_7\}\}$	$E_T = \{e_1, e_2, e_3\}$
$e_4(v_2, v_4) = 3$	$\rho_4 = \{\{v_1, v_2, v_3, v_4\}, \{v_5, v_6\}, \{v_7\}\}$	$E_T = \{e_1, e_2, e_3, e_4\}$
$e_5(v_4, v_5) = 3$	$\rho_5 = \{\{v_1, v_2, v_3, v_4, v_5, v_6\}, \{v_7\}\}$	$E_T = \{e_1, e_2, e_3, e_4, e_5\}$
$e_6(v_1, v_4) = 4$	ρ_5	$e_6 \notin E_T$
$e_7(v_2, v_5) = 5$	ρ_5	$e_7 \notin E_T$
$e_8(v_3, v_5) = 6$	ρ_5	$e_8 \notin E_T$
$e_9(v_3, v_6) = 7$	ρ_5	$e_9 \notin E_T$
$e_{10}(v_1, v_5) = 8$	ρ_5	$e_{10} \notin E_T$
$e_{11}(v_6, v_7) = 8$	$\rho_6 = \{\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}\}$	$E_T = \{e_1, e_2, e_3, e_4, e_5, e_{11}\}$
$e_{12}(v_3, v_7) = 9$	-	-

При приєднанні ребра e_{11} робота алгоритму закінчується, так як вже приєднано $n-1 = 7-1=6$ ребер і всі підмножини розбиття об'єдналися в одну $\rho_6 = \{\{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}\}$. Остовне дерево мінімальної ваги T утворюють ребра $e_1, e_2, e_3, e_4, e_5, e_{11}$ (рис. 3).

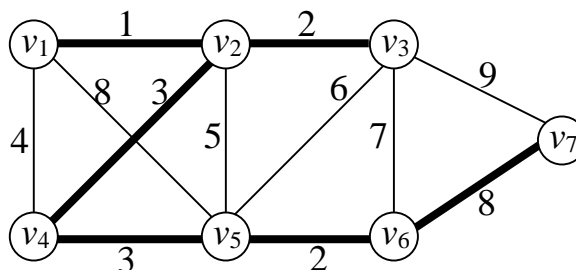


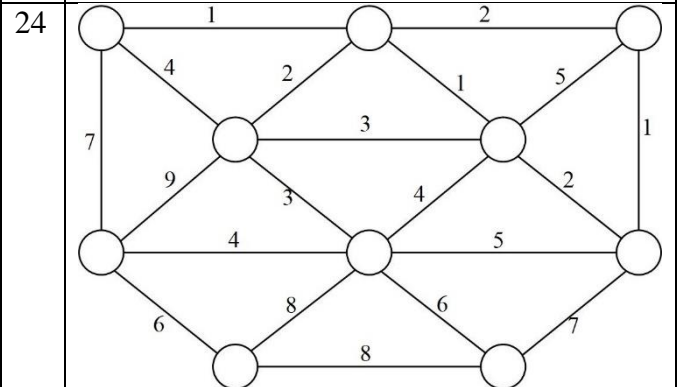
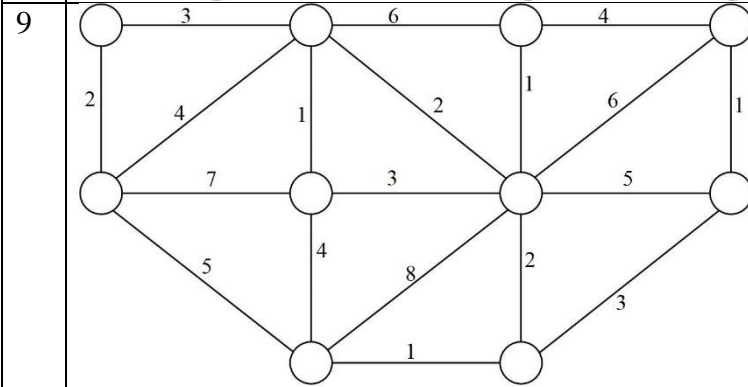
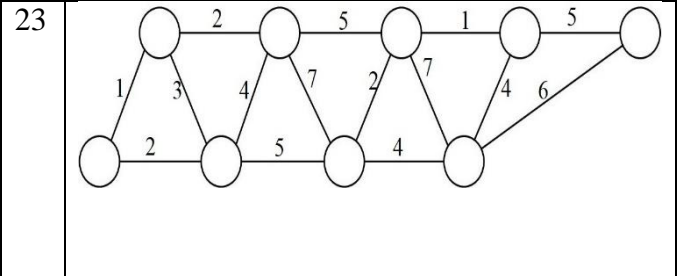
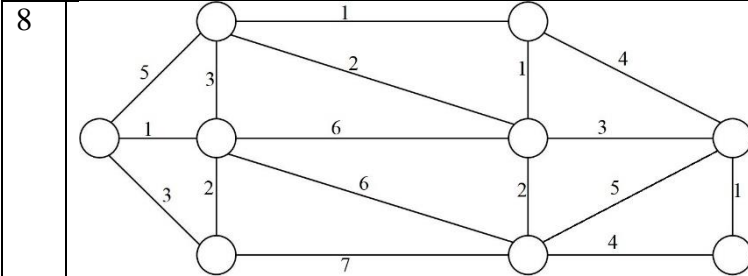
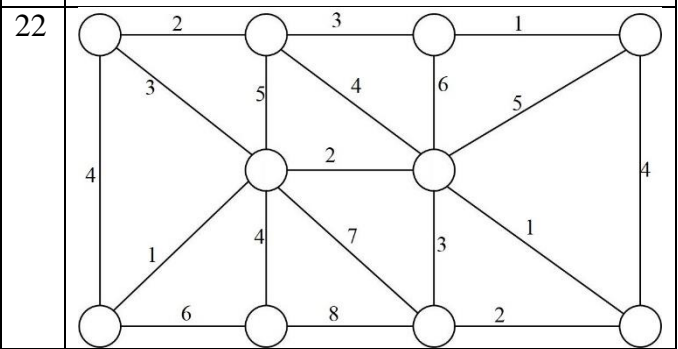
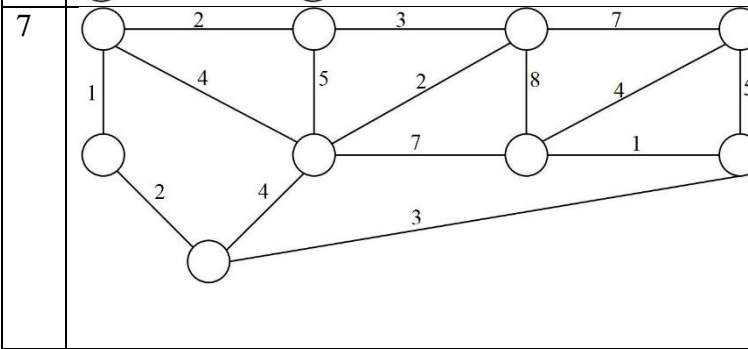
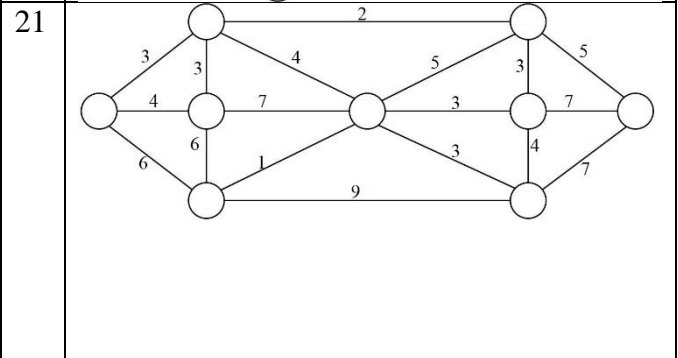
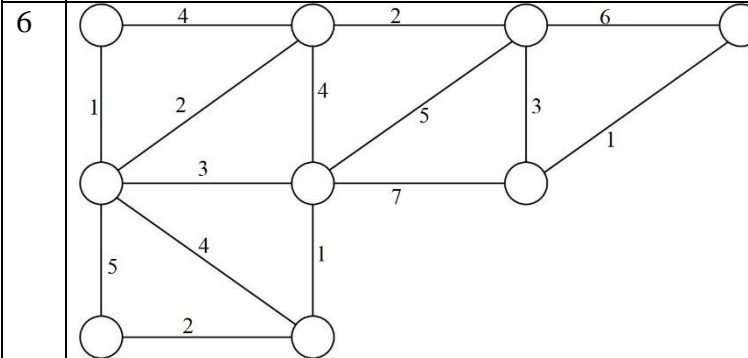
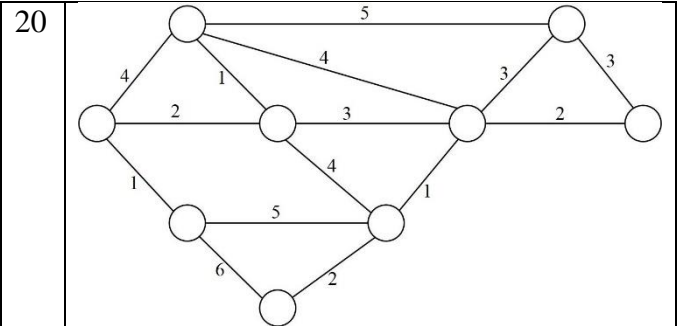
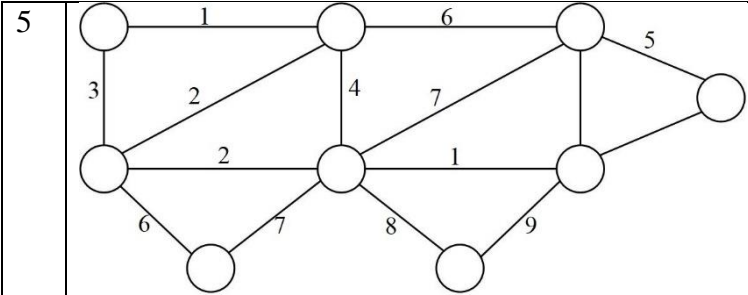
Рис. 3. Алгоритм Крускала

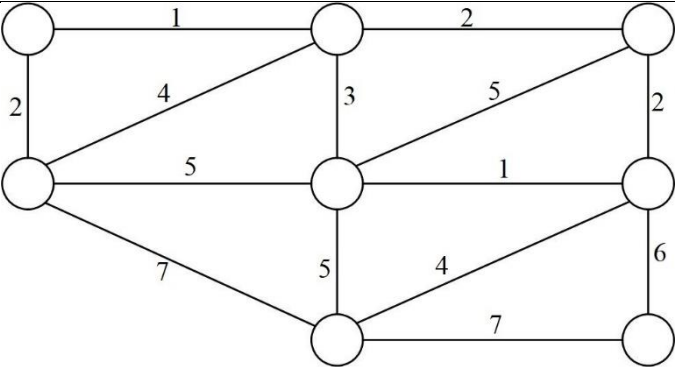
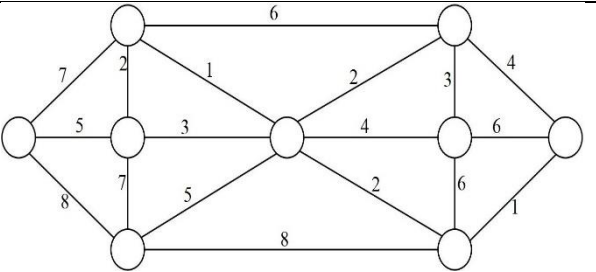
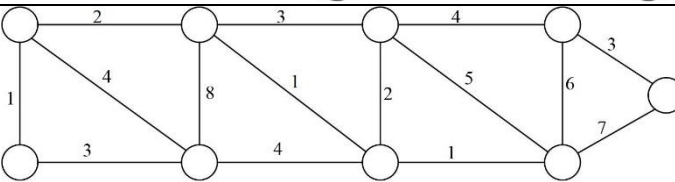
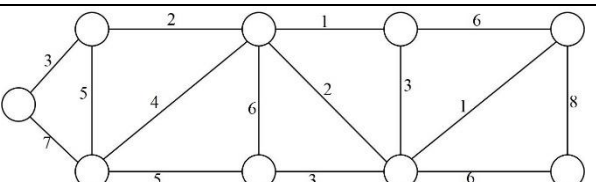
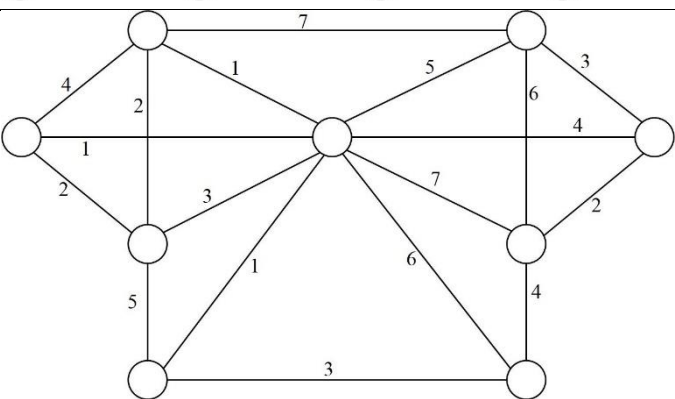
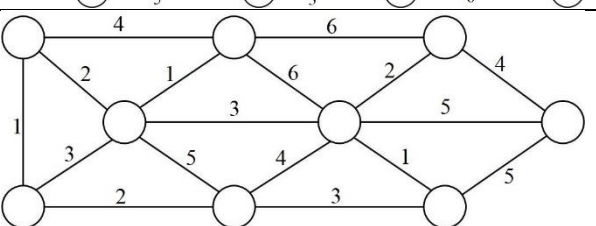
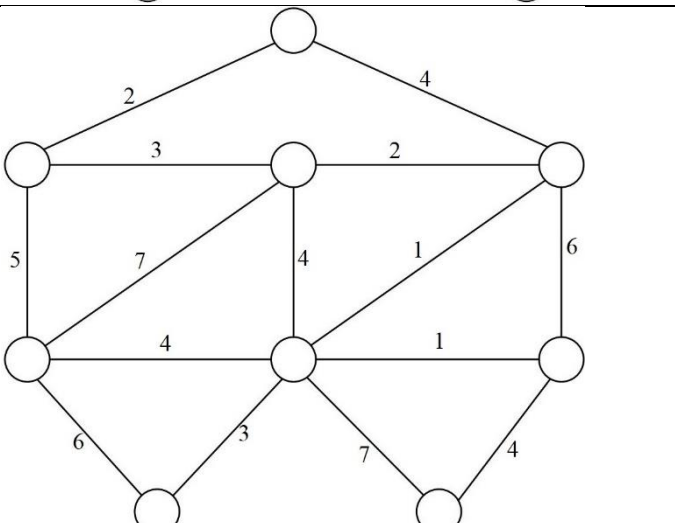
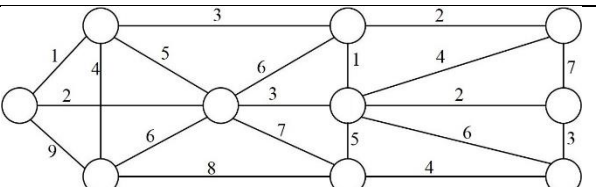
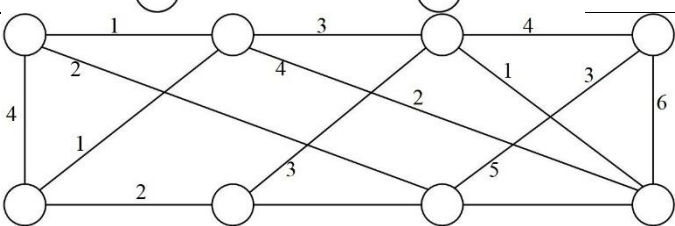
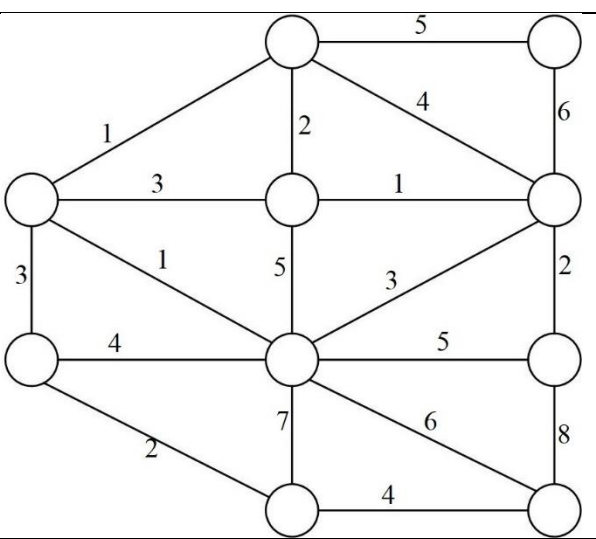
Завдання.

1. Нерієнтований граф $G(V,E)$ задано в табл.
2. Для заданого графа побудувати остовне дерево мінімальної вартості:
 - а) за допомогою алгоритму Прима;
 - б) за допомогою алгоритму Крускала.
3. Скласти схему алгоритму та написати програми, що реалізують ці алгоритми (парні варіанти за списком – Прима, непарні варіанти - Крускала).
4. Написати процедуру обчислення вартості побудованого остовного дерева.

Варіанти графів

1		16	
2		17	
3		18	
4		19	



10		25	
11		26	
12		27	
13		28	
14		29	

15		30
----	--	----

Контрольні запитання.

1. Дайте визначення: граф, орієнтований граф, степінь вершини, простий, псевдо, дводольний, ізоморфний, граф, мультиграф, остовний підграф, зважений граф.
2. Способи задання графів.
3. Постановка задачі пошуку остовного підграфу мінімальної ваги.
4. Чи будуть коректно працювати алгоритми Крускала та Прима з графом, у якого є ребра з від'ємною вагою?
5. Дайте визначення: фрагмент, ізольований фрагмент, ізольована вершина.
6. В чому полягає ідея алгоритмів Крускала та Прима?