# Лабораторна робота №4

**Тема: Основні операції над графами. Знаходження остова мінімальної ваги за алгоритмом Прима-Краскала.**

Варіант №9

***Завдання 1.*** Виконати наступні операції над графами:

1) знайти доповнення до першого графу

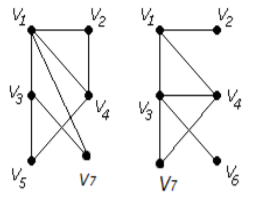
2) об’єднання графів

3) кільцеву суму G1 та G2 (G1+G2)

4) розщепити вершину у другому графі

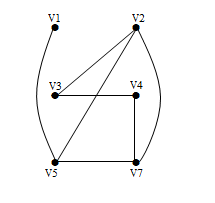
5) виділити підграф А, що складається з 3-х вершин в G1 і знайти стягнення А в G1 (G1\ A)

6) добуток графів.

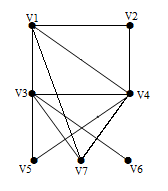


1. Доповнення:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** | **7** |
| **1** | 0 | 1 | 1 | 1 | 0 | 1 |
| **2** | 1 | 0 | 0 | 1 | 0 | 0 |
| **3** | 1 | 0 | 0 | 0 | 1 | 1 |
| **4** | 1 | 1 | 0 | 0 | 1 | 0 |
| **5** | 0 | 0 | 1 | 1 | 0 | 0 |
| **7** | 1 | 0 | 1 | 0 | 0 | 0 |



1. Об’єднання:

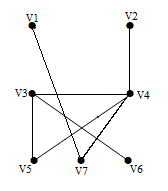


1. Кільцева сума G = G1 + G2

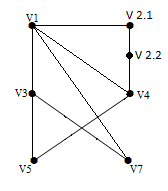
G1 = <V,E>: V={V1, V2, V3, V4,V5,V7}, E={(V1,V2), (V1,V3), (V1,V4), (V1,V7), (V2,V4), (V3,V5), (V3,V7), (V4,V5)}

G2 = <V,E>: V={V1, V2, V3, V4,V5,V6}, E={(V1,V2), (V1,V3), (V1,V4), (V3,V4), (V3,V6), (V3,V7), (V4,V7)}

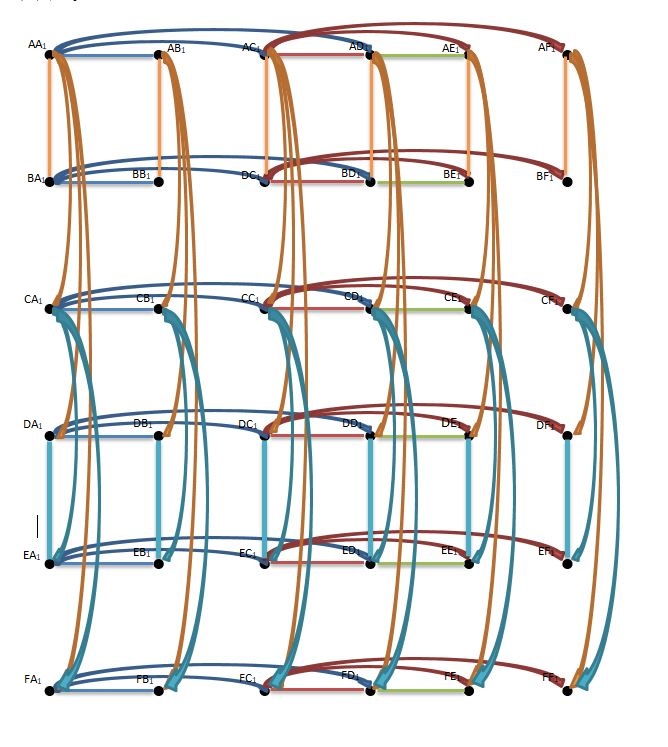
G = <V,E>: V={V1, V2, V3, V4,V5,V6,V7}, E={(V1,V7), (V2,V4), (V3,V5), (V4,V5), (V3,V4), (V3,V6), (V4, V7)}



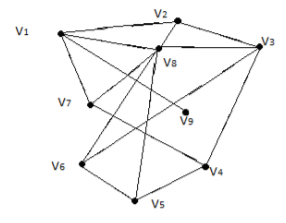
1. Розщеплення:



1. Добуток:



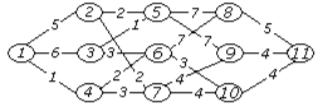
**Завдання 2:** Знайти таблицю суміжності та діаметр графа.



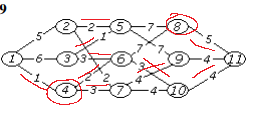
V9-V1-V7-V4-V5-V6-V3-V2-V8

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| **1** | **0** | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| **2** | 1 | **0** | 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| **3** | 0 | 1 | **0** | 1 | 0 | 1 | 0 | 1 | 0 |
| **4** | 0 | 0 | 1 | **0** | 1 | 0 | 1 | 0 | 0 |
| **5** | 0 | 0 | 0 | 1 | **0** | 1 | 0 | 1 | 0 |
| **6** | 0 | 0 | 1 | 0 | 1 | **0** | 0 | 0 | 0 |
| **7** | 0 | 0 | 0 | 1 | 0 | 0 | **0** | 1 | 0 |
| **8** | 1 | 1 | 1 | 0 | 1 | 1 | 1 | **0** | 0 |
| **9** | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **0** |

**Завдання 3:** Знайти двома методами (Краскала і Прима) мінімальне остове дерево графа.



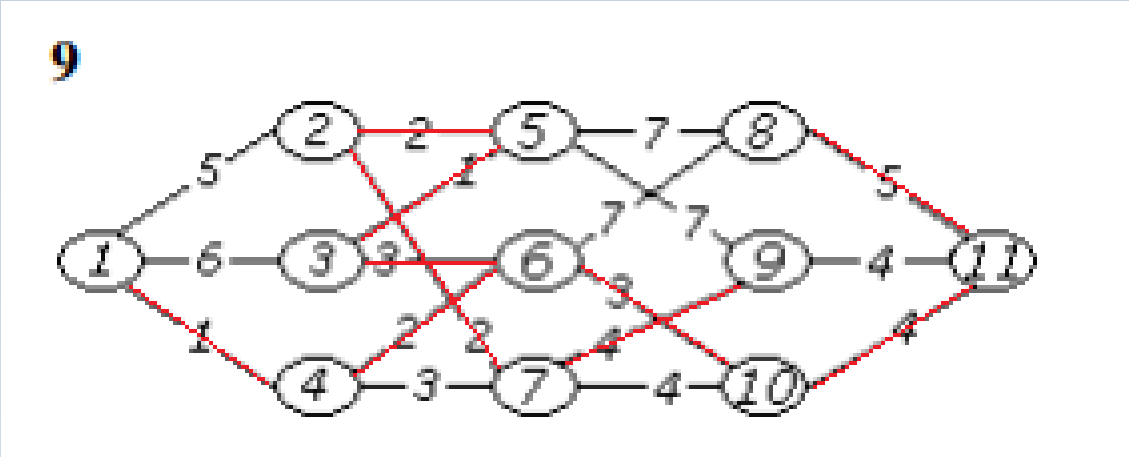
1. *Метод Прима:*

**

*Мінімальне остове дерево = 27*

*2) Метод Краскаля:*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Ребро** | 1,4 | 3,5 | 2,5 | 2,7 | 4,6 | 3,6 | 4,7 | 6,10 | 7,9 | 9,11 | 10,11 | 1,2 | 8,11 |
| **Вага** | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 5 | 5 |
| **Ребро** | 1,3 | 5,8 | 5,9 | 6,8 |
| **Вага** | 6 | 7 | 7 | 7 |



*Мінімальне остове дерево = 27*

***Завдання 2:*** Написати програму, яка реалізує алгоритм знаходження остового дерева мінімальної ваги за алгоритмом Прима чи Краскала. Етапи розв'язання задачі виводити на екран. Протестувати розроблену програму на задачі 3 із завдання № 1.

*Лістинг коду:*

using System;

using System.Collections.Generic;

using System.Text;

namespace KMD4

{

class Program

{

static void Main(string[] args)

{

Console.OutputEncoding = Encoding.Unicode;

Console.InputEncoding = Encoding.Unicode;

int n, m;

Console.Write("Введіть кількість вершин:");

n = int.Parse(Console.ReadLine());

Console.WriteLine("----------------------------------");

Console.Write("Введіть кількість ребер:");

m = int.Parse(Console.ReadLine());

Console.WriteLine("----------------------------------");

int[,] matrix = new int[n, n];

Console.WriteLine("Введення здійснюйте в форматі: 1 вершина,2 верши-на,вага.");

for (int i = 1; i <= m; i++)

{

Console.Write($"Введіть вершини {i} ребра та його вагу: ");

string[] vrt = Console.ReadLine().Split(',');

int a = int.Parse(vrt[0]);

int b = int.Parse(vrt[1]);

matrix[a - 1, b - 1] = int.Parse(vrt[2]);

matrix[b - 1, a - 1] = int.Parse(vrt[2]);

}

Console.WriteLine("----------------------------------");

Console.WriteLine("Матриця суміжності графа:");

for (int i = 0; i < matrix.GetLength(0); i++)

{

for (int j = 0; j < matrix.GetLength(1); j++)

{

Console.Write($"{matrix[i, j],4}");

}

Console.WriteLine();

}

Console.WriteLine("----------------------------------");

PrimMethod(matrix, n);

void PrimMethod(int[,] mtrx, int n\_count)

{

List<int> vrt\_2 = new List<int>();

bool[] used = new bool[n\_count];

for (int i = 0; i < used.Length; i++)

used[i] = false;

used[0] = true;

int sum = 0;

int existVertex = 0, newVertex = 0;

for (int k = 0; k < mtrx.GetLength(0) - 1; k++)

{

int min = int.MaxValue;

for (int i = 0; i < mtrx.GetLength(0); i++)

{

if (!used[i])

continue;

for (int j = 0; j < mtrx.GetLength(1); j++)

{

if (mtrx[i, j] != 0 && !used[j])

vrt\_2.Add(j);

}

for (int j = 0; j < vrt\_2.Count; j++)

{

if (mtrx[i, vrt\_2[j]] < min)

{

min = mtrx[i, vrt\_2[j]];

existVertex = i;

newVertex = vrt\_2[j];

}

}

vrt\_2.Clear();

}

used[newVertex] = true;

sum += min;

Console.WriteLine($"{k + 1}. ({existVertex + 1},{newVertex + 1})");

}

Console.WriteLine("----------------------------------");

Console.WriteLine($"Мінімальне остове дерево (за методом Прима) = {sum}");

}

Console.ReadKey();

}

}

}

