**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Розрахункова робота**

з дисципліни

«Дискретна математика»

Виконав:

студент КН-111

Пучак Віталій

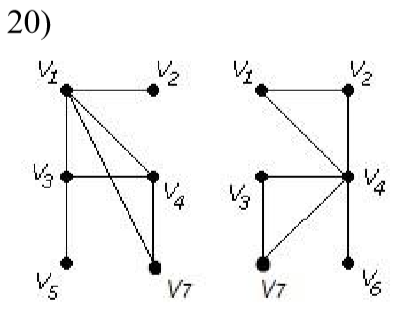
Викладач:

Мельникова Н.І

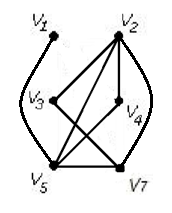
Львів – 2018 р.

**Варіант 20**

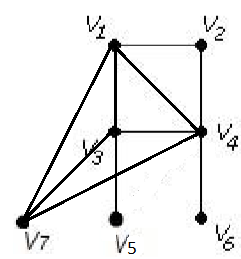
1. **Виконати наступні операції над графами:**



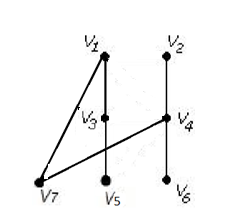
1. Знайти доповнення до першого графу.



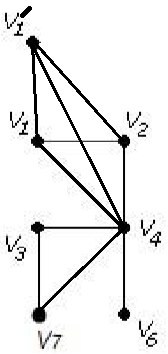
1. об’єднання графів.



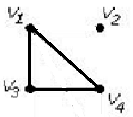
1. кільцеву суму G1 та G2 (G1+G2)



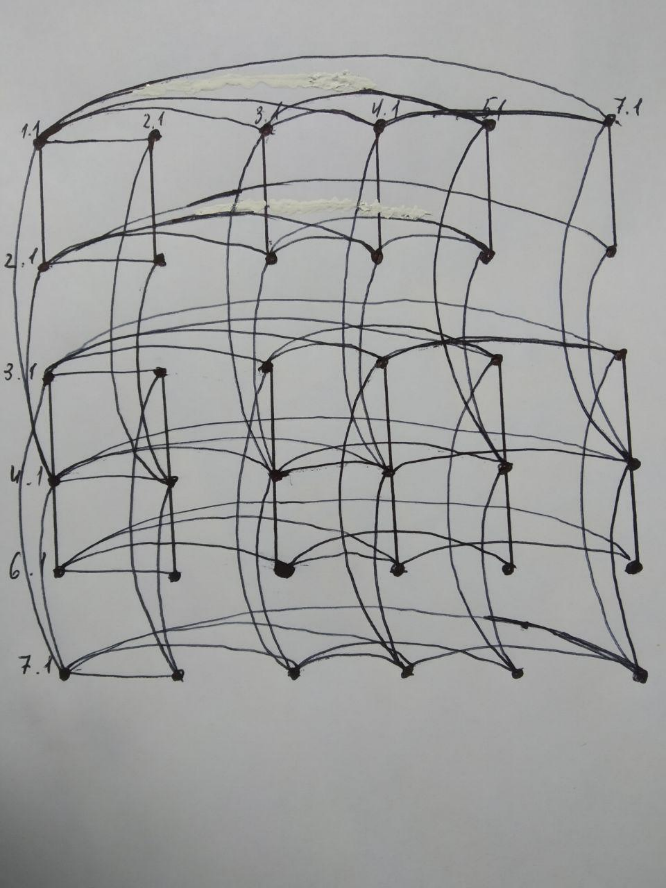
1. розщепити вершину у другому графі



1. виділити підграф А, що складається з 3-х вершин в G1



1. добуток графів



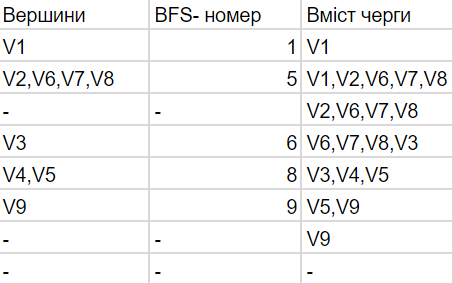
1. **Знайти таблицю суміжності графа.**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 |
| V1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| V2 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| V3 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| V4 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| V5 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| V6 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| V7 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| V8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| V9 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

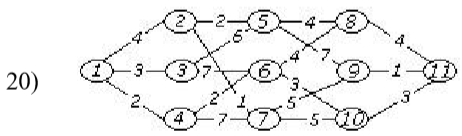
1. **Діаметр графа:**

Діаметр графа 3 (1->6->5->9) оскільки максимальний ексцентриситет графа це ексцентриситет вершин V2 і V9,який рівний 3.

1. **Обхід графа в шир:**



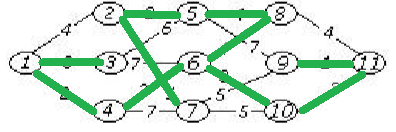
1. **Знайти двома методами (Краскала і Прима) мінімальне остове дерево графа.**

****

Краскала:

|  |  |  |
| --- | --- | --- |
| Звідки | Куди | Вага |
| 2 | 7 | 1 |
| 9 | 11 | 1 |
| 2 | 5 | 2 |
| 1 | 4 | 2 |
| 4 | 6 | 2 |
| 1 | 3 | 3 |
| 6 | 10 | 3 |
| 10 | 11 | 3 |
| 5 | 8 | 4 |
| 6 | 8 | 4 |

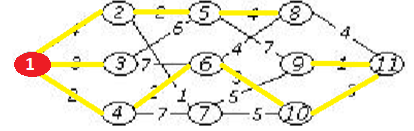
***Сума ваг 25***

******

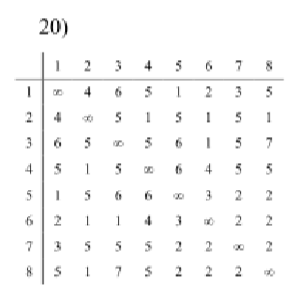
1. Прима

|  |  |  |
| --- | --- | --- |
| Звідки | Куди | Вага |
| 1 | 4 | 2 |
| 4 | 6 | 2 |
| 1 | 3 | 3 |
| 6 | 10 | 3 |
| 10 | 11 | 3 |
| 11 | 9 | 1 |
| 6 | 8 | 4 |
| 8 | 5 | 4 |
| 5 | 2 | 2 |
| 2 | 7 | 1 |

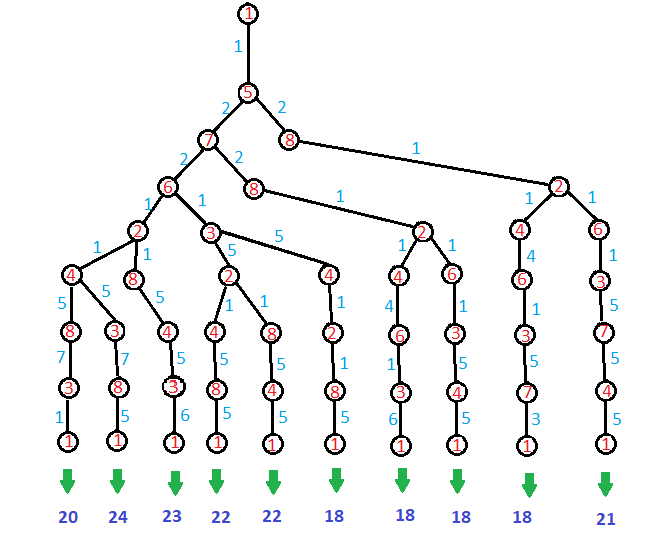
***Сума ваг 25***



1. **Задача комівояжера:**



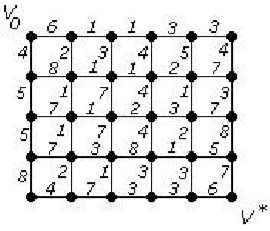
Принцип виконання: стартуємо наприклад із першої вершини та продовжуємо рухатися в ту, в яку відстань є найкоротшою, у нашому випадку це вершина 5 і відстань дорівнює 1. Тепер працюємо із вершиною 5, найкоротша відстань є до вершин 7 та 8 і дорівнює 2. Продовжуємо робити те саме із вершиною 7. Завдання комівояжера проглянути всі можливі випадки із усіх вершин та вибрати найкоротший. Складність цього завдання полягає в складній матриці суміжності, адже в ній дуже багато повторів, а отже і багато можливих ситуацій, які потрібно розглянути із однієї вершини.



Отже, із малюнка ми можемо побачити що найкоротший шлях дорівнює 18 і його можна здійснити чотирьома способами.

1. **За допомогою алгоритму Дейкстра знайти найкоротший шлях**

**у графі поміж парою вершин V0 і V\***



**Алгоритм Дейкстри**(знаходить найкоротший шлях від однієї вершини графа до всіх інших вершин):

1) Перший по порядку сусід (V0) вершини — 1-а вершина. Шлях до неї через 1-у вершину дорівнює найкоротшій відстані до 1-ї вершини + довжина дуги між V0 та 1-ю вершиною, тобто 0 + 6 = 6. Це менше поточного найкоротшого шляху до 1-ї вершини, тому найкоротший шлях до 1-ї вершини дорівнює 6.

2) Аналогічну операцію проробляємо з іншим сусідом V0 вершини — 6-ю вершиною (відстань до неї тепер = 4).

3) Всі сусіди вершини V0 перевірені. Поточна мінімальна відстань до вершини V0 вважається остаточною. Тому викреслимо її з графа, щоб відмітити цей факт.

4)Повертаємось до кроку 2. Знову знаходимо «найближчу» необроблену вершину. Це вершина 1 з поточною найкоротшою відстанню до неї = 6. І знову намагаємося зменшити відстань до всіх сусідів 1-ї вершини, намагаючись пройти в них через 1-у. Сусідами 1-ї вершини є V0, 2, 7.

5) Перший (по порядку) сусід вершини № 1 — V0 вершина. Але вона вже оброблена. Тому з V0 вершиною нічого не робимо.

6) Інший сусід вершини 1 — вершина 2. Якщо йти в неї через 1-у, то шлях буде = найкоротша відстань до 1-ї + відстань між 1-ю і 2-ю вершинами = 6 + 1 = 7. Оскільки 7 < ∞, встановлюємо відстань до вершини № 2 рівним 7.

7) Ще один сусід вершини 1 — вершина 7. Якщо йти в неї через 1-у, то шлях буде = 6 + 2 = 8. Оскільки 8 < ∞, встановлюємо відстань до вершини № 7 рівним 8.

8) Всі сусіди вершини 2 перевірені. Поточна мінімальна відстань до вершини

2 -3. Тому викреслимо її з графа, щоб відмітити цей факт.

(Проробляємо аналогічні кроки з ще не відвіданими вершинами.)

Алгоритм закінчує роботу, коли викреслені всі вершини.

Найкоротші шляхи:

0 - 1 = 6

0 - 2 = 7

0 - 3 = 8

0 - 4 = 11

0 - 5 = 14

0 - 6 = 4

0 - 7 = 8

0 - 8 = 9

0 - 9 = 10

0 - 10 = 12

0 - 11 = 18

0 - 12 = 9

0 - 13 = 9

0 - 14 = 10

0 - 15 = 12

0 - 16 = 13

0 - 17 = 20

0 - 18 = 14

0 - 19 = 10

0 - 20 = 13

0 - 21 = 16

0 - 22 = 15

0 - 23 = 20

0 - 24 = 16

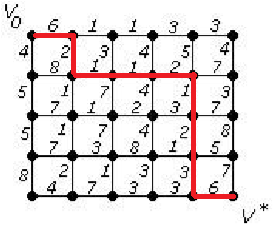
0 - 25 = 12

0 - 26 = 14

0 - 27 = 17

0 - 28 = 18

V0 - V\* = 24



Вага=24.

Шлях V -> V\* (0-29)

0-1-8-9-10-16-22-28-29

1. **Знайти ейлеровий цикл в ейлеровому графі двома методами**

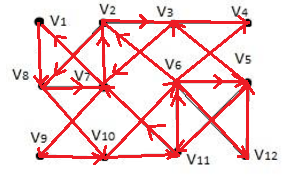
**а) Флері б) Елементарних циклів.**

*Алгоритм Флері*

Починаємо з деякої вершини V1, що належить графу і кожен раз викреслюємо пройдене ребро.

Не проходимо по ребру, якщо видалення цього ребра призводить до розбиття графа на дві зв'язні компоненти (не рахуючи ізольованих вершин), тобто необхідно перевіряти, чи є ребро мостом чи ні.

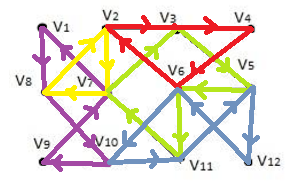
Отже, викреслюємо ребра так: 1- 8 -10 -11- 6 -12 -5- 11- 7- 9- 10- 6- 5- 3- 4- 6- 2- 3- 7- 2- 8- 7- 1



Метод елементарних циклів

Для пошуку ейлерова циклу скористаємося тим, що ейлерів цикл - це об'єднання всіх простих циклів графа.   
Отже, завдання – ефективнознайти всі цикли і ефективно об'єднати їх в один.

Отже, ми отримаємо цикли:



* [7-1-8-10-9-7]
* [7-8-2-7]
* [6-11-7-3-5-6]
* [6-2-3-4-6]
* [6-10-11-5-12-8]

1. **Спростити формули (привести їх до скороченої ДНФ).**



= =