

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

«ЛЬВІВСЬКА ПОЛІТЕХНІКА»



Лабораторна робота №3

з курсу «Дискретні моделі в САПР»:

## АЛГОРИТМ РІШЕННЯ ЗАДАЧІ КОМІВОЯЖЕРА

**Виконав:**

Ст.гр.КН-409

Погуляєв В.В.

Львів – 2023

## Мета роботи

Метою даної лабораторної роботи є вивчення і дослідження алгоритмів рішення задачі комівояжера.

## Теоретичні відомості

Невідомо, коли проблему комівояжера було досліджено вперше. Однак, відома видана в 1832 році книжка з назвою «Комівояжер — як він має поводитись і що має робити для того, аби доставляти товар та мати успіх в своїх справах — поради старого Кур'єра», в якій описано проблему, але математичний апарат для її розв'язання не застосовується. Натомість, в ній запропоновано приклади маршрутів для деяких регіонів Німеччини та Швейцарії. Раннім варіантом задачі може розглядатись англ. Icosian Game Вільяма Гамільтона 19 століття, яка полягала в тому, щоб знайти маршрути на графі з 20 вузлами. Перші згадки в якості математичної задачі на оптимізацію нелажать Карлу Менгеру (нім. Karl Menger), який сформулював її в математичному колоквиумі в 1930 році наступним чином: Ми називаємо проблемою женця (оскільки це питання виникає в кожного листоноші, зокрема, її вирішують багато мандрівників) завдання віднайти найкоротший шлях між скінченною множиною місць, відстань між якими відома Невдовзі з'явилась відома зараз назва задача мандруючого продавця (англ. Traveling Salesman Problem), яку запропонував Гаслер Вітні (англ. Hassler Whitney) з Принстонського Університету Для можливості застосування математичного апарату для розв'язання проблеми, її слід представити у вигляді математичної моделі. Проблема комівояжера можна представити у вигляді моделі на графі, тобто, використовуючи вершини та ребра між ними. Таким чином, вершини графу відповідають містам, а ребра між вершинами і та  $j$  сполучення між цими містами. У відповідність кожному ребру можна зіставити вагу, яку можна розуміти як, наприклад, відстань між містами, час або вартість подорожі. Маршрутом (також гамільтоновим маршрутом) називається маршрут на цьому графі до якого входить по одному разу кожна вершина графа. Задача полягає у відшукуванні найкоротшого маршруту.

## Лабораторне завдання

Реалізувати алгоритм рішення задачі Комівояжера методом гілок і меж.

## Код програми

Метод який повертає вже готовий фінальний шлях:

```
public void TSP(int adj[][]) {
    int curr_path[] = new int[N + 1];

    int curr_bound = 0;
    Arrays.fill(curr_path, -1);
    Arrays.fill(visited, false);

    for (int i = 0; i < N; i++)
        curr_bound += (firstMin(adj, i) +
                      secondMin(adj, i));
```

```
curr_bound = (curr_bound == 1) ? curr_bound / 2 + 1 :  
             curr_bound / 2;  
  
visited[0] = true;  
curr_path[0] = 0;  
  
TSPRec(adj, curr_bound, 0, 1, curr_path);  
}
```

Посилання на GitHub [https://github.com/flippfflopp/DM\\_Pohuliaiev](https://github.com/flippfflopp/DM_Pohuliaiev)

**Аналіз результатів**

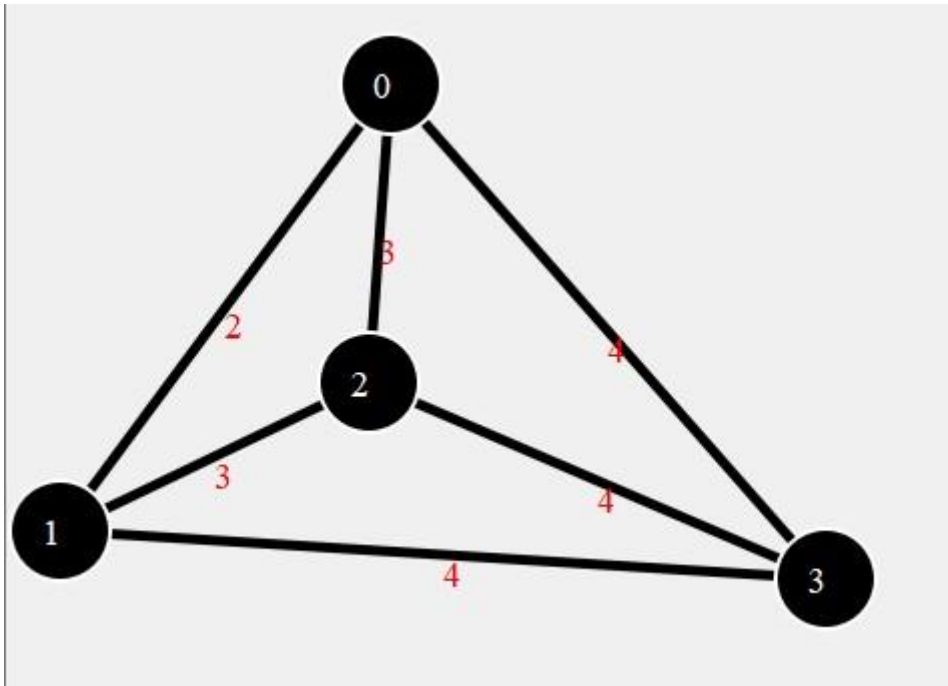
**Аналітичний розв’язок:**

Матриця суміжності:

0 2 3 4  
1 0 3 4  
1 2 0 4  
1 2 3 0

Граф:

$V = \{0, 1, 2, 3\}$   
 $E = \{01(2), 02(3), 03(4), 12(3), 13(4), 23(4)\}$   
 $G = \{V, E\}$



Аналітичний обрахунок:

Насперед, перевіримо чи даний граф має гамільтоновий цикл за допомогою умови Дірака:

$p/2 = 4/2 = 2$ , де  $p$  – кількість вершин.

$\text{Deg}(0) = 3$ .

$\text{Deg}(1) = 3$ .

$\text{Deg}(2) = 3$ .

$\text{Deg}(3) = 3$ .

Степінь кожної вершини не менший ніж  $p/2$ , значить даний граф є гамільтоновий.

Тепер вибираємо навмання вершину з якої почнемо пошук гамільтонового шляху, нехай це буде вершина «0». Після чого необхідно вибрати наступну вершину, так щоб вага дуги було найменшою, і так продовжуємо поки не дійдемо до початкової вершини. В результаті чого в нас вийшов такий шлях:

$0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 0 = 13$  – ціна оптимального шляху.

Результат виконання програми для аналітичного обрахунку:

```
Minimum cost : 13
Path Taken : 0 1 2 3 0
Process finished with exit code 0
```

Результат виконання програми використовуючи тестовий файл I3\_1.txt:

```
Minimum cost : 254
Path Taken : 0 2 4 1 3 5 0
Process finished with exit code 0
```

Результат виконання програми використовуючи тестовий файл I3\_2.txt:

```
Minimum cost : 262
Path Taken : 0 2 3 1 4 5 0
Process finished with exit code 0
```

Результат виконання програми використовуючи тестовий файл I3\_3.txt:

```
Minimum cost : 314
Path Taken : 0 1 4 3 2 5 0
Process finished with exit code 0
```

### Висновок

В ході виконання лабораторної роботи, вивчив і дослідив алгоритми рішення задачі комівояжера.