МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

КРЕМЕНЧУЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ   
ІМЕНІ МИХАЙЛА ОСТРОГРАДСЬКОГО

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕЛЕКТРИЧНОЇ ІНЖЕНЕРІЇ   
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА АВТОМАТИЗАЦІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

НАВЧАЛЬНА ДИСЦИПЛІНА  
 «Алгоритми та методи обчислень»

ЗВІТ

З ПРАКТИЧНИХ РОБІТ

Виконав:

студент групи КН-23-1

Ярковий Т.С.

Кременчук 2023

**Практична робота № 8**

**Тема. Жадібні алгоритми. Наближене розв’язання екстремальних задач**

**Мета:** набути практичних навичок застосування деяких жадібних алгоритмів для розв’язання екстремальних задач.

**Завдання**

1. Розв’язати задачу комівояжера для графа, заданого варіантом, використовуючи код, наведений вище.

2. Візуалізувати граф.

3. Обґрунтувати асимптотику для обох алгоритмів, неведену в табл. 1.4.

7. Заданий зважений граф: [(1,3,10), (1,4,15), (1,5,20), (2,3,25), (2,4,30), (2,5,35)]

Початкові дані: Шлях S[]. Довжина W=0. Поточна вершина V=0. Найближча вершина T=0 (вершина, довжина ребра).

Починаємо розглядати, обираємо початковою вершиною вершину 4.

S[4]; W=0; V=4; T=(1, 15);

Найближча вершина 1, додаємо її і продовжуємо розглядати:

S[4, 1]; W=15; V=1; T=(3, 10);

Найближча вершина 3, додаємо її і продовжуємо розглядати:

S[4, 1, 3]; W=25; V=3; T=(2, 25);

Найближча вершина 2, додаємо її і продовжуємо розглядати:

S[4, 1, 3, 2]; W=50; V=2; T=(5, 35);

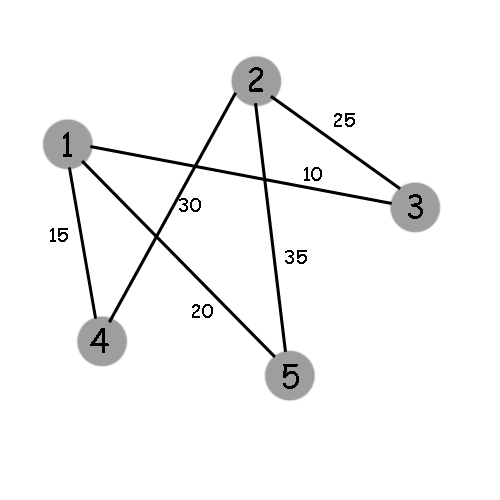
Найближча вершина 5, додаємо її і продовжуємо розглядати:

S[4, 1, 3, 2, 5]; W=85; V=5; T=();

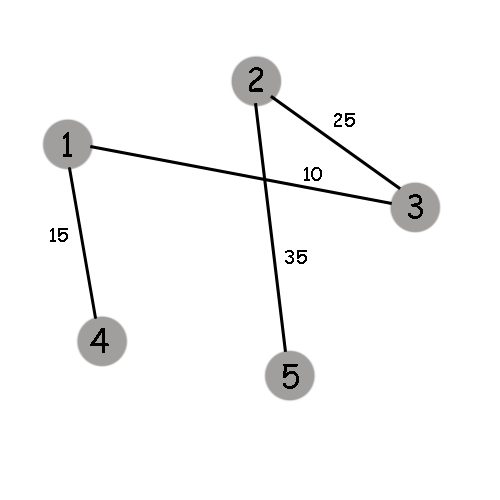
Усі вершини розглянуті.

Отриманий маршрут S[4, 1, 3, 2, 5]. Довжина маршруту: W=85.

Візуалізація:



Маршрут:



Обґрунтувати асимптотику для обох алгоритмів, неведену в табл. 1.4.

Складність алгоритму найближчого сусіда: у цьому алгоритмі побудова дерева займає час log n. Пошук відбувається для n вершин і у кожній вершині розглядаються усі варіанти куди можна піти, тому складність пошуку займає n^2.

Для алгоримту грубої сили складність n! через необхідність розглядати усі можливі варіанти для кожної вершини (розглядаємо усі можливі комбінації для n вершин).

**Контрольні питання**

1. Що таке жадібний алгоритм?

Жадібний алгоритм – це метод, що робить локально оптимальні вибори, сподіваючись, що вони приведуть до глобально оптимального розв’язку.

2. Які головні принципи роботи жадібних алгоритмів?

На кожному кроці обираємо локально оптимальний вибір, що часто дає дуже хороші наближені розв’язки та допомагають розв’язувати задачі з великою складністю.

3. Яка головна відмінність між жадібними алгоритмами та динамічним програмуванням?

У динамічному програмуванні на кожному кроці приймається рішення, яке також враховує наступні кроки, це дає глобально оптимальний розв’язок.

У жадібних алгоритмах вибір приймається спираються тільки на локально найкращий варіант.

4. Наведіть приклади задач, які можна розв’язати за допомогою жадібних алгоритмів.

Знаходження найкоротшого шляху від однієї вершини до всіх інших у зваженому ациклічному графі, знаходження мінімального кістякового дерева, задача про рюкзак, задача комівояжера.

5. Які можуть бути обмеження у використанні жадібних алгоритмів для розв'язання екстремальних задач?

Основні обмеження: не завжди дає оптимальний розв’язок, їх ефективність залежить від структури задачі та їх складність може бути NP-складною для деяких задач/

6. Чому жадібні алгоритми часто використовуються для наближеного розв’язання екстремальних задач?

Вони часто дають дуже хороші наближені розв’язки, при цьому не потребуючи повного перебору усіх варіантів.