Основи програмування – 2. Алгоритми та структури даних

Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 3 з дисципліни

«Алгоритми та структури даних-2.

Структури даних»

«Прикладні задачі теорії графів частина 1»

Варіант 5

Виконав студент ІП-15, Буяло Дмитро Олександрович

(шифр, прізвище, ім'я, по батькові)

Перевірив Соколовський Владислав Володимирович

(прізвище, ім'я, по батькові)

Київ 2022

Основи програмування – 2. Алгоритми та структури даних

**Лабораторна робота 3**

**Прикладні задачі теорії графів частина 1**

**Мета** – вивчити основні прикладні алгоритми на графах та способи їх імплементації.

**Індивідуальне завдання**

**Варіант 5**

**Завдання**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Задача** | **Алгоритм** | **Тип графу** | **Спосіб задання** |
| 5 | Пошук найкоротшого шляху між парою вершин | Дейкстри | Орієнтований | Матриця вагів |

Основи програмування – 2. Алгоритми та структури даних

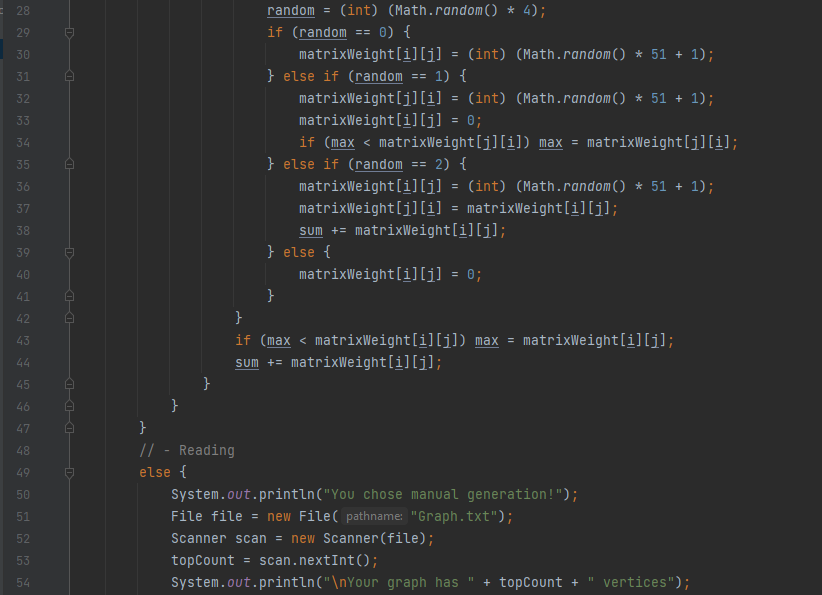
1. Псевдокод алгоритму

|  |
| --- |
| **Початок** |
| Введення start |
| **Повторити** для і від 0 до topCount |
| minDist[i] = sum |
| vert[i] = 1 |
| **Все повторити** |
| minDist[start] = 0 |
| **Повторити** |
| minIndex = sum |
| min = minIndex |
| **Повторити** для і від 0 до topCount |
| **Якщо** vert[i] = 1 та minDist[i] < min |
| то |
| min = minDist[i] |
| minIndex = i |
| **Все якщо** |
| **Якщо** minIndex != sum |
| **то** |
| **Повторити** для і від 0 до topCount |
| **Якщо** matrixWeight[minIndex][i] > 0 та |
| min + matrixWeight[minIndex][i] < minDist[i] |
| **то** |
| minDist[i] = min + matrixWeight[minIndex][i] |
| **Все якщо** |
| **Все повторити** |
| vert[minIndex] = 0 |
| **Все якщо** |
| **Поки** minIndex < sum |
| **Все повторити** |
| **Кінець** |

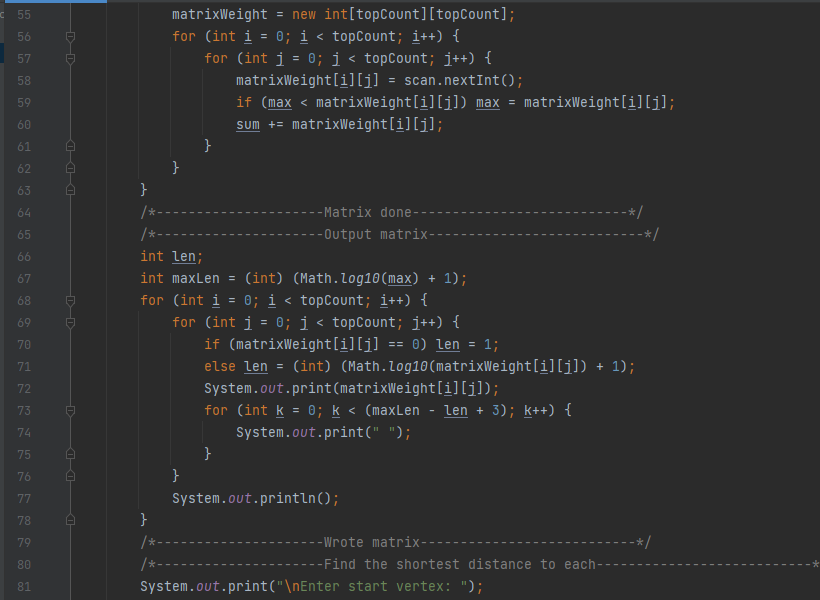
Основи програмування – 2. Алгоритми та структури даних

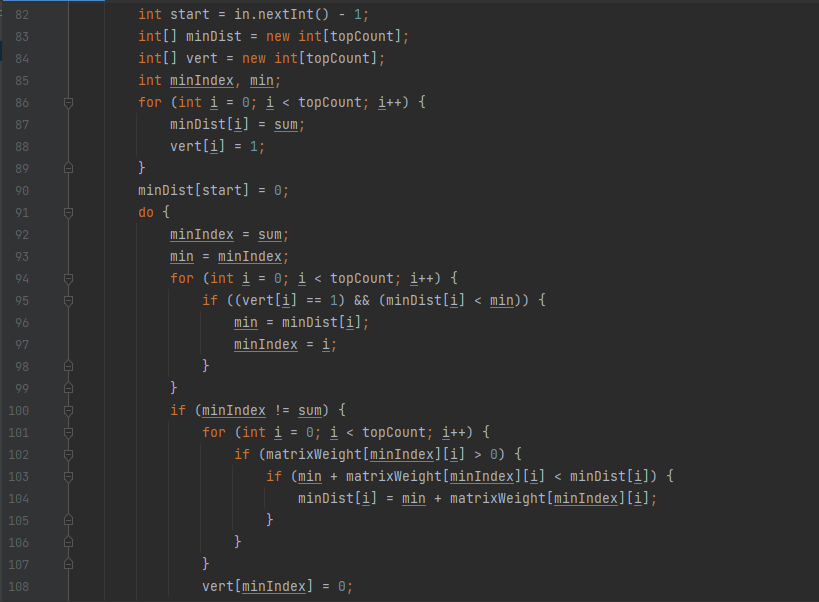
1. Програмна реалізація алгоритму
   1. Вихідний код



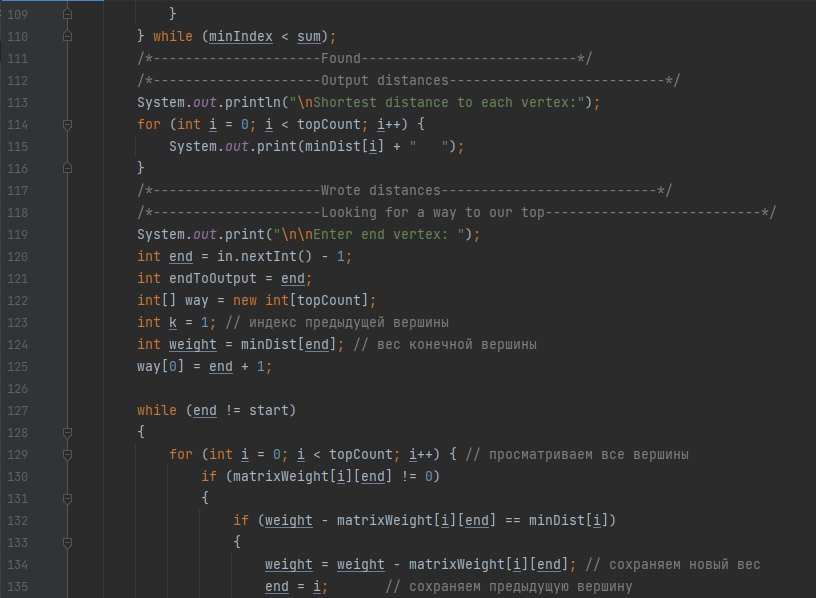


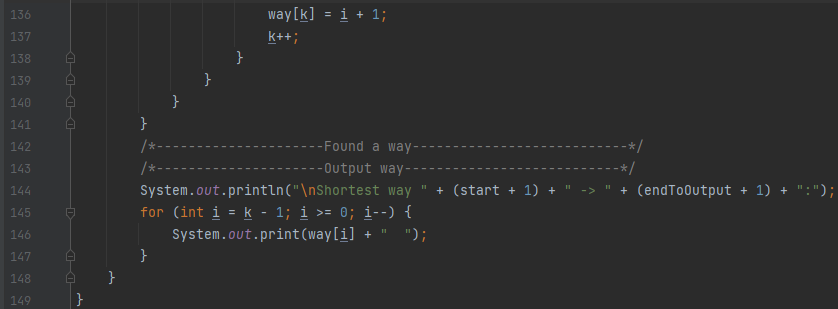
Основи програмування – 2. Алгоритми та структури даних





Основи програмування – 2. Алгоритми та структури даних





* 1. Приклад роботи

На рисунках 2.1 і 2.2 показані приклади роботи програми для графів на 7 і 15 вершин відповідно.

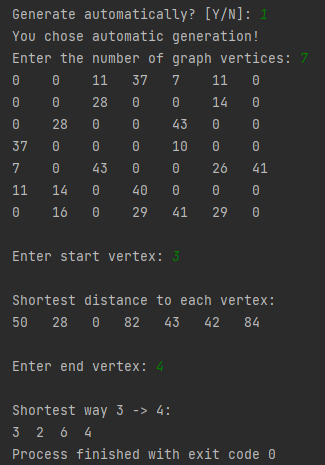


Рисунок 2.1 – Приклад роботи на 7 вершинах

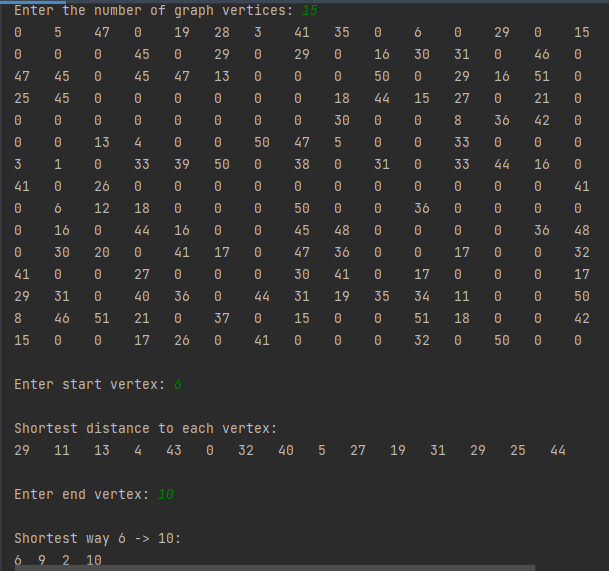


Рисунок 2.2 – Приклад роботи на 15 вершинах

На рисунках 2.31 і 2.4 виконана перевірка для наведених графів на 7 і 15 вершин відповідно.

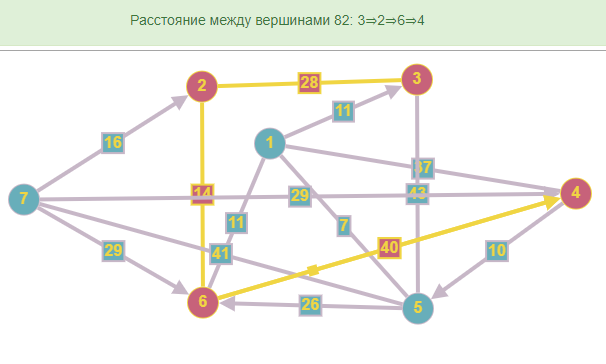


Рисунок 2.3 – Перевірка роботи для графа на 7 вершинах

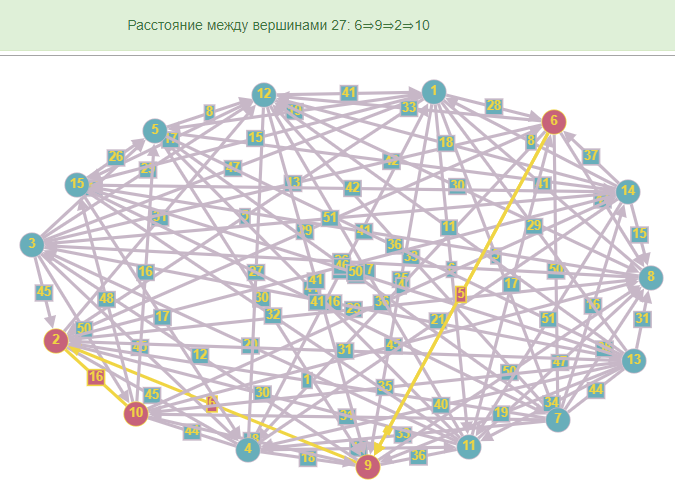


Рисунок 2.4 – Перевірка роботи для графа на 15 вершинах

1. Розв’язання задачі вручну

На рисунку 3.1 наведено ручне розв’язання задачі для графу на 9 вершин.

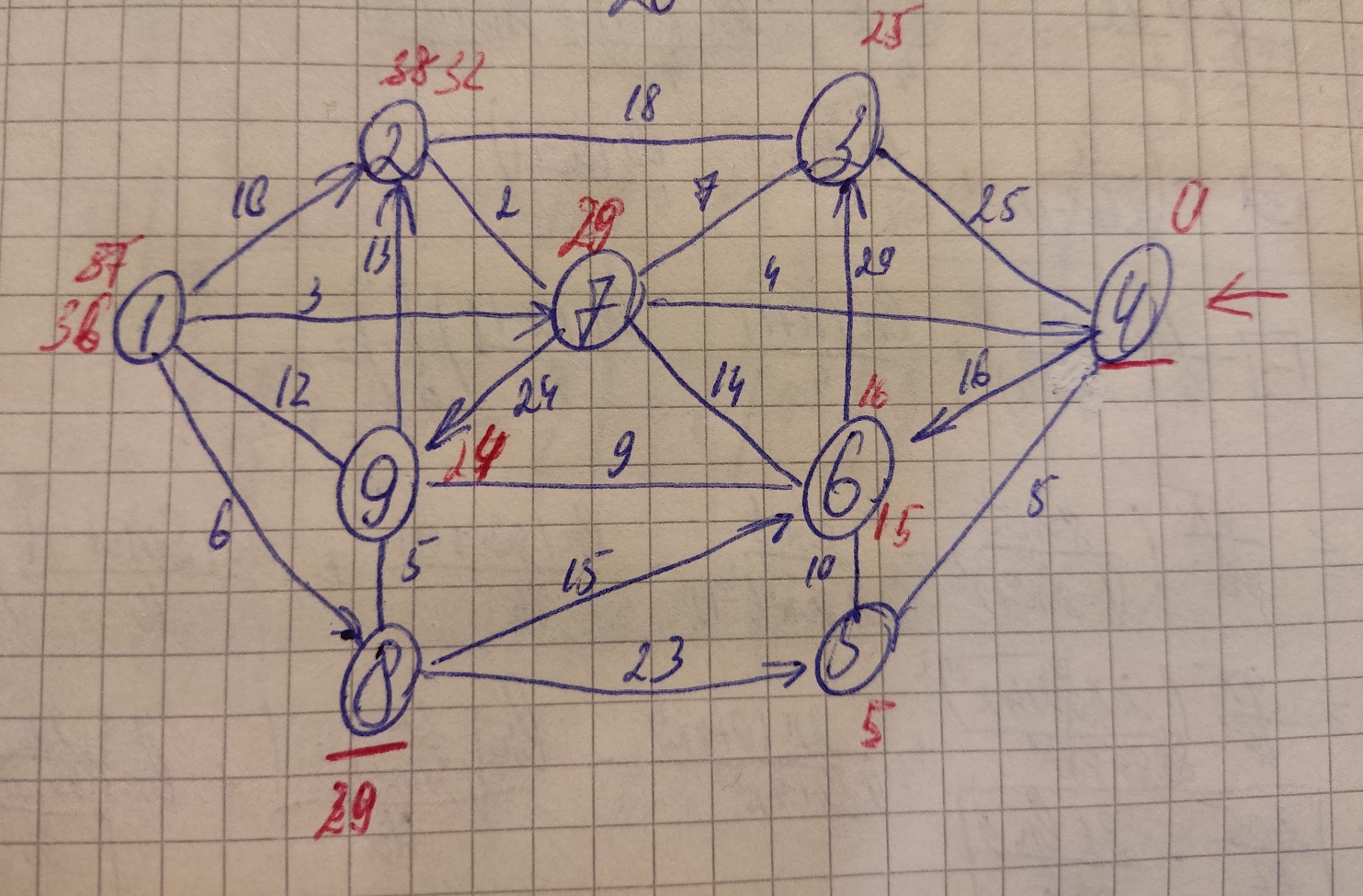


Рисунок 3.1 – Ручне розв’язання

На рисунку 3.2 наведено умову для цієї задачі

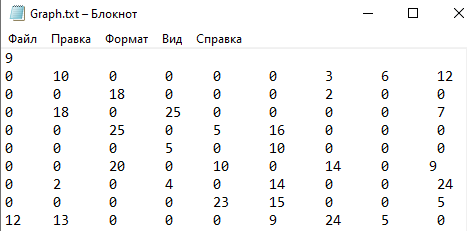


Рисунок 3.2 – Умова

На рисунку 3.3 наведено розв’язок цієї задачі створеною програмою

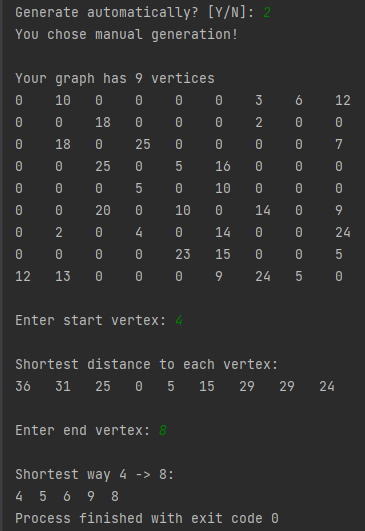


Рисунок 3.3 – Програмне рішення

На рисунку 3.4 виконана перевірка для наведеного графу на 9 вершин.

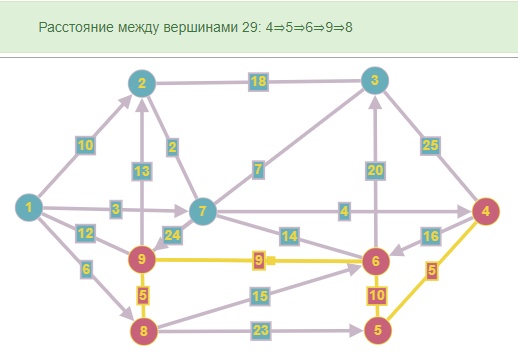


Рисунок 3.4 – Перевірка роботи для графа на 9 вершинах

**ВИСНОВОК**

При виконанні третьої лабораторної роботи, вивчили алгоритм Дейкстри для графів та спосіб його імплементації, задаючи вершини графу матрицею вагів, яка може генеруватися автоматично чи задаватися вручну. Алгоритм Дейкстри знаходить найкоротший шлях до кожної з вершин.

Програмне розв’язання подібне до ручного, але дозволяє проводити розрахунки набагато швидше.