**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота №5**

З дисципліни

“Дискретна математика”

**Виконав:**

студент групи КН-112

Думич Іван

**Викладач:**

Мельникова Н.І.

Львів – 2019 р.

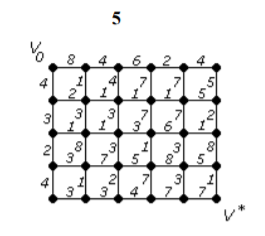
**Тема:** Знаходження найкоротшого маршруту за алгоритмом Дейкстри. Плоскі планарні графи.

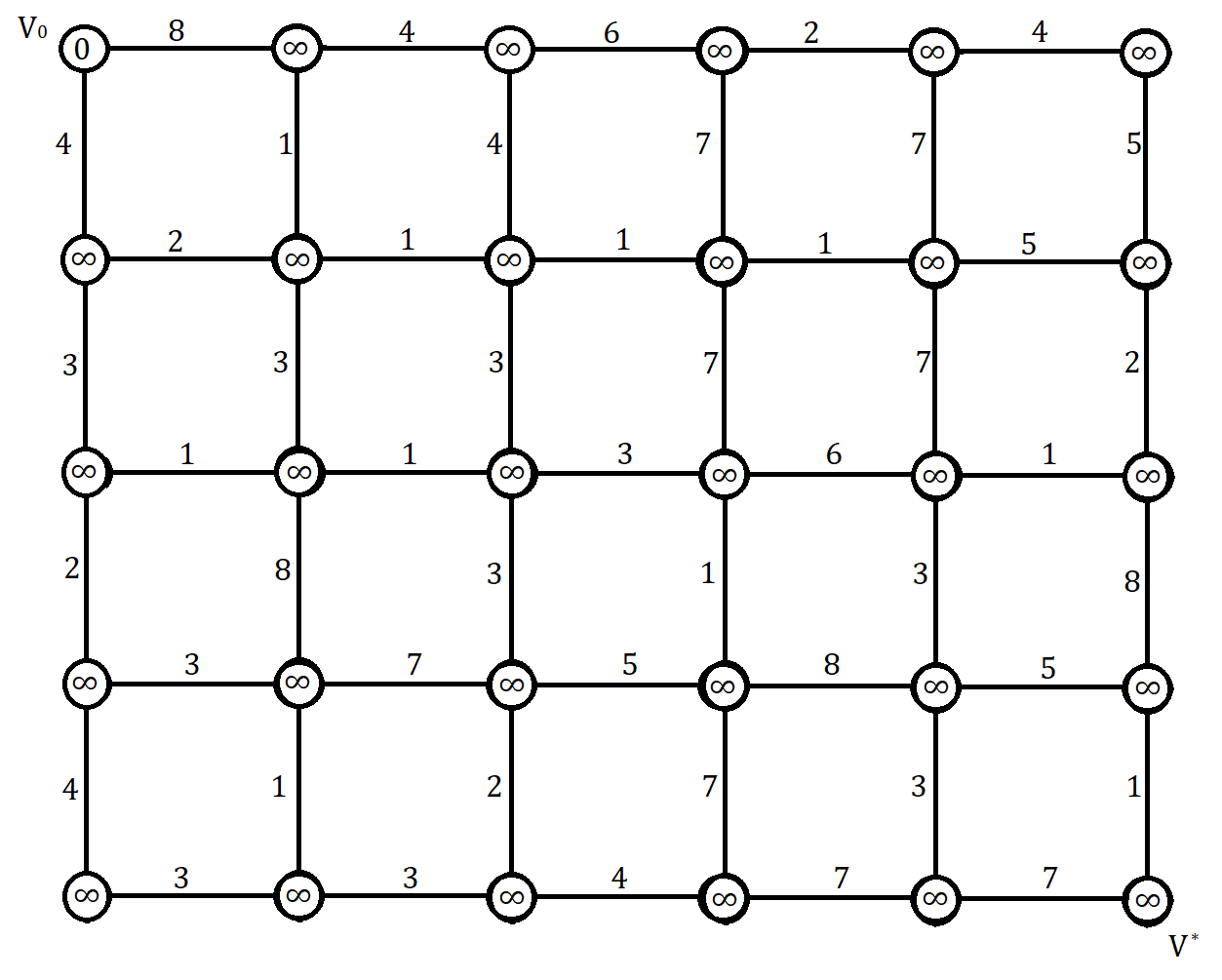
**Мета роботи:** набуття практичних вмінь та навичок з використання алгоритму Дейкстри.

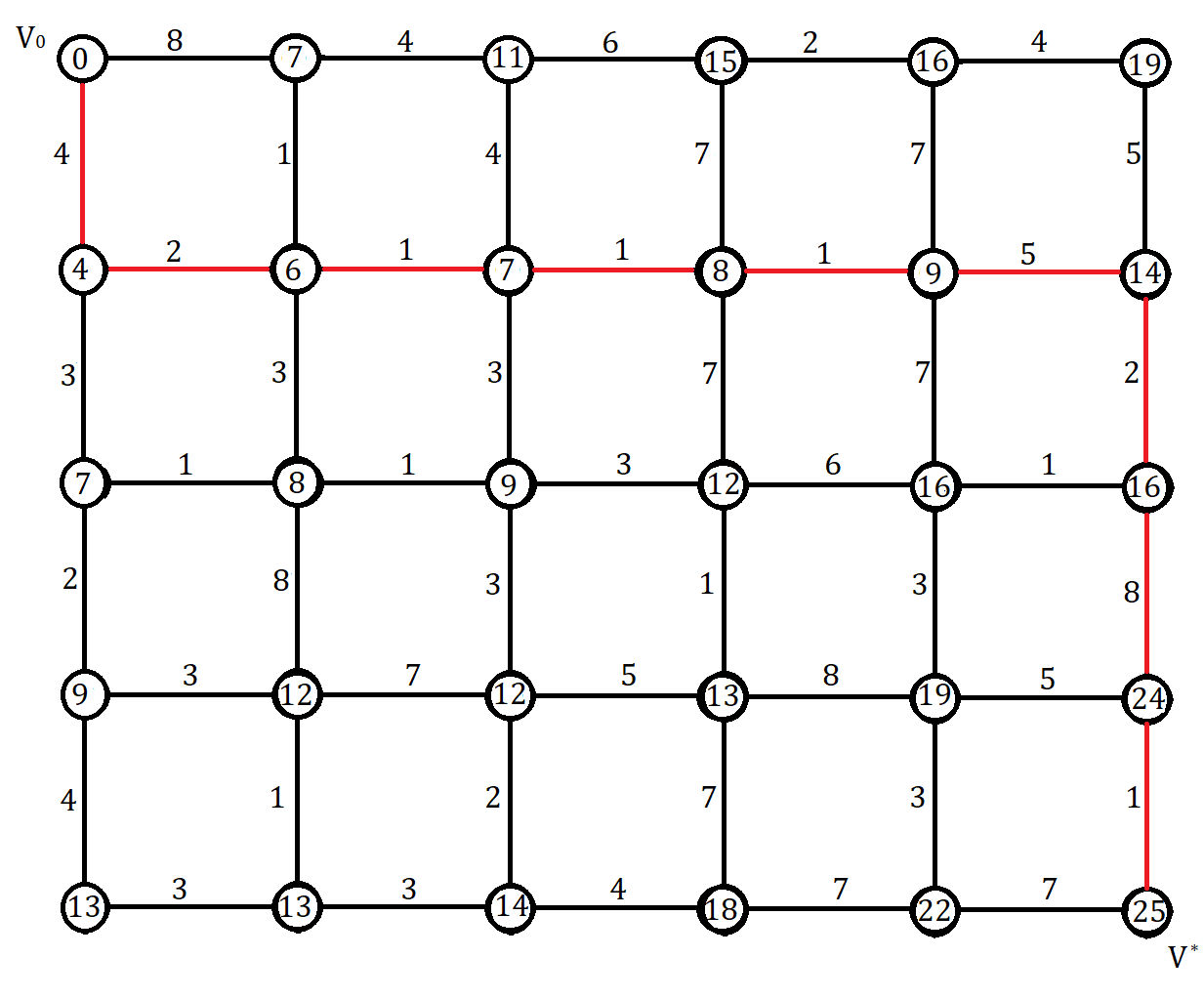
**Варіант № 5**

**Завдання № 1.** Розв'язати на графах наступні 2 задачі:

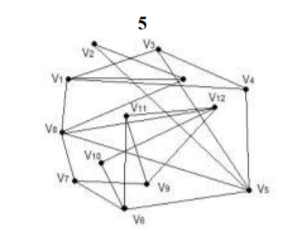
1. За допомогою алгоритму Дейкстра знайти найкоротший шлях у графі поміж парою вершин V0 і V \* .

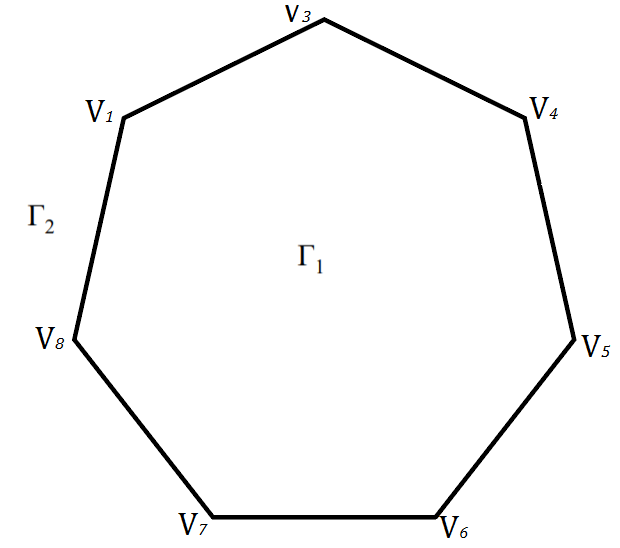


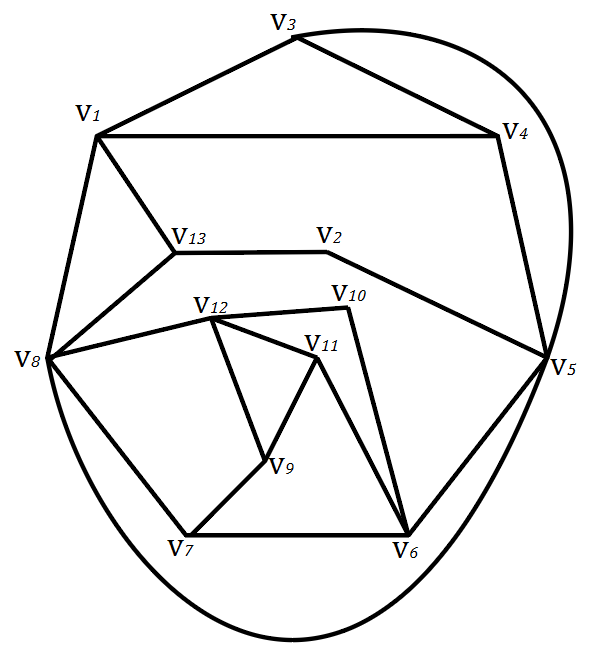


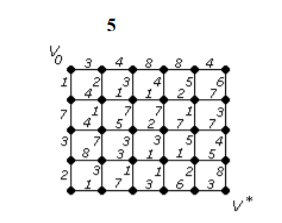


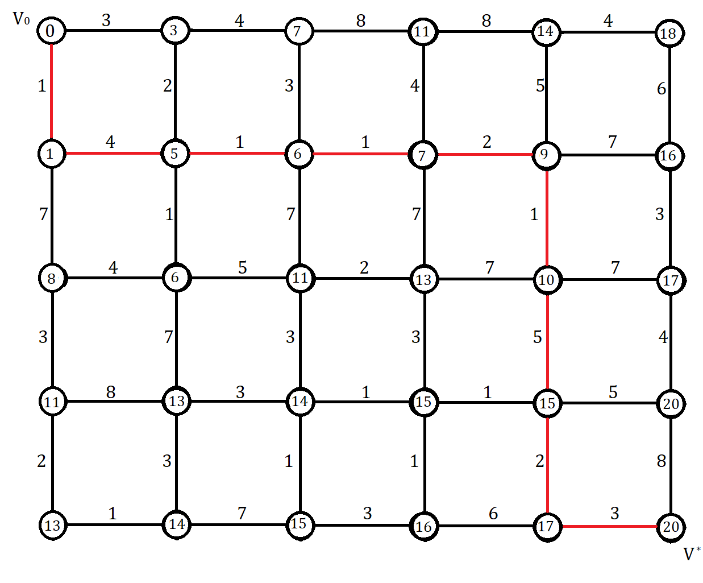
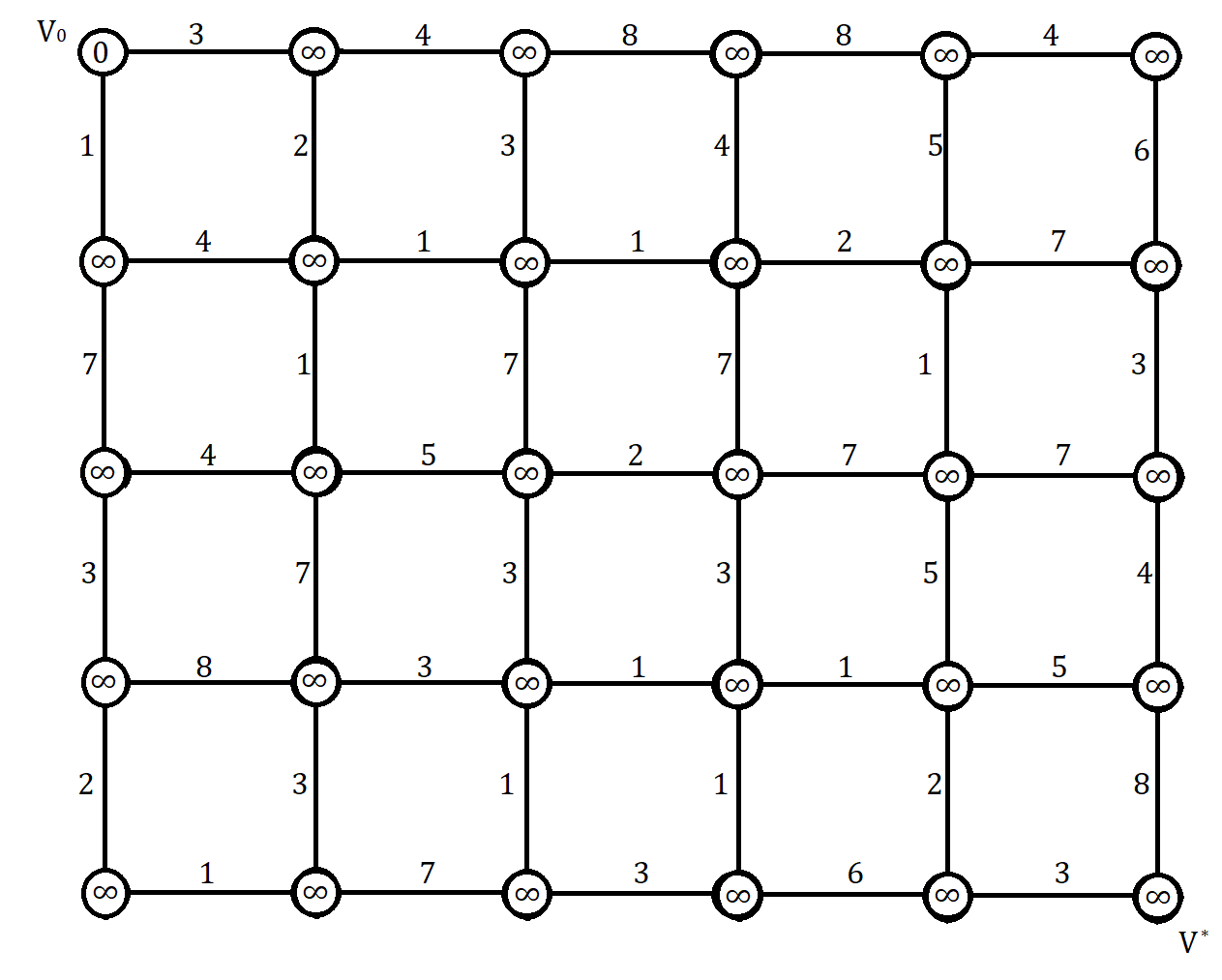
2. За допомогою γ -алгоритма зробити укладку графа у площині, або довести що вона неможлива.

****

****

****

**Завдання №2.** Написати програму, яка реалізує алгоритм Дейкстри знаходження найкоротшого шляху між парою вершин у графі. Протестувати розроблену програму на графі згідно свого варіанту. 



**Програмна реалізація:**

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

#define inf 1000000

using namespace std;

int min\_top(int\*\* arr,int v) {

    int m=0;

    for (int i = 0; i < v; i++) {

        if (arr[i][1]) {

            m = i; break;

        }

    }

    for (int i = 1; i < v; i++) {

        if (arr[m][0] >= arr[i][0] && arr[i][1]==1) {

            m = i;

        }

    }

    return m;

}

int main()

{

    setlocale(LC\_ALL, "Ukrainian");

    int a, b, c;

    int v = 0;

    cout << "Кiлькiсть вершин графа : ";

    cin >> v;

    int\*\* graph = new int\* [v];

    for (int j = 0; j < v; j++) {

        graph[j] = new int[v];

    }

    for (int a = 0; a < v; a++) {

        for (int j = 0; j < v; j++) {

             graph[a][j] = 0;

        }

    }

    cout << "Введiть вагу ребер графа : " << endl;

    while (true) {

        cin >> a;

        if (a == -1) { break; }

        cin >> b;

        cin >> c;

        graph[a-1][b-1] = graph[b-1][a-1] = c;

    }

    int p;

    int\*\* tops = new int\*[v];

    for (int j = 0; j < v; j++) {

        tops[j] = new int[2];

    }

    int\* tops\_path = new int[v];

    cout << "Вихiдна вершина: ";

    cin >> p;

    for (int i = 0; i < v; i++) {

        if (i == p-1) {

            tops[i][0] = 0;

            tops[i][1] = 1;

        }

        else {

            tops[i][0] = inf;

            tops[i][1] = 1;

        }

    }

    tops\_path[p-1] = 0;

    int m;

    for (int i = 0; i < v; i++) {

        m = min\_top(tops, v);

        for (int j = 0; j < v; j++) {

            if (graph[m][j]) {

                if (tops[j][0] > tops[m][0] + (graph[m][j])) {

                    tops[j][0] = tops[m][0] + (graph[m][j]);

                    tops\_path[j] = m;

                }

            }

        }

        tops[m][1] = 0;

    }

    ////шлях

    cout << "Введiть потрiбну вершину: ";

    int k; cin >> k;

    cout << "Мiнiмальний шлях: ";

    cout << tops[k-1][0];

    cout << endl << k <<" <-- ";

    k--;

    for (int a = 0; tops\_path[k] != p-1; a++) {

        cout << tops\_path[k]+1 <<" <-- ";

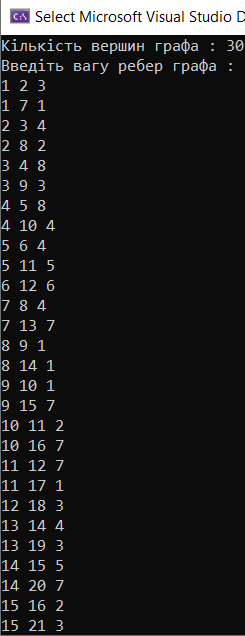
        k = tops\_path[k];

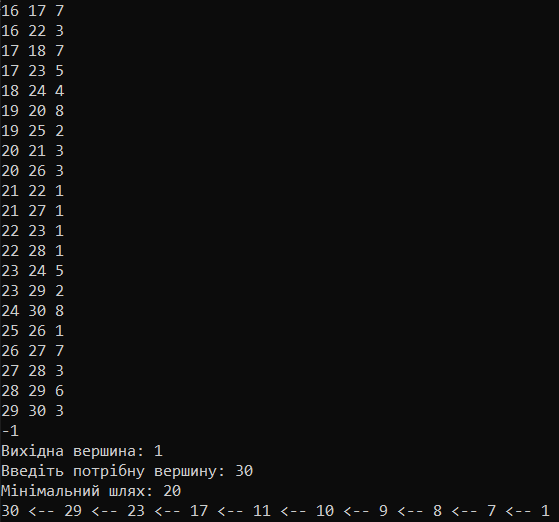
    }

    cout << p << endl;

    return 0;

}

**Результат:**



**Висновок**: я набув практичних вмінь та навичок з використання алгоритму Дейкстри.