

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА
ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Лабораторна робота №5

З дисципліни

“Дискретна математика”

Виконав:

Студент групи КН-115

Лукавий Мар'ян

Викладач:

Мельникова Н.І.

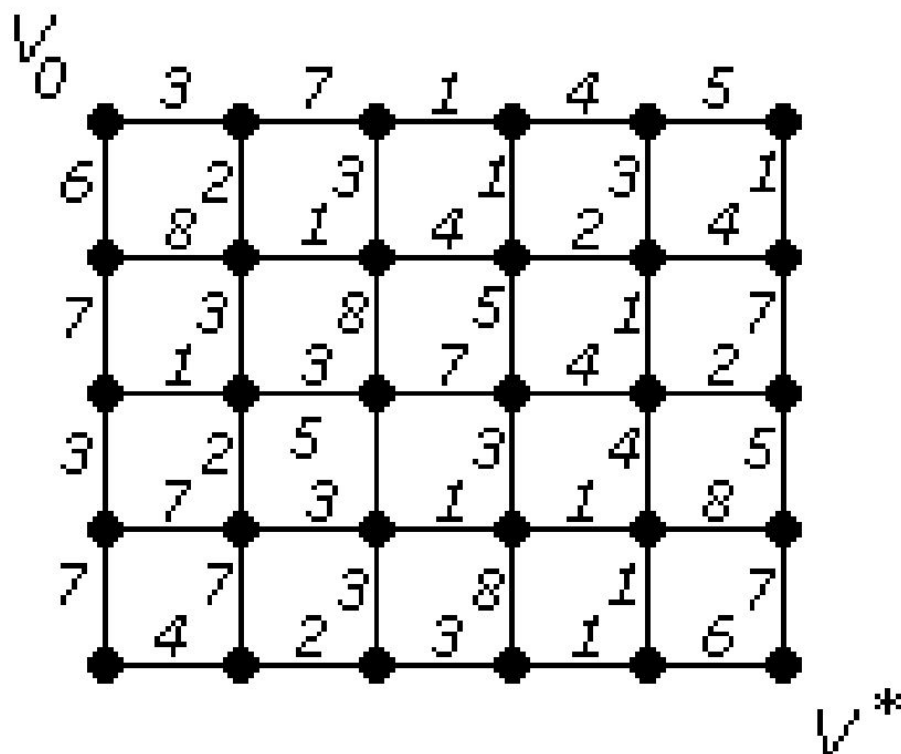
Тема: Знаходження найкоротшого маршруту за алгоритмом Дейкстри. Плоскі планарні графи

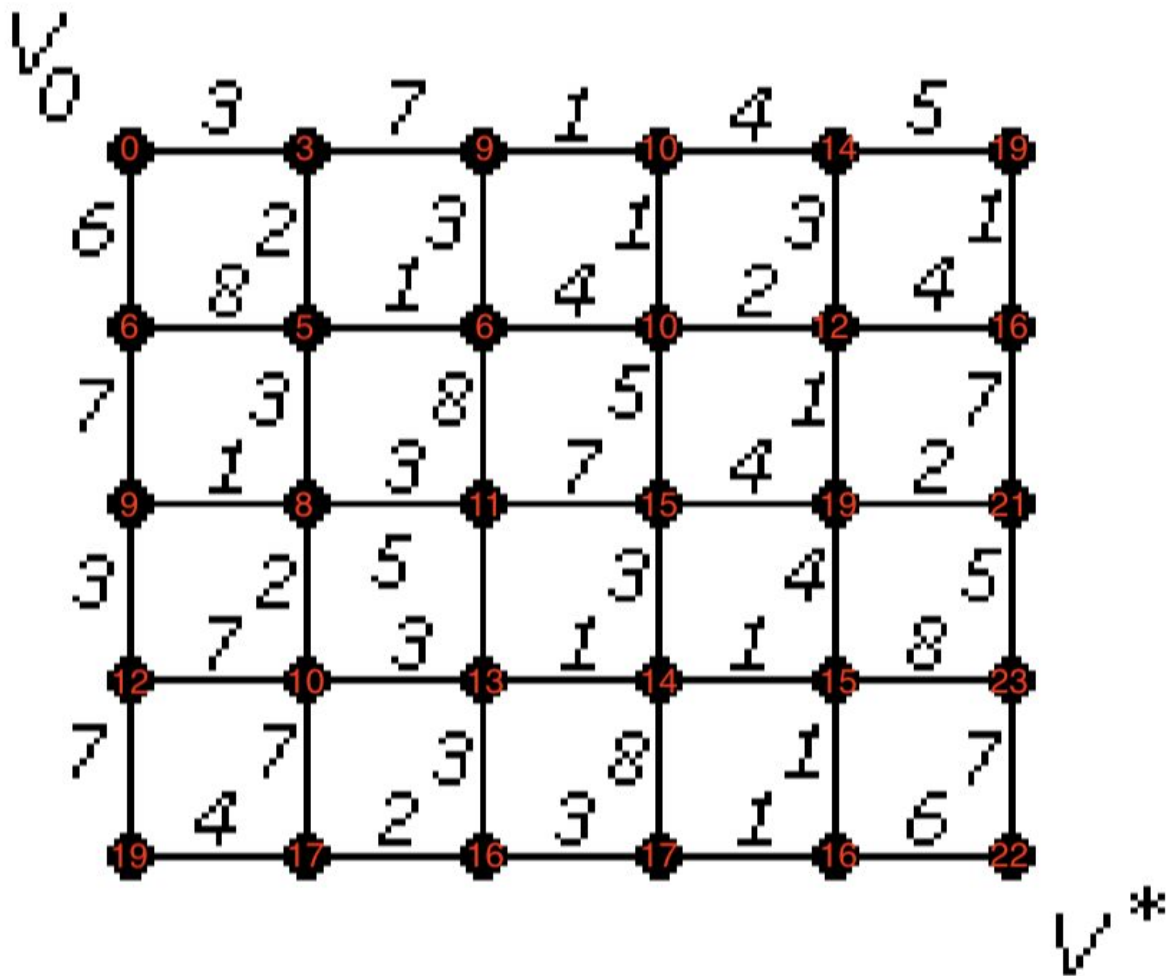
Мета роботи: набуття практичних вмінь та навичок з використання алгоритму Дейкстри.

Варіант 16

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

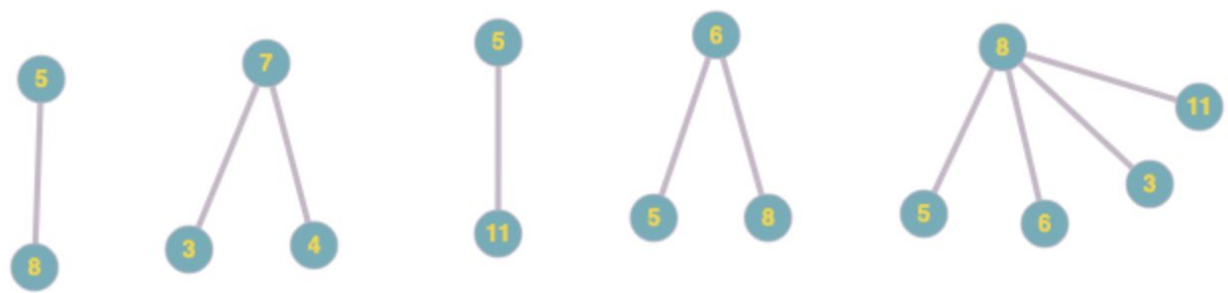
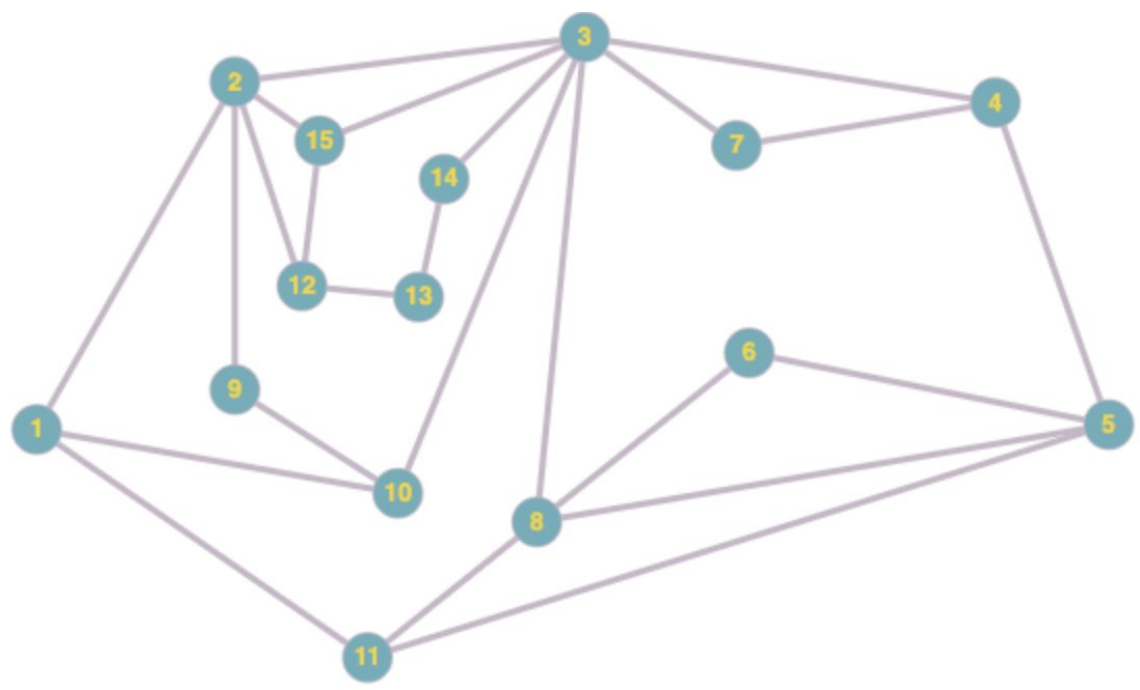
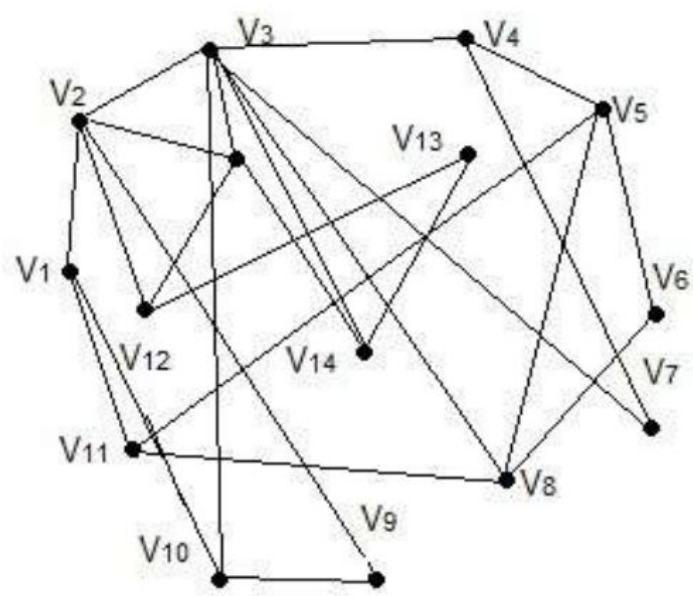
1. За допомогою алгоритму Дейкстри знайти найкоротший шлях у графі поміж парою вершин V_0 і V^* .

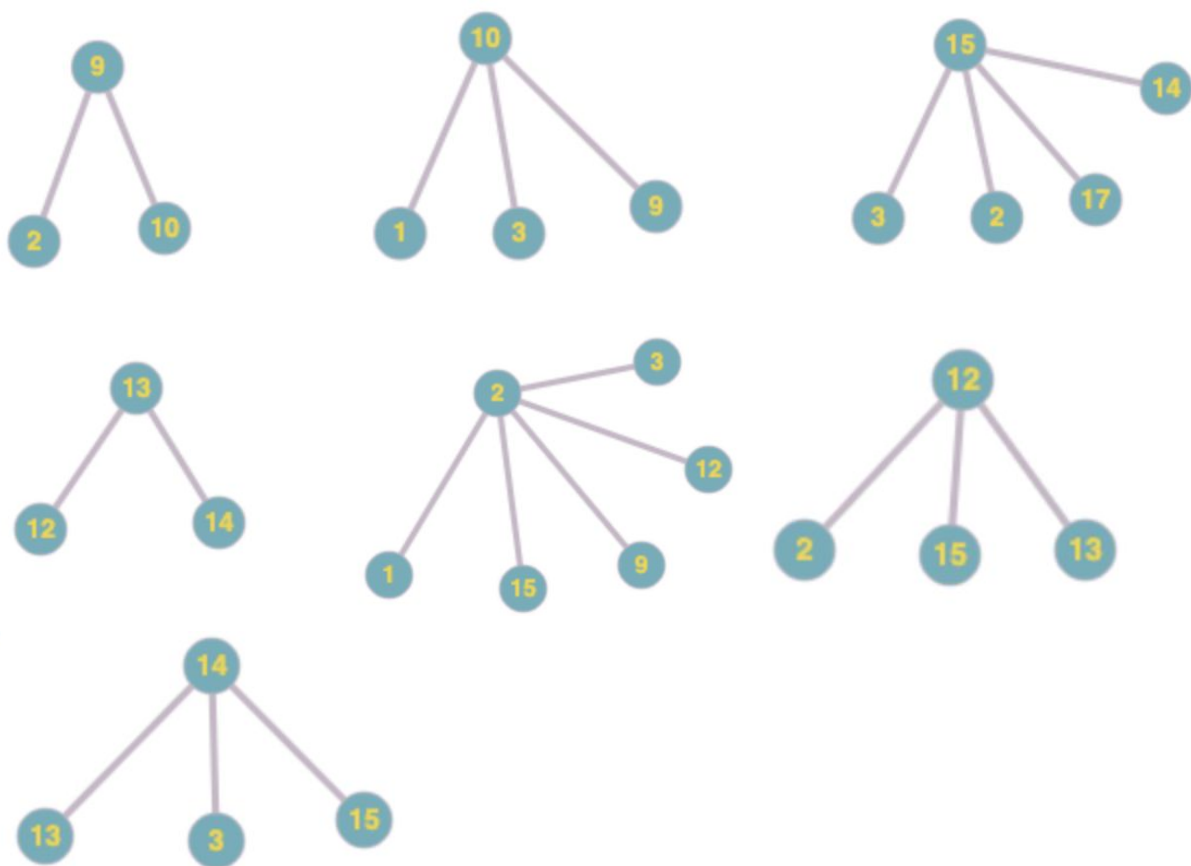




Найкоротший шлях у графі поміж парою вершин V_0 і V^* - 22.

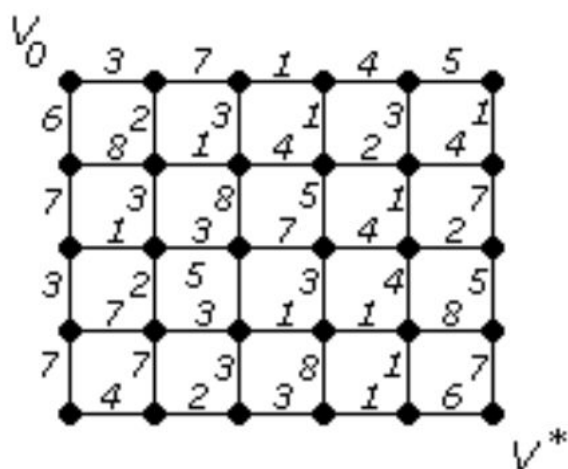
2. За допомогою -алгоритма зробити укладку графа у площині, або довести що вона неможлива.





Завдання №2.

Написати програму, яка реалізує алгоритм Дейкстри знаходження найкоротшого шляху між парою вершин у графі. Протестувати розроблену програму на графі згідно свого варіанту.




```

int shortest_dist[30];
bool visited[30] = {0};
int parent[30];

int getNearest() {
    int minValue = 999, minNode = 0;
    for (int i = 0; i < V; i++) {
        if (!visited[i] && shortest_dist[i] < minValue) {
            minValue = shortest_dist[i];
            minNode = i;
        }
    }
    return minNode;
}

void dijkstra() {
    for (int i = 0; i < V; i++) {
        int nearest = getNearest();
        visited[nearest] = true;
        for (int adj = 0; adj < V; adj++) {
            if (matrix[nearest][adj] != INF && shortest_dist[adj] >
                shortest_dist[nearest] + matrix[nearest][adj]) {
                shortest_dist[adj] = shortest_dist[nearest] + matrix[nearest][adj];
                parent[adj] = nearest;
            }
        }
    }
}

```



```

int main(void) {
    V = 30;
    for (int i = 0; i < V; i++) {
        for (int j = 0; j < V; j++) {
            if (matrix[i][j] == 0 && i != j) {
                matrix[i][j] = 999;
            }
        }
    }
    start = 0;
    for (int i = 0; i < V; i++) {
        parent[i] = i;
        shortest_dist[i] = INF;
    }
    shortest_dist[start] = 0;

    dijkstra();

    cout << "Довжина мінамального шляху : ";
    cout << shortest_dist[V - 1] << endl;
    cout << "Шлях : ";
    cout << endl;
    cout << V << " ";
    pnode = parent[V - 1];
    cout << endl;

    while (pnode != start) {
        cout << pnode + 1 << endl;

        pnode = parent[pnode];
    }
    cout << '1';
    cout << endl;

    return 0;
}

```