

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

**Кафедра систем штучного інтелекту**

**Лабораторна робота № 5**

З дисципліни «Дискретна математика»

Виконав:

студент групи КН-112

Матвіїв Остап-Василь

Викладач:

Мельникова Н.І.

Львів – 2019 р.

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

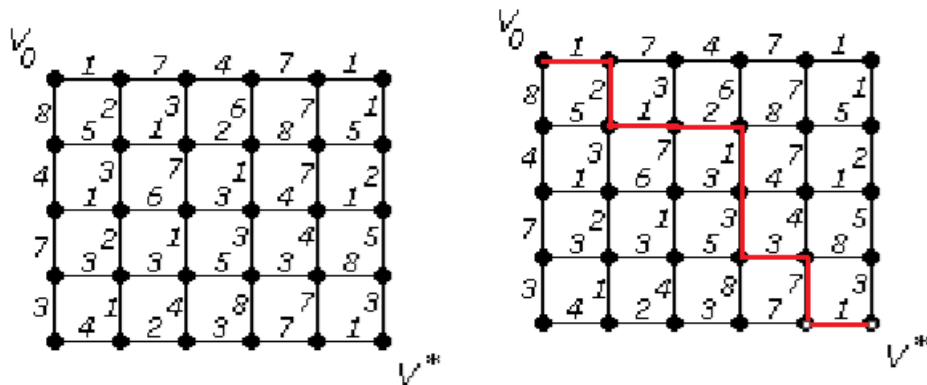
**Мета роботи:** набуття практичних вмінь та навичок з використання алгоритму Дейкстри.

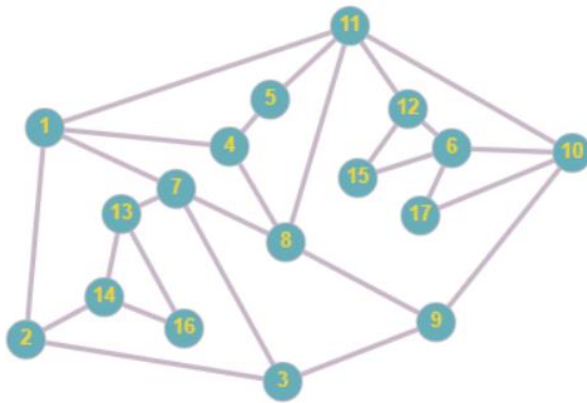
### Варіант-11

**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  $V^*$ .

11

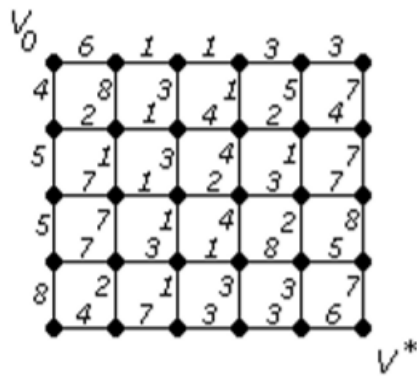




3)

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

11



```

4  #include <stdio.h>
5  #include <stdlib.h>
6  #define SIZE 30
7  using namespace std;
8  int main()
9  {
10     int a[SIZE][SIZE];
11     int d[SIZE];
12     int v[SIZE];
13     int temp, minindex, min;
14     int begin_index = 0;
15     system("chcp 1251");
16     system("cls");
17     printf("Введіть матрицю суміжності:\n");
18     for (int i = 0; i < SIZE; i++)
19     {
20         a[i][i] = 0;
21         for (int j = 0; j < SIZE; j++) {
22             scanf("%d", &temp);
23             a[i][j] = temp;
24         }
25     }
26
27     for (int i = 0; i < SIZE; i++)
28     {
29         d[i] = 10000;
30         v[i] = 1;
31     }
32     d[begin_index] = 0;
33     do {
34         minindex = 10000;
35         min = 10000;
36         for (int i = 0; i < SIZE; i++)
37         {
38             if ((v[i] == 1) && (d[i] < min))
39             {
40                 min = d[i];
41                 minindex = i;
42             }
43         }

```

```

43     }
44     if (minindex != 10000)
45     {
46         for (int i = 0; i < SIZE; i++)
47         {
48             if (a[minindex][i] > 0)
49             {
50                 temp = min + a[minindex][i];
51                 if (temp < d[i])
52                 {
53                     d[i] = temp;
54                 }
55             }
56         }
57         v[minindex] = 0;
58     }
59 } while (minindex < 10000);
60 printf("\nНайкоротші відстані до вершин: \n");
61 for (int i = 0; i < SIZE; i++)
62     printf("%5d", d[i]);
63 int ver[SIZE];
64 int end = 29;
65 ver[0] = end + 1;
66 int k = 1;
67 int weight = d[end];
68
69 while (end != begin_index)
70 {
71     for (int i = 0; i < SIZE; i++)
72         if (a[end][i] != 0)
73         {
74             int temp = weight - a[end][i];
75             if (temp == d[i])
76             {
77                 weight = temp;
78                 end = i;
79                 ver[k] = i + 1;
80                 k++;
81             }
82         }
83 }
84
85 printf("\nВивід найкоротшого шляху:");
86 for (int i = k - 1; i >= 0; i--)
87     printf("%3d ", ver[i]);
88 return 0;
89 }

```

Введіть матрицю суміжності:

```

0 6 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
6 0 1 0 0 0 0 8 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 1 0 1 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 3 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 3 0 3 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 3 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
4 0 0 0 0 0 0 2 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 8 0 0 0 0 2 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 3 0 0 0 0 1 0 4 0 0 0 0 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 4 0 2 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 5 0 0 0 0 2 0 4 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 7 0 0 0 0 4 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 5 0 0 0 0 0 0 7 0 0 0 0 0 5 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1 0 0 0 0 7 0 1 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 3 0 0 0 0 1 0 2 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 4 0 0 0 0 2 0 3 0 0 0 0 4 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 3 0 7 0 0 0 0 2 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 7 0 0 0 0 0 8 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 7 0 0 0 0 8 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 7 0 3 0 0 0 0 2 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 3 0 1 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 4 0 0 0 0 1 0 8 0 0 0 0 3 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 8 0 5 0 0 0 3 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 8 0 0 0 0 5 0 0 0 0 0 7 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 8 0 0 0 0 0 4 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 0 0 0 0 4 0 7 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 7 0 3 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 3 0 3 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 3 0 0 0 0 3 0 6 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 7 0 0 0 0 6 0 0 0 0

```

Найкоротші відстані до вершин:

```

0 6 7 8 11 14 4 6 7 9 11 15 9 7 8 10 12 19 14 12 9 10 14 19 18 14 10 13 16 22
Вивід найкоротшого шляху: 1 7 8 14 15 21 22 28 29 30

```

Висновок: на цій лабораторній роботі я здобув практичні вміння та навички з використання алгоритму Дейкстри.