

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ
“ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Лабораторна робота №5
з дисципліни «Дискретна математика»

Виконав:

студент групи КН-114

Кміть Христина

Викладач:

Мельникова Н.І.

Львів – 2019р

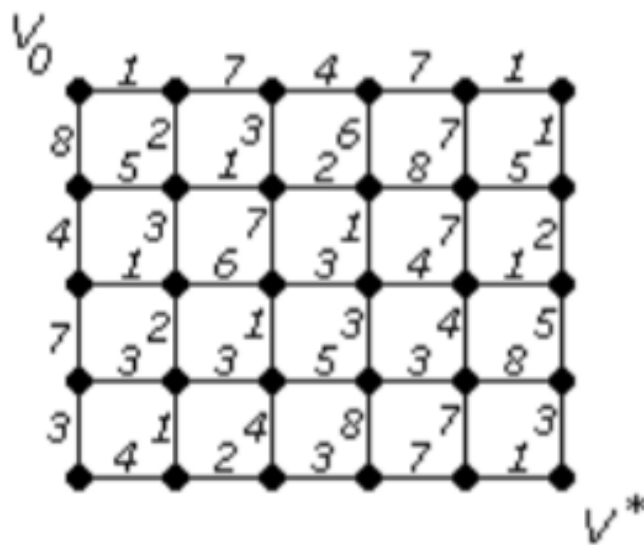
Тема: Знаходження найкоротшого маршруту за алгоритмом Дейкстри. Плоскі планарні граfi

Мета набуття практичних вмінь та навичок з використання алгоритму Дейкстри

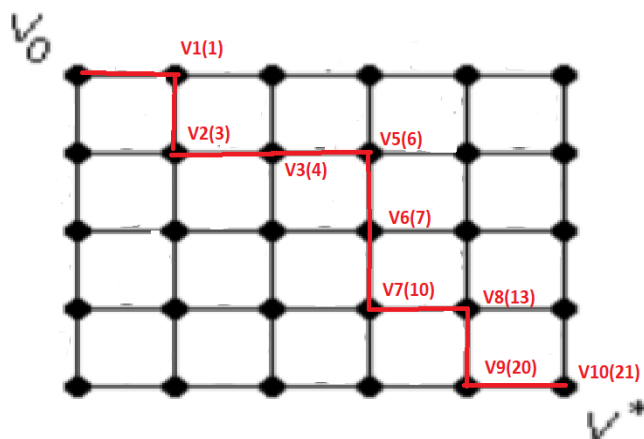
Завдання і розв'язок 1 додатку 11 варіанту:

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

11

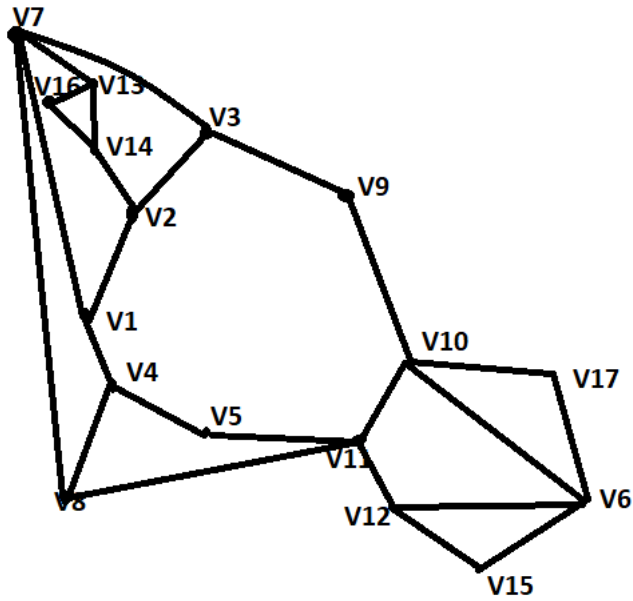
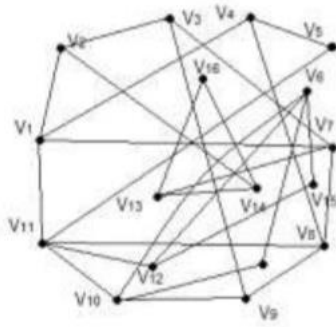


11



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

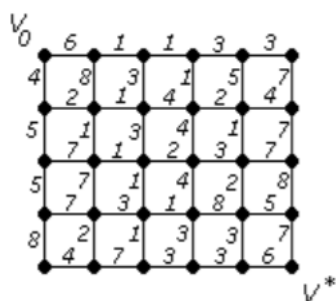
11



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



```
ConsoleApplication5 (Global Scope)
1  #include "pch.h"
2  #define _CRT_SECURE_NO_WARNINGS
3  #include <stdio.h>
4  #include <stdlib.h>
5  #define SIZE 30
6  int main()
7  {
8      int a[SIZE][SIZE]; // матриця зв'язків
9      int d[SIZE]; // мінімальна відстань
10     int v[SIZE]; // відвідані вершини
11     int temp, minindex, min;
12     int begin_index = 0;
13     system("chcp 1251");
14     system("cls");
15     // Ініціалізація матриці зв'язків
16     for (int i = 0; i < SIZE; i++)
17     {
18         a[i][i] = 0;
19         for (int j = i + 1; j < SIZE; j++) {
20             printf("Введіть відстань [%d-%d] : ", i + 1, j + 1);
21             scanf("%d", &temp);
22             a[i][j] = temp;
23             a[j][i] = temp;
24         }
25     }
26     // Вивід матриці зв'язків
27     for (int i = 0; i < SIZE; i++)
28     {
29         for (int j = 0; j < SIZE; j++)
30             printf("%5d ", a[i][j]);
```

```
printf("%5d ", a[i][j]);
printf("\n");
}
// Ініціалізація вершин і відстаней
for (int i = 0; i < SIZE; i++)
{
    d[i] = 10000;
    v[i] = 1;
}
d[begin_index] = 0;
do {
    minindex = 10000;
    min = 10000;
    for (int i = 0; i < SIZE; i++)
    { // Якщо вершину ще не оминули й вага менше min
        if ((v[i] == 1) && (d[i] < min))
        {
            min = d[i];
            minindex = i;
        }
    }
    // Додаємо знайдений мінімальна вага до поточного в
    if (minindex != 10000)
    {
        for (int i = 0; i < SIZE; i++)
        {
            if (a[minindex][i] > 0)
            {
                temp = min + a[minindex][i];
                if (temp < d[i])
```

```
ConsoleApplication5.cpp  X
ConsoleApplication5  (Global Scope)

59         if (temp < d[i])
60         {
61             d[i] = temp;
62         }
63     }
64 }
65 v[minindex] = 0;
66 }
67 } while (minindex < 10000);
68 // Висновок найкоротших відстаней до вершин
69 printf("\nНайкоротші відстані до вершин: \n");
70 for (int i = 0; i < SIZE; i++)
71     printf("%5d ", d[i]);
72
73 // Відновлення шляху
74 int ver[SIZE]; // масив відвіданих вершин
75 int end = 28; // індекс кінцевої вершини = 5 - 1
76 ver[0] = end + 1; // початковий елемент - кінцева вершина
77 int k = 1; // індекс попередньої вершини
78 int weight = d[end]; // вага кінцевої вершини
79
80 while (end != begin_index) // поки не дійшли до початкової вершини
81 {
82     for (int i = 0; i < SIZE; i++) // переглядаємо всі вершини
83     if (a[end][i] != 0) // якщо зв'язок є
84     {
85         int temp = weight - a[end][i]; // визначаємо вагу шляху з попередньої вершини
86         if (temp == d[i]) // якщо вага припала на розрахованим
87         { // значить з цієї вершини і був перехід
88             weight = temp; // зберігаємо новий вага
```

```
//
78     int k = 1; // індекс попередньої вершини
79     int weight = d[end]; // вага кінцевої вершини
80
81     while (end != begin_index) // поки не дійшли
82     {
83         for (int i = 0; i < SIZE; i++) // перегля
84         if (a[end][i] != 0) // якщо зв'язок
85         {
86             int temp = weight - a[end][i]; //
87             if (temp == d[i]) // якщо вага пр
88             { // значить з ці
89                 weight = temp; // зберігаємо
90                 end = i; // зберігаємо
91                 ver[k] = i + 1; // і записуєм
92                 k++;
93             }
94         }
95     }
96     // Вивід шляху (початкова вершина виявилася є
97     printf("\nВивід найкоротшого шляху\n");
98     for (int i = k - 1; i >= 0; i--)
99         printf("%3d ", ver[i]);
100     getchar(); getchar();
101     return 0;
102 }
```

Реалізація програми :

```
C:\Users\Admin\Documents\Visual Studio 2017\Projects\ConsoleApplication5\Debug\ConsoleApplication5.exe

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 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 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 0 0 0 0
0 0 1 0 3 0 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 0 0
0 0 0 3 0 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 7 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
4 0 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 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 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 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 0 0
0 0 0 0 5 0 0 0 0 0 2 0 4 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 0 0 7 0 0 0 0 4 0 0 0 0 0 7 0
0 0 0 0 0 0 0 0 5 0 0 0 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 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 7 0 1 0 7
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

```
C:\Users\Admin\Documents\Visual Studio 2017\Projects\ConsoleApplication5\Debug\ConsoleApplication5.exe

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

```
C:\Users\Admin\Documents\Visual Studio 2017\Projects\ConsoleApplication5\Debug\ConsoleApplication5.exe

0 0 5 0 0 0 0 0 0 7 0 0 0 0 0 0 0 8 0
0 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 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
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 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 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 0
0 0 0 7 0 0 0 0 6 0 0 0 0 0 0 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 21 14 12
9 10 14 19 18 14 10 13 16 22

Вивід найкоротшого шляху
1 7 8 14 15 21 22 28 29
```

Висновок :

Завдяки цій лабораторній роботі я набула практичні вміння та навички з використання алгоритму Дейкстри