

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА ”
Кафедра систем штучного інтелекту

Звіт

Лабораторна робота №5

З дисципліни:

Дискретна математика

Виконав

Студент групи КН-113

Омелюх Роман

Викладач:

Мельникова Н.І.

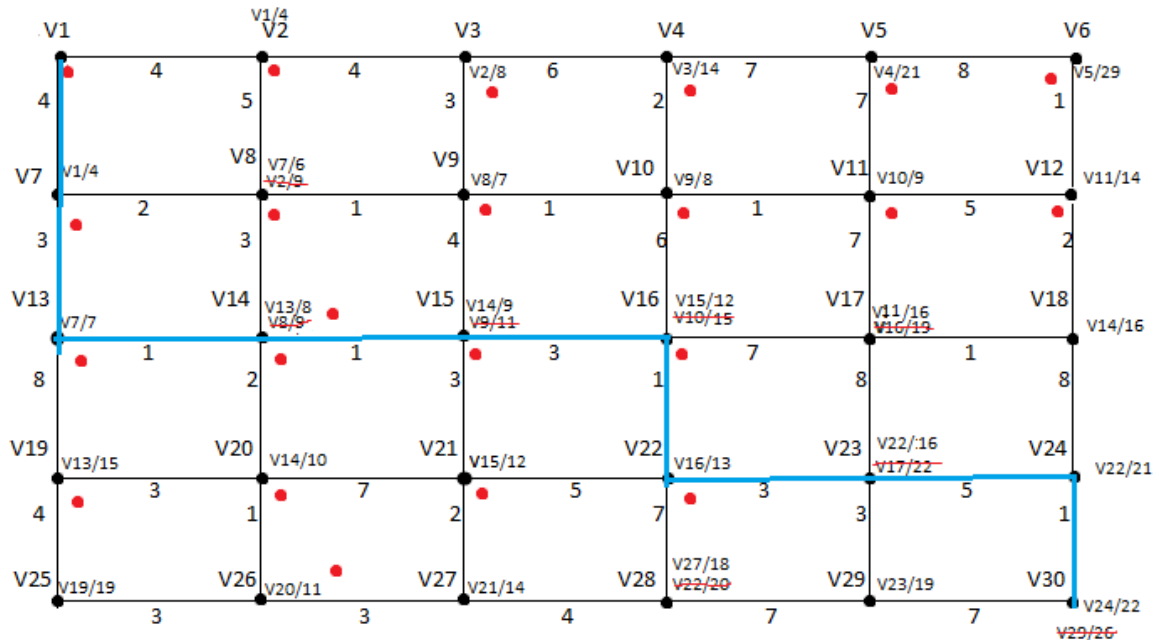
Тема: Знаходження найкоротшого маршруту за алгоритмом Дейкстри. Плоскі планарні графи

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

Варіант №6

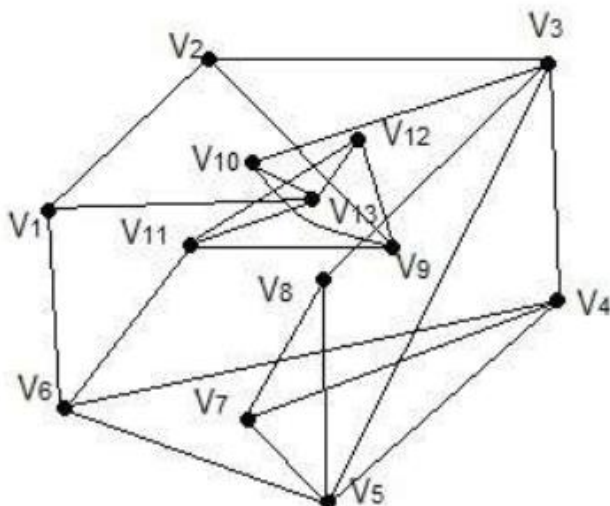
Завдання № 1. Розв'язати на графах наступні 2 задачі:

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

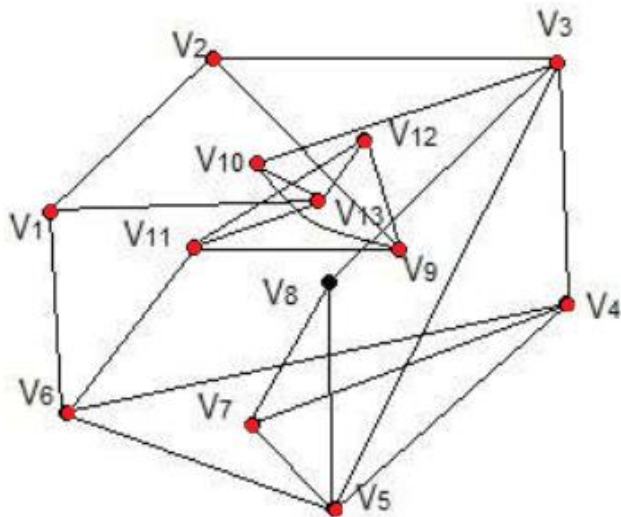


$V_{min}=22$;

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



Необхідна умова: Граф непланарний, то він повинен містити більше 4 вершин, степінь яких 3, або більше 5 вершин степінь яких більше 2.



*Червоним позначено вершини степінь яких більше 2. Загалом їх є 12 тому, зробити укладку графа у площині не можлив

```

1  #include<iostream>
2  using namespace std;
3  int n,i, j, q,dist[40],pred[40],c[40][40];
4  bool visited[40];
5  int minDist()
6  {
7      int minimum = 9999, minD;
8      for (int v=0; v<n; v++)
9          if (!visited[v] && dist[v]<= minimum)
10             {
11                 minimum=dist[v];
12                 minD=v;
13             }
14      return minD;
15  }
16  void printPath(int j)
17  {
18      if (pred[j]==-1)
19          return;
20      printPath(pred[j]);
21      cout<<"v"<<j+1<<" ";
22  }
23  void dijkstra(int c[40][40])
24  {
25      int point;
26      cout << "Enter start point : ";
27      cin >> point;
28      for (int i = 0; i < n; i++)
29      {
30          pred[i]=-1;

```

```

30         pred[0]=-1;
31         dist[i]=9999;
32         visited[i]=false;
33     }
34     dist[point-1]=0;
35     for (int i=0;i<n - 1;i++)
36     {
37         int u = minDist();
38         visited[u] = true;
39         for (int v=0; v<n; v++)
40             if (!visited[v] && c[u][v] &&
41                 dist[u]+c[u][v] < dist[v])
42             {
43                 pred[v]=u;
44                 q++;
45                 dist[v]=dist[u]+c[u][v];
46             }
47     }
48     cout << "The least way is: ";
49     cout << dist[29] << endl;
50     cout << "The way is: ";
51     cout << "V1 ";
52     printPath(29);
53     cout << endl;
54 }
55 int main()
56 {
57
58     int g1, g2;
59     cout << "Enter the number of vertices: ";
60
61     int g1, g2;
62     cout << "Enter the number of vertices: ";
63     cin >> n;
64     for (int i = 0; i < n; i++) {
65         for (int j = 0; j < n; j++)
66         {
67             c[i][j] = 0;
68         }
69     }
70     g1=6;
71     g2=5;
72     for (i = 0; i < n; i++) {
73         for (j = i + 1; j < n; j++)
74         {
75             if (j == i + 1 || j == i + g1) {
76                 cout << "From " << i+1 << " to " << j+1 << ": ";
77                 cin >> c[i][j];
78             }
79             else {
80                 c[i][j] = 0;
81             }
82         }
83     }
84     dijkstra(c);
85     return 0;
86 }

```

```
"E:\яЁюуЁрьу\5 ырсп эрярь ыртрэфюё\bin\Debug\5 ырсп эрярь ыг
From 16 to 17: 7
From 16 to 22: 3
From 17 to 18: 7
From 17 to 23: 4
From 18 to 19: 1000
From 18 to 24: 5
From 19 to 20: 8
From 19 to 25: 2
From 20 to 21: 3
From 20 to 26: 2
From 21 to 22: 1
From 21 to 27: 1
From 22 to 23: 1
From 22 to 28: 1
From 23 to 24: 5
From 23 to 29: 3
From 24 to 25: 1000
From 24 to 30: 8
From 25 to 26: 1
From 26 to 27: 7
From 27 to 28: 3
From 28 to 29: 6
From 29 to 30: 3
Enter start point : 1
The least way is: 20
The way is: V1 V7 V8 V9 V10 V11 V17 V23 V29 V30
```

о.

Висновок: на цій лабораторній я опанував алгоритм Дейкстри та реалізував його програмно.