

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Лабораторна робота № 5

з дисципліни

«Дискретна математика»

Виконала:

студентка групи КН-112

Сидір Олена Юріївна

Викладач:

Мельникова Н.І.

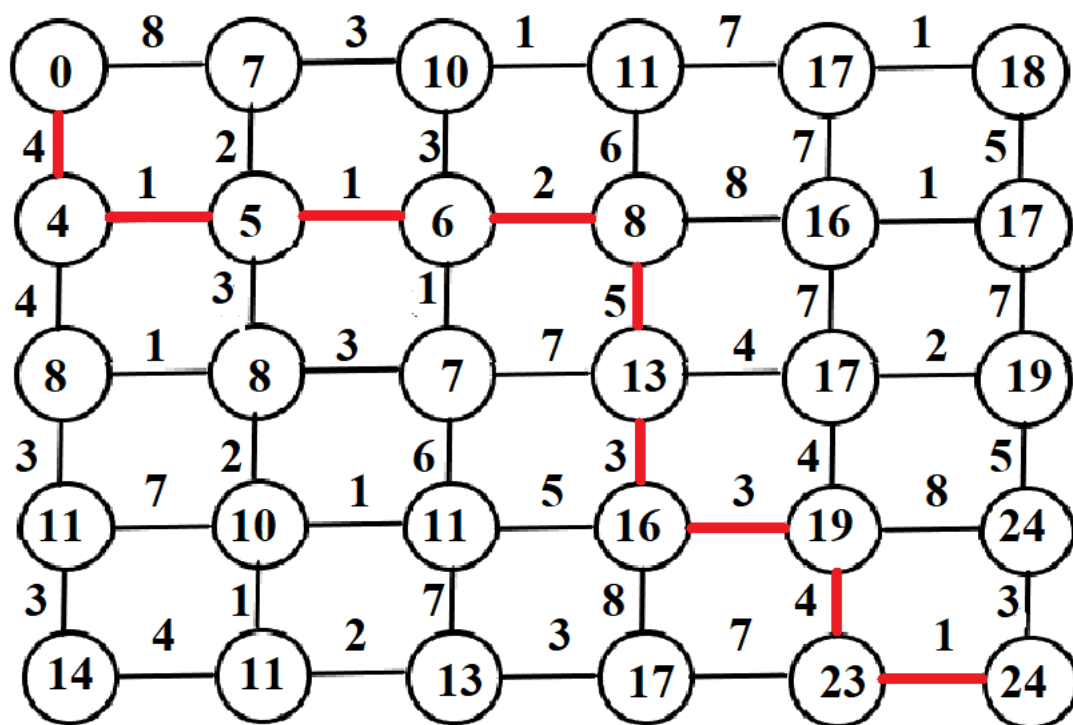
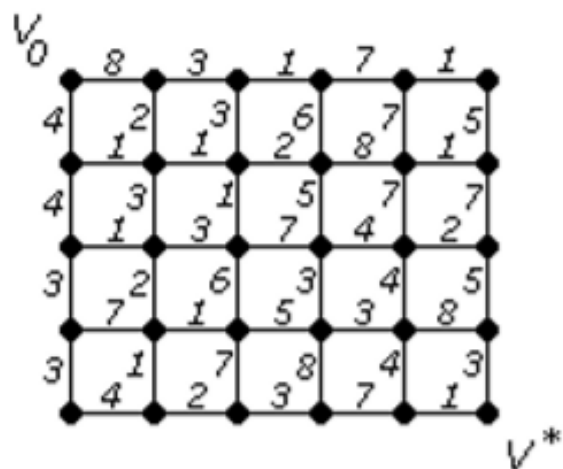
Львів – 2019 р.

Варіант №13

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

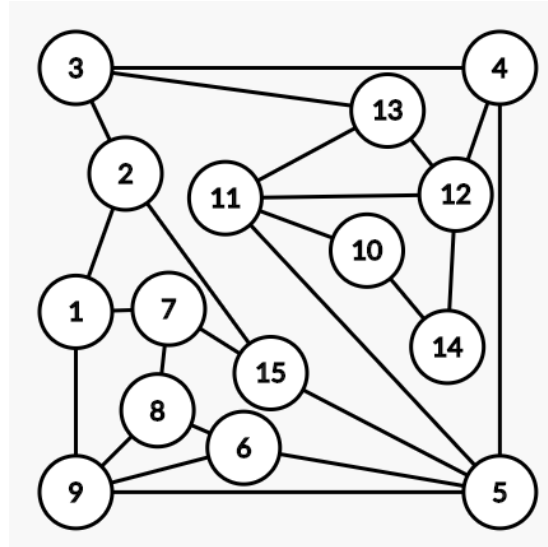
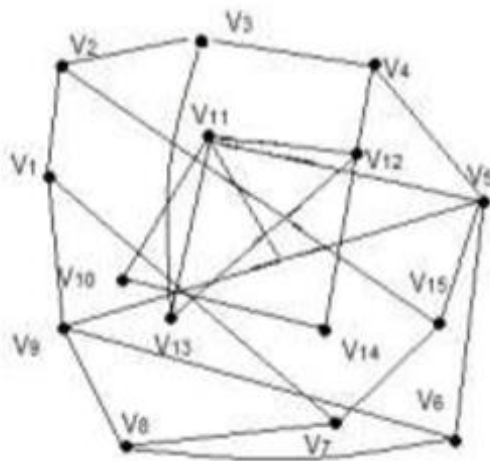
1. За допомогою алгоритму Дейкстри знайти найкоротший шлях у графі поміж парою вершин V_0 і V^* .

13

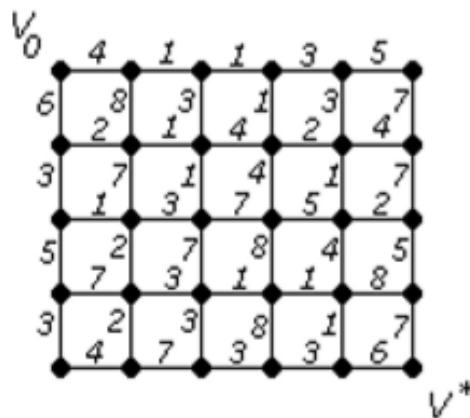


2. За допомогою γ -алгоритма зробити укладку графа у площині, або довести що вона неможлива.

13



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



```
#include <iostream>
#include <stdio.h>
#include <stdlib.h>
#define SIZE 30

#pragma warning(disable : 4996);
using namespace std;
int main()
{
    int a[SIZE][SIZE];
    int d[SIZE];
    int v[SIZE];
    int t, minindex, min;
    int first = 0;
    cout << "enter matrix : " << endl;
    for (int i = 0; i < SIZE; i++) {
        for (int j = 0; j < SIZE; j++) {
            cin >> a[i][j];
        }
    }
    for (int i = 0; i < SIZE; i++)
    {
        d[i] = 10000;
        v[i] = 1;
    }
}
```

```

}
d[first] = 0;
do {
    minindex = 10000;
    min = 10000;
    for (int i = 0; i < SIZE; i++)
    {
        if ((v[i] == 1) && (d[i] < min))
        {
            min = d[i];
            minindex = i;
        }
    }
    if (minindex != 10000)
    {
        for (int i = 0; i < SIZE; i++)
        {
            if (a[minindex][i] > 0)
            {
                t = min + a[minindex][i];
                if (t < d[i])
                {
                    d[i] = t;
                }
            }
        }
        v[minindex] = 0;
    }
} while (minindex < 10000);
cout << "the shortest distance to the vertexes: " << endl;
for (int i = 0; i < SIZE; i++)
    cout << d[i] << " ";
int ver[SIZE];
int end = 29;
ver[0] = end + 1;
int k = 1;
int weight = d[end];

while (end != first)
{
    for (int i = 0; i < SIZE; i++)
        if (a[end][i] != 0)
        {
            int t = weight - a[end][i];
            if (t == d[i])
            {
                weight = t;
                end = i;
                ver[k] = i + 1;
                k++;
            }
        }
}
cout << endl;
cout << "the shortest way:" << endl;
for (int i = k - 1; i >= 0; i--)
    cout << " -> " << "v" << ver[i];
return 0;
}
/*

```

```

0 4 0 0 0 0 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
4 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 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 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 3 0 5 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 5 0 0 0 0 0 0 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

