

**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

Садовнік Ілля

Викладач:

Мельникова Н.І.

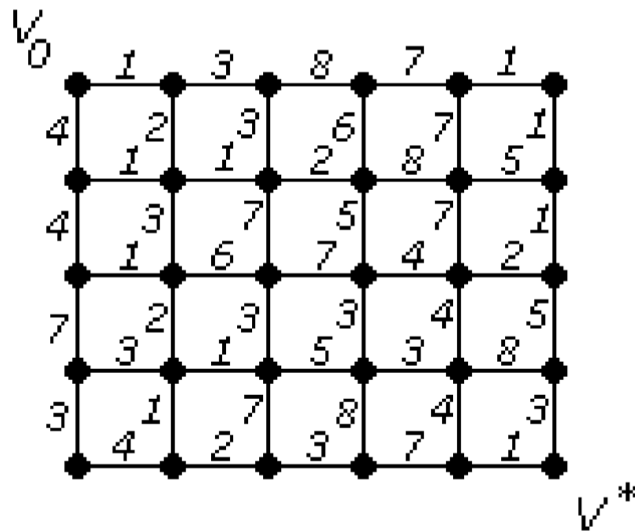
Лабораторна робота № 5.

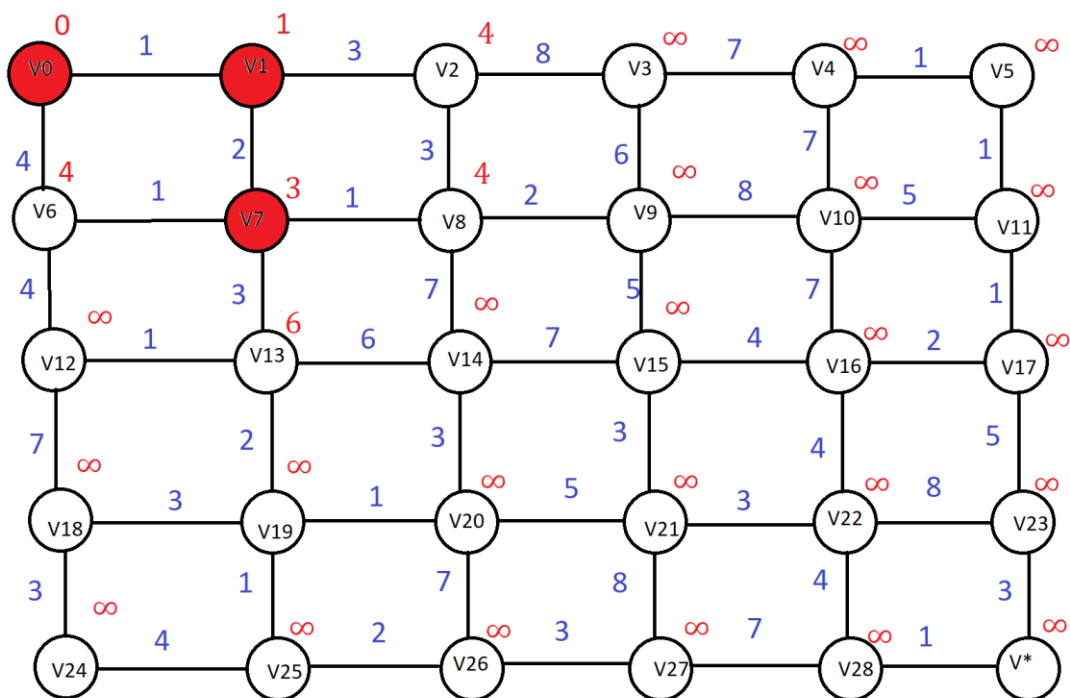
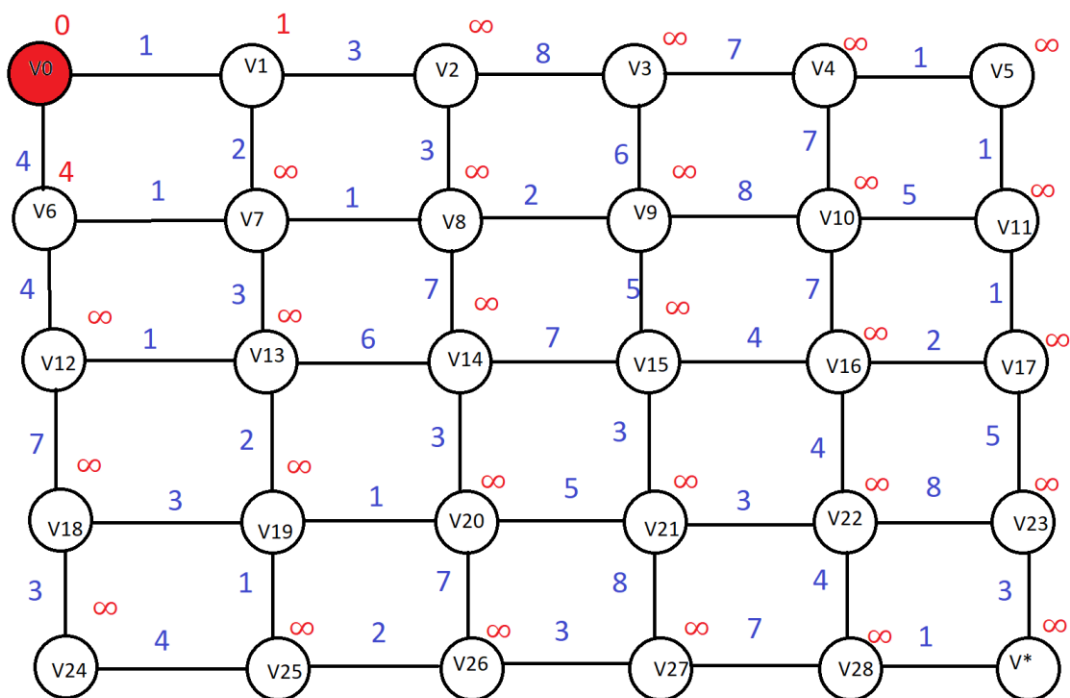
Тема: Знаходження найкоротшого маршруту за алгоритмом Дейкстри. Плоскі планарні граfi

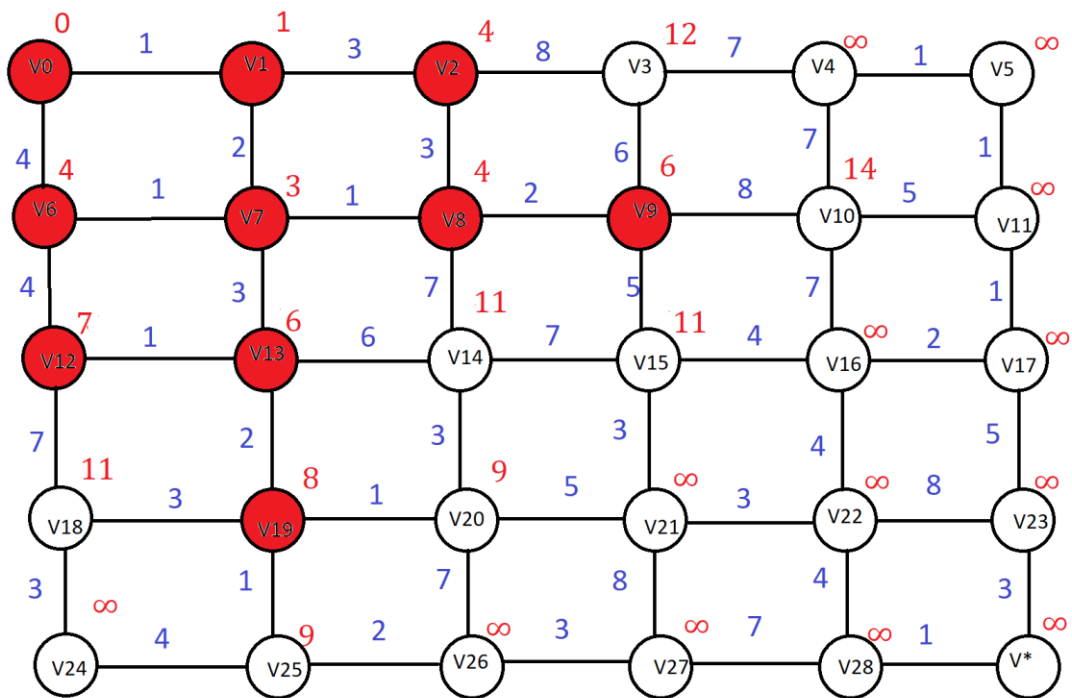
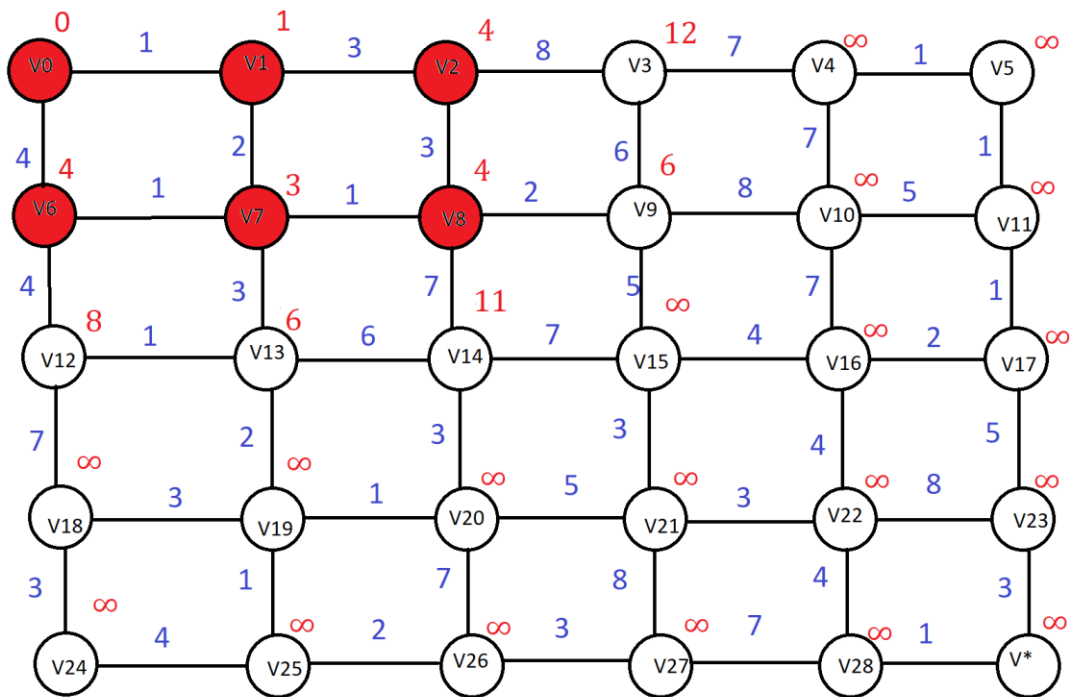
Мета роботи: набуття практичних вмінь та навичок з використання алгоритму Дейкстри.

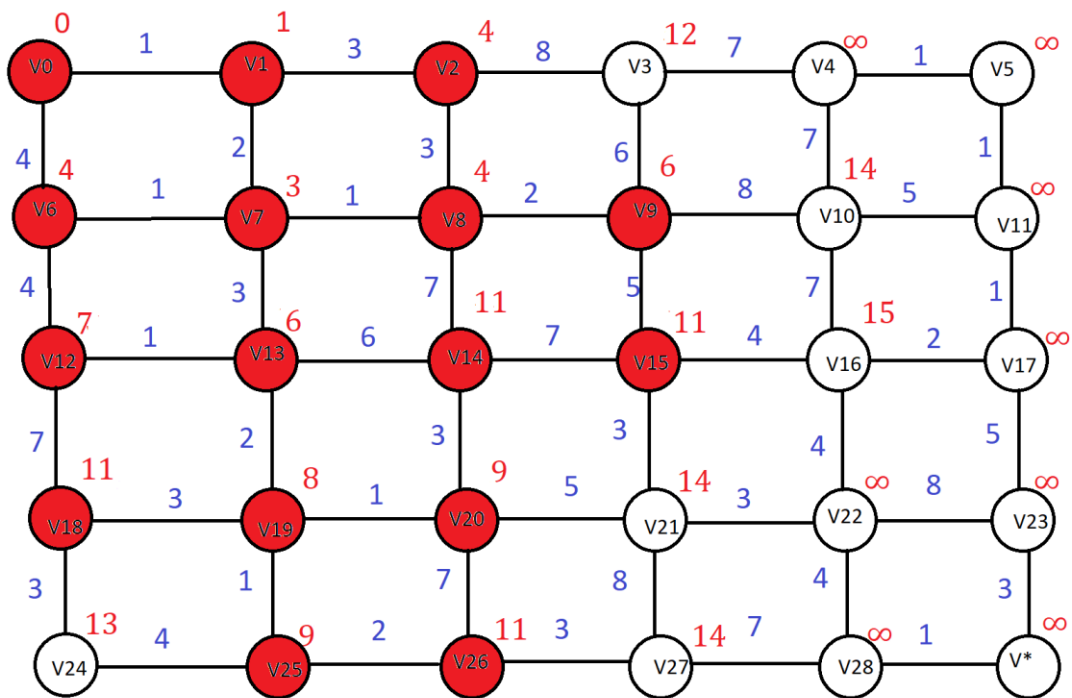
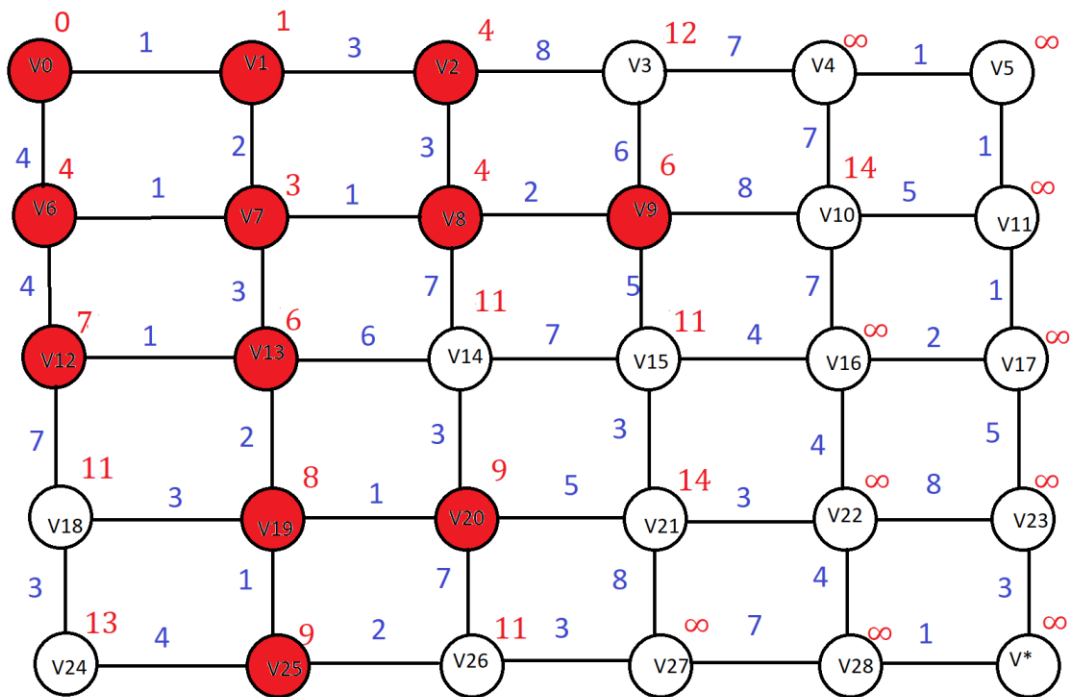
Варіант 12.

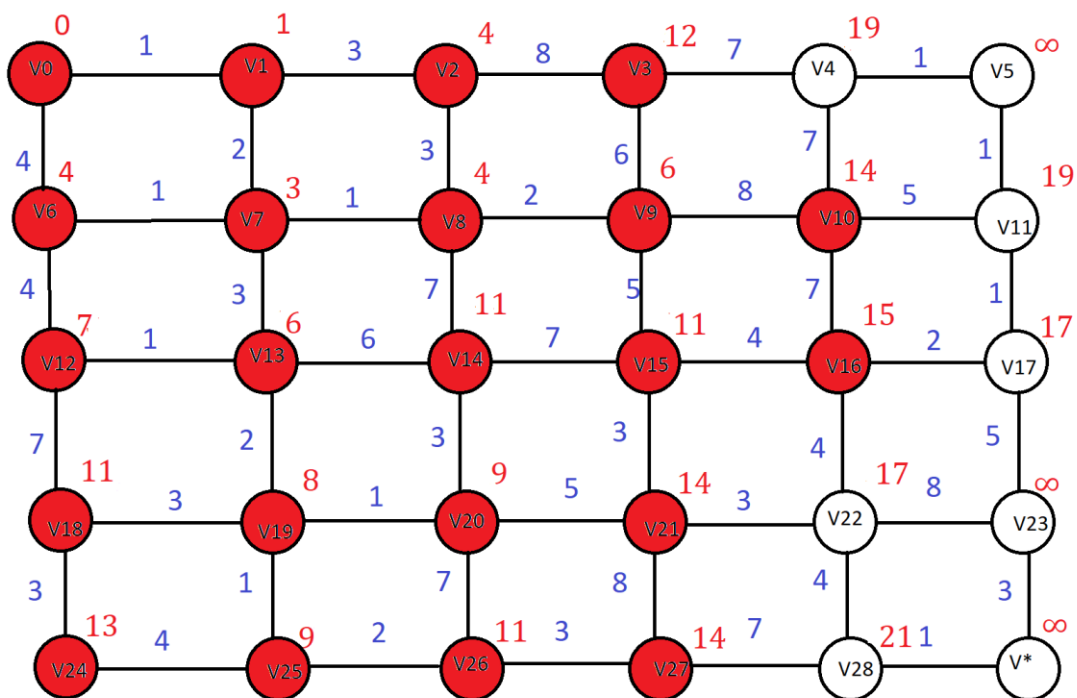
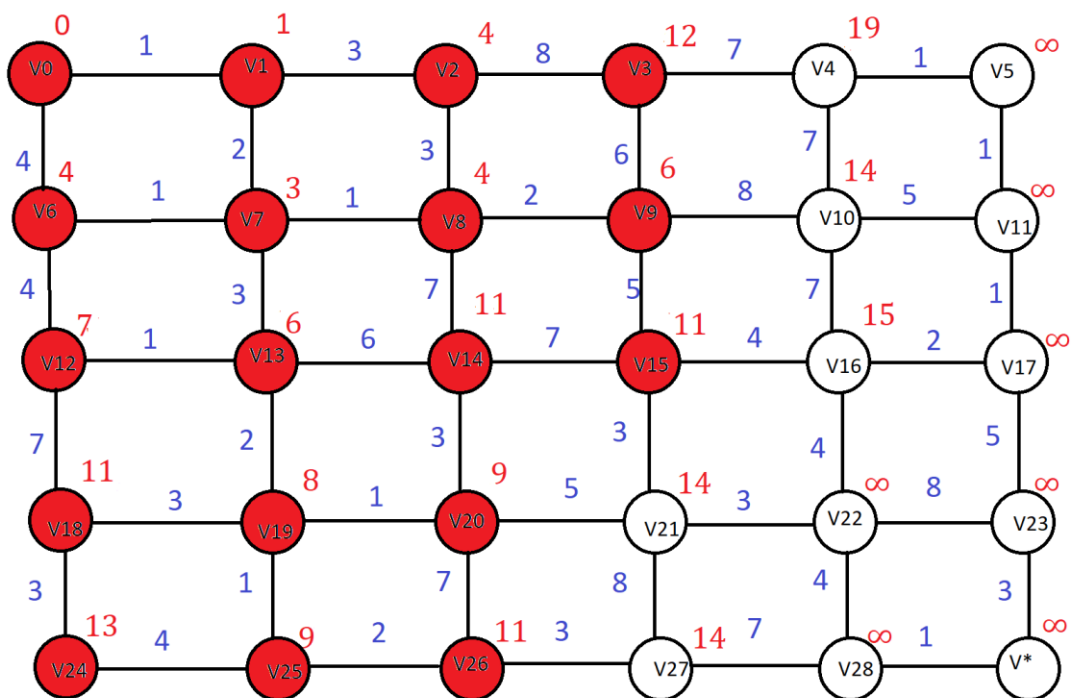
1. За допомогою алгоритму Дейкстра знайти найкоротший шлях у графі поміж парою вершин V_0 і V^* .

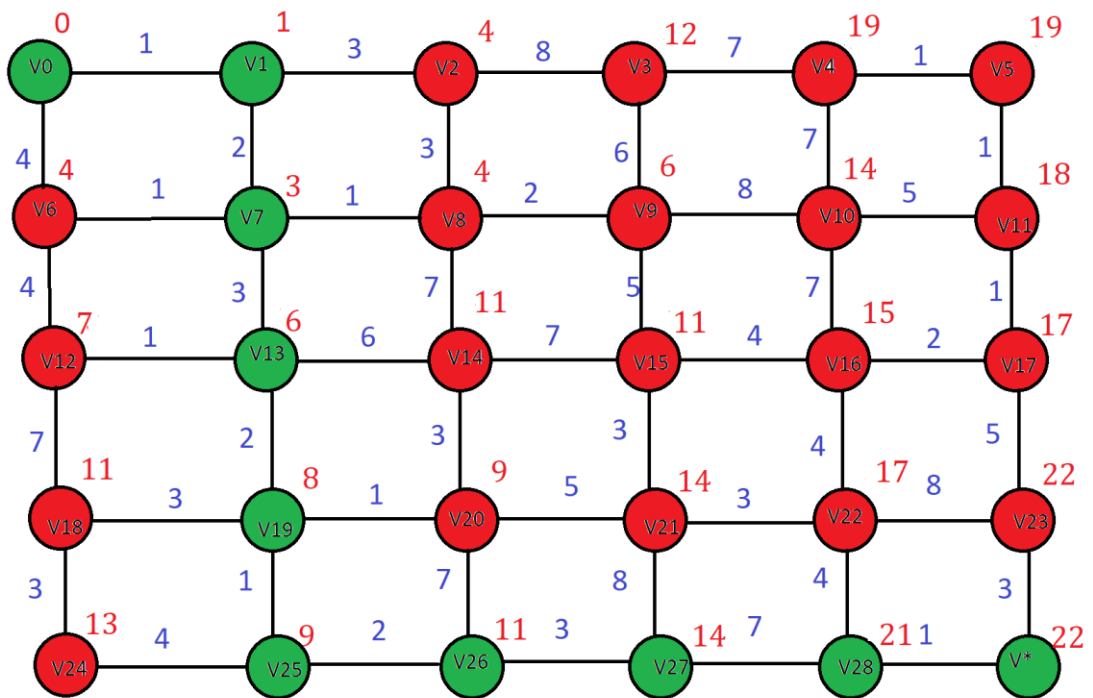
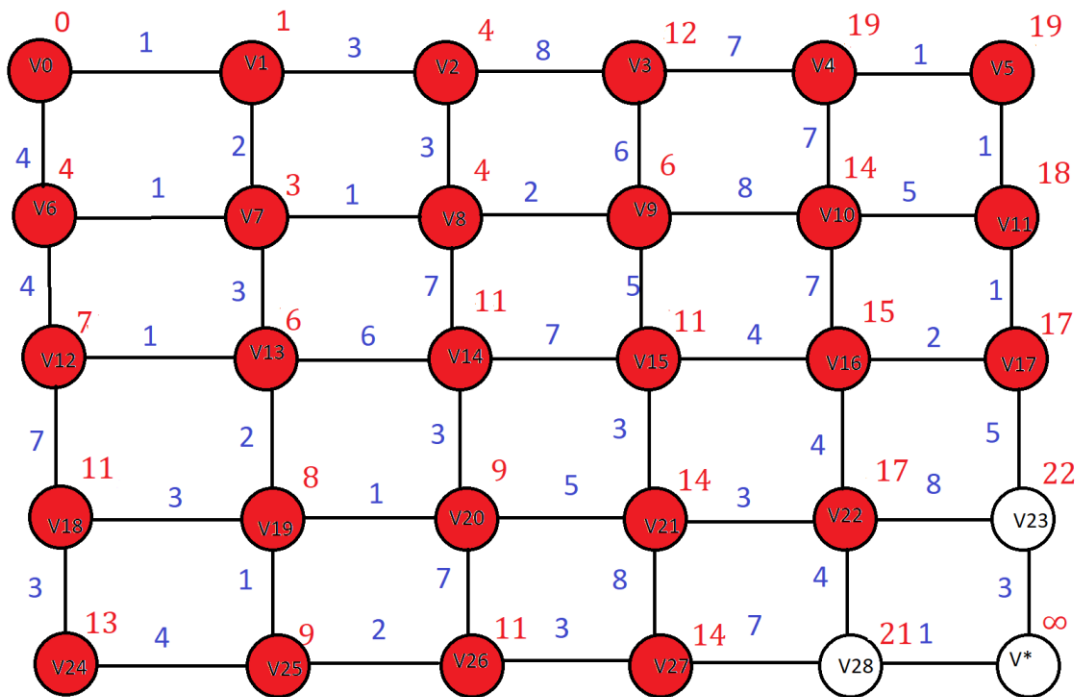






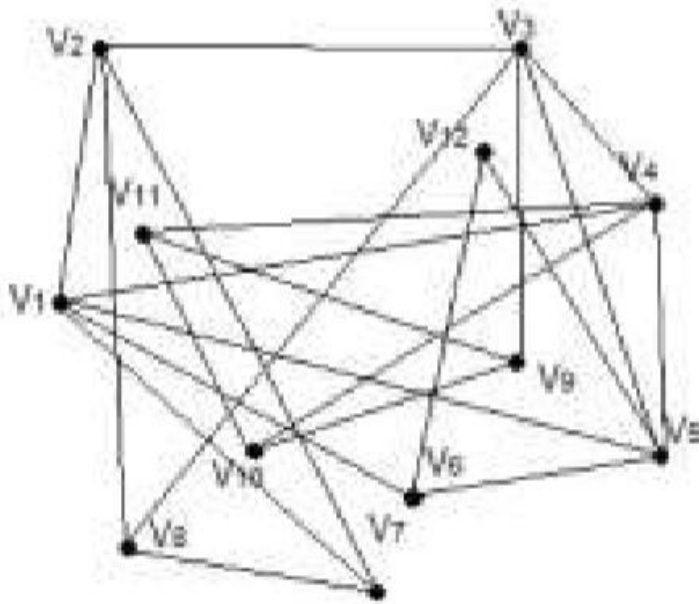


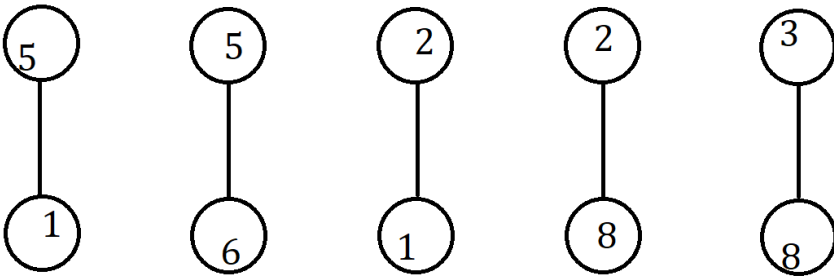
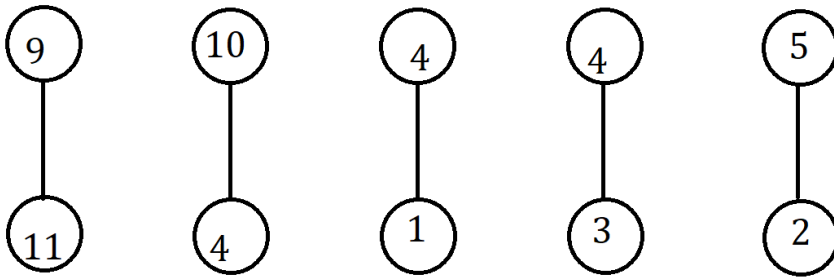
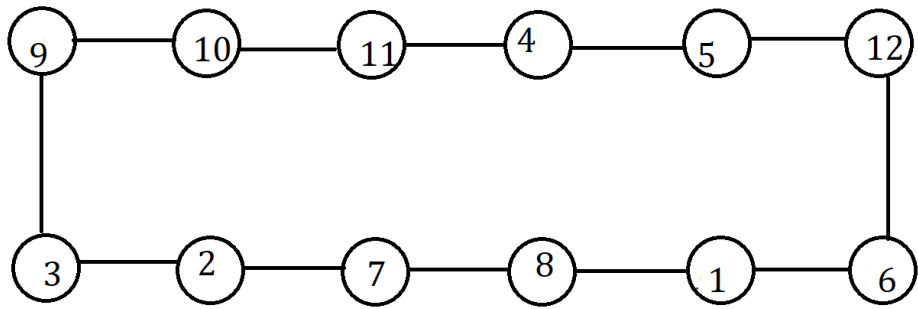


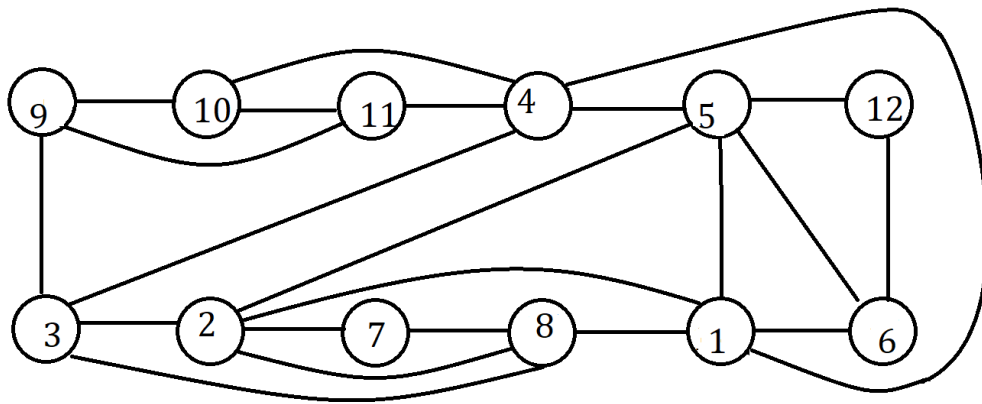


2. За допомогою Y-алгоритма зробити укладку графа у площині, або довести що вона неможлива.

12







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"
#define ver 30
using namespace std;

void dikstra(int matrix[][ver])
{
    int masivproidenihversin[ver];
    int masivvagivershin[ver];
    int masivrad[ver];
    masivvagivershin[0] = 1000;
    for (int i = 1; i < ver; i++)
    {
        masivvagivershin[i] = 99999;
    }
    for (int i = 0; i < ver; i++)
    {
        masivproidenihversin[i] = 0;
    }
    for (int i = 0; i < ver; i++)
    {
        masivrad[i] = 99;
    }
    int min, mini;
    for (int i = 0; i < ver; i++)
    {
        min = 2000;
        if (i == 0)
        {
            for (int j = 0; j < ver; j++)
            {
                if (matrix[i][j] != INT_MAX)
                {
                    masivrad[j] = i;
                }
            }
        }
    }
}
```

```

        masivvagivershin[j] = matrix[i][j];
    }
    }
    masivproidenihversin[i] = 1;
}
for (int m = 0; m < ver; m++)
{
    if (masivvagivershin[m] < min && masivproidenihversin[m]!=1)
    {
        min = masivvagivershin[m];
        mini = m;
    }
}

for (int j = 0; j < ver; j++)
{
    if (matrix[mini][j] != INT_MAX && ((masivvagivershin[mini]+
matrix[mini][j])< masivvagivershin[j]) )
    {
        masivrad[j] = mini;
        masivvagivershin[j] = masivvagivershin[mini] + matrix[mini][j];
    }
}
masivproidenihversin[mini] = 1;

}

cout << "Shlah:" << endl;
int printer = 29;
for (int i = 0; i < 8; i++)
{
    cout << masivrad[printer]<<" ";
    printer = masivrad[printer];
}
cout<< endl;
cout << "Vaga:" << endl;
cout << masivvagivershin[29];
}
void main()
{
    int matrix[][ver] = {
{INT_MAX,6,INT_MAX,INT_MAX,INT_MAX,INT_MAX,4,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,
INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT
_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX},
{6,INT_MAX,1,INT_MAX,INT_MAX,INT_MAX,INT_MAX,2,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_M
AX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,I
NT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX},
{INT_MAX,1, INT_MAX, 1, INT_MAX, INT_MAX, INT_MAX,
INT_MAX,3, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX,
INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX, INT_MAX,
INT_MAX,INT_MAX,INT_MAX,INT_MAX},
{INT_MAX,INT_MAX,1,INT_MAX,3,INT_MAX,INT_MAX,INT_MAX,INT_MAX,4,INT_MAX,INT_MAX,INT_MA
X,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,IN
T_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX},
{INT_MAX,INT_MAX,INT_MAX,3,INT_MAX,3,INT_MAX,INT_MAX,INT_MAX,INT_MAX,5,INT_MAX,INT_MA
X,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,IN
T_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX,INT_MAX},

```