

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ
“ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

Звіт

Лабораторна робота №5

з дисципліни «Дискретна математика»

Виконав: студент групи КН-113

Власюк Олександр

Викладач: Мельникова Н.І.

Львів – 2019 р.

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

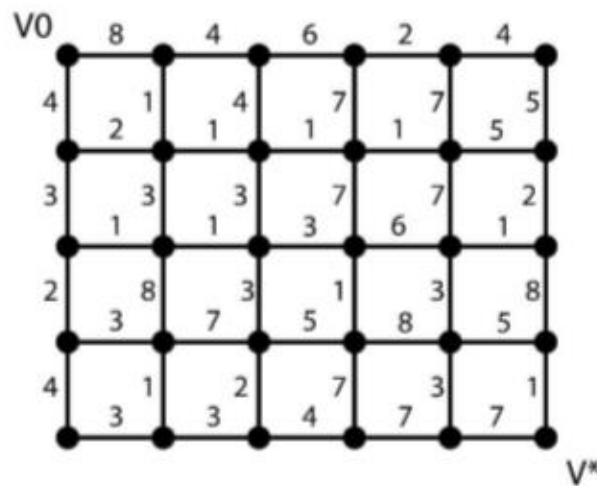
Мета. Набуття практичних вмінь та навичок з використання алгоритму Дейкстри.

Постановка завдання :

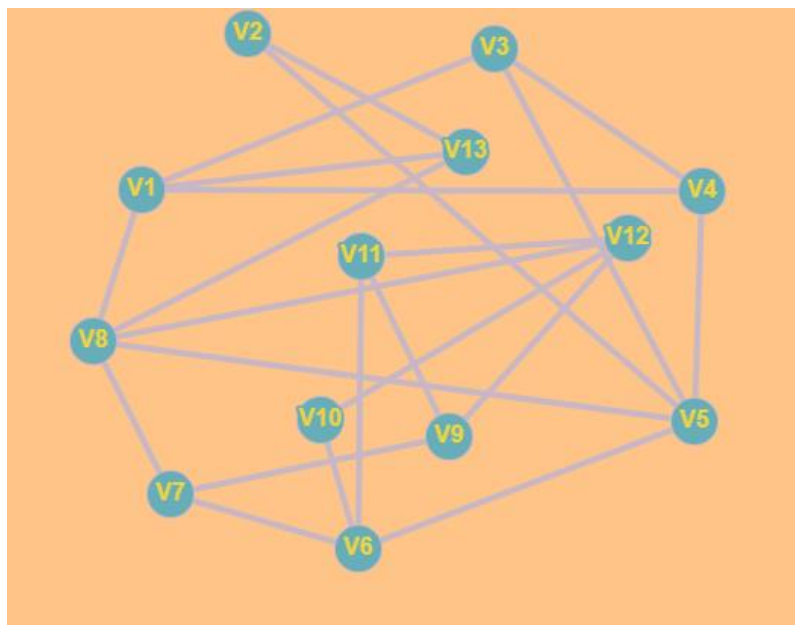
5-Варіант

Розв'язати на графах наступні 2 задачі:

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

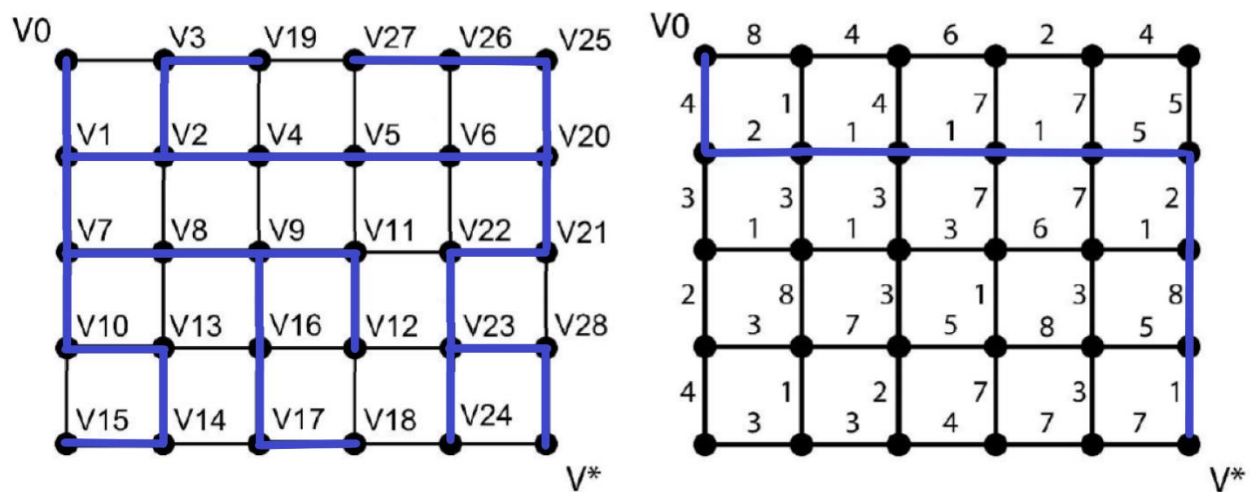


2. За допомогою γ -алгоритма зробити укладку графа у площині, або довести що вона неможлива.

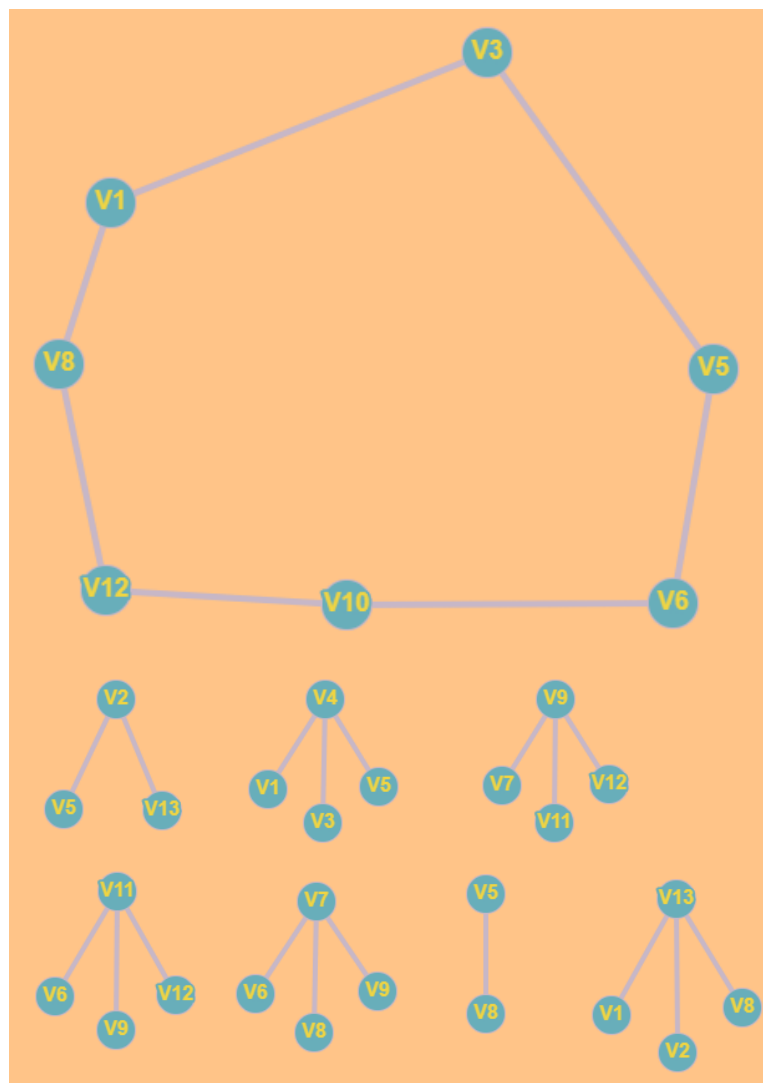


Розв'язки:

1)



2)



Код програмної реалізації:

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int n,m;
```

```
    cout<<"Skilky vershyn hochete?";
```

```
    cin>>n;
```

```
    cout<<"Skilky reber hochete? ";
```

```
    cin>>m;
```

```
    cout<<endl;
```

```
    int s=n;
```

```
    int a[s][s];
```

```
    for (int i=0; i<n; i++)
```

```
    {
```

```
        for (int j=0; j<n; j++)
```

```
        {
```

```
            a[i][j]=0;
```

```
        }
```

```
    }
```

```
    //вводимо потрібні ребра
```

```
    int t1,t2,w;
```

```
    cout<<"Enter rebra(1 vershyna | 2 vershyna | waga):\n";
```

```
    for (int k=0;k<m;k++)
```

```

{
    cout<<"Rebro"<<k+1<<": ";
    cin>>t1>>t2>>w;
    while (t1<0 || t1>n || t2<0 || t2>n || w<0)
    {
        cout<<"The data entered incorrectly."<<endl;
        cout<<"Try again please."<<endl;
        cin>>t1>>t2>>w;
    }
    a[t1-1][t2-1]=w;
    a[t2-1][t1-1]=w;
}
cout<<endl;

```

```

int sh[n];//Мін. шляхи
int v[n];//пройдені вершини
int temp, index, minn;
for (int i = 0; i<n; i++)
{
    sh[i] = 300000;
    v[i] = 1;
}
sh[0] = 0;

do
{
    index = 300000;
    minn = 300000;
    for (int i = 0; i<n; i++)
    {

```

```

//проходимо всі вершини

//якщо вершина не пройдена і вага менша 300000

if ((v[i] == 1) && (sh[i]<minn))

{
    //переприсвоюємо значення

    minn = sh[i];

    index = i;

}

}

//додаємо знайдену мінімальну вагу і порівнюємо

if (index != 300000)

{
    for (int i = 0; i<n; i++)

    {
        if (a[index][i] > 0)

        {
            temp = minn + a[index][i];

            if (temp < sh[i])

            {
                sh[i] = temp;

            }

        }

    }

    v[index] = 0;

}

} while (index < 300000);

cout<<"korotshyi shliah z V1 do V"<<n<<" E: "<<sh[n-1]<<endl;

```

```

//пошук шляху

int vvv[n];

int kin = n-1;

vvv[0] = kin + 1;

int k = 1; //індекс попередньої вершини

int weight = sh[kin];

while (kin != 0) //поки не дійдемо до поч. вершини
{
    for (int i = 0; i < n; i++)

        if (a[kin][i] != 0) //перевірка чи є таке ребро
        {
            int temp = weight - a[kin][i]; //шукаємо вагу попередньої вершини

            if (temp == sh[i])
            {
                weight = temp;

                kin = i;

                vvv[k] = i + 1;

                k++;
            }
        }
}

//вивід шляху

cout << "\nkorotshyi shliach:\n";

for (int i = k - 1; i >= 0; i--)
{
    cout << vvv[i] << " ";
}

cout << endl;

```



```
return 0;  
}
```

Результат виконаної програми:

Результати співпадають.

Висновок: Протягом виконання лабораторної роботи я попрактикувався у використанні алгоритму Дейкстри.