

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА
ПОЛІТЕХНІКА»

Інститут комп'ютерних наук та інформаційних технологій Кафедра систем
штучного інтелекту

Лабораторна робота №5 з курсу
“Дискретна математика”

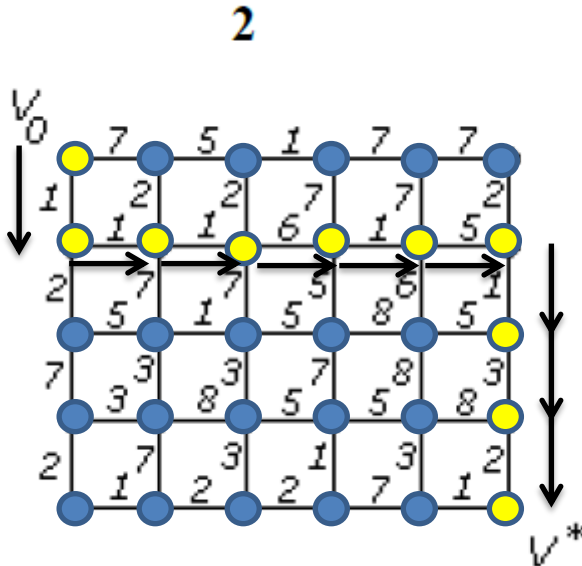
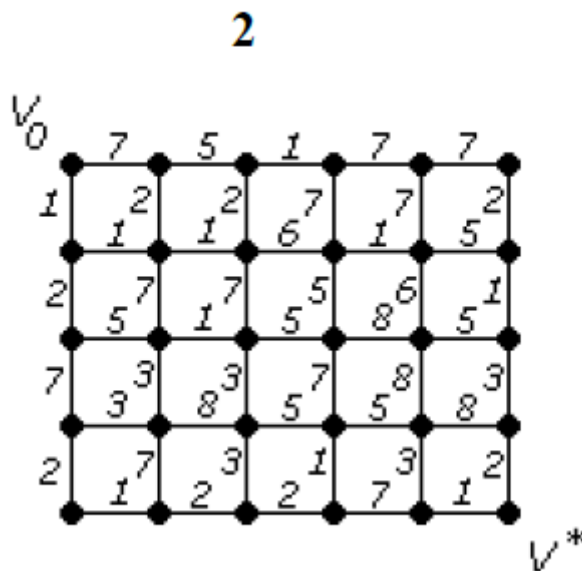
Виконав: ст. гр. КН-113

Байдич Володимир

Викладач: Мельникова Н.І.

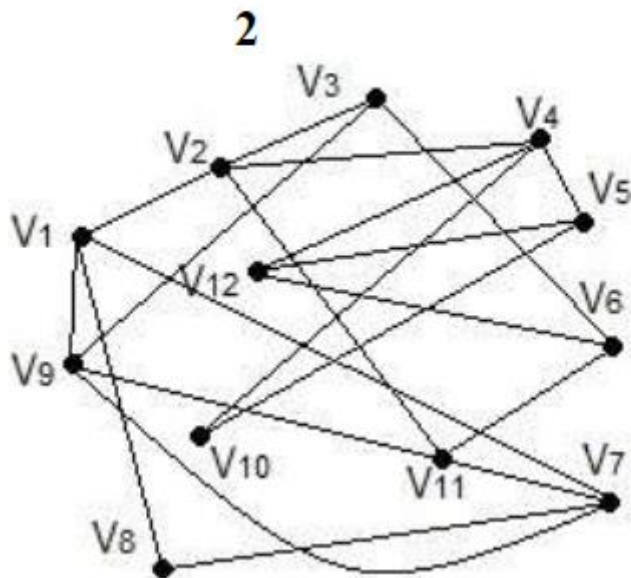
Постановка завдання

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

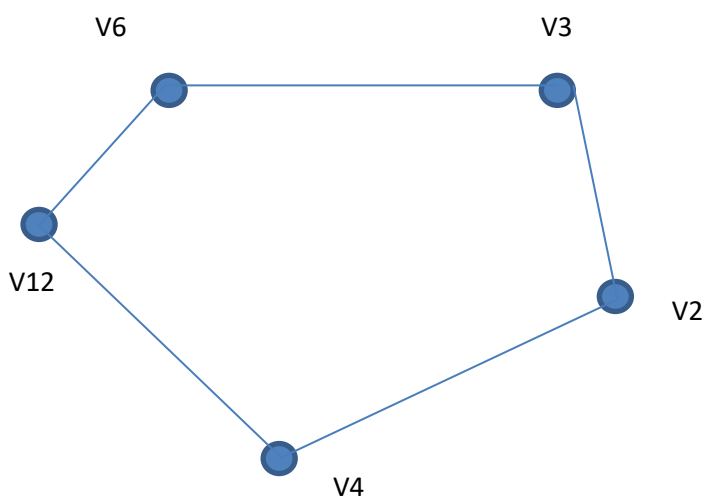


Вага шляху: 21.

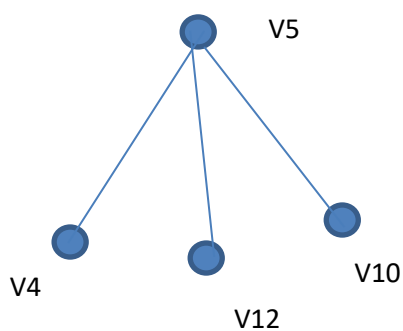
За допомогою γ -алгоритма зробити укладку графа у площині, або довести що вона неможлива.

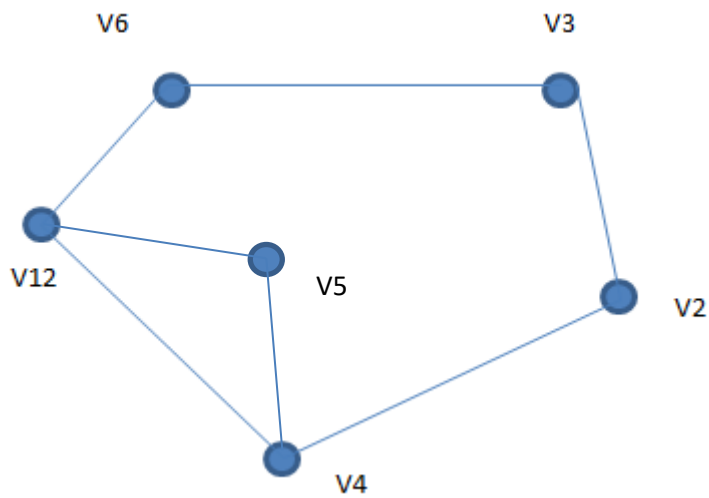


Виділимо цикл з найбільшою кількістю вершин.

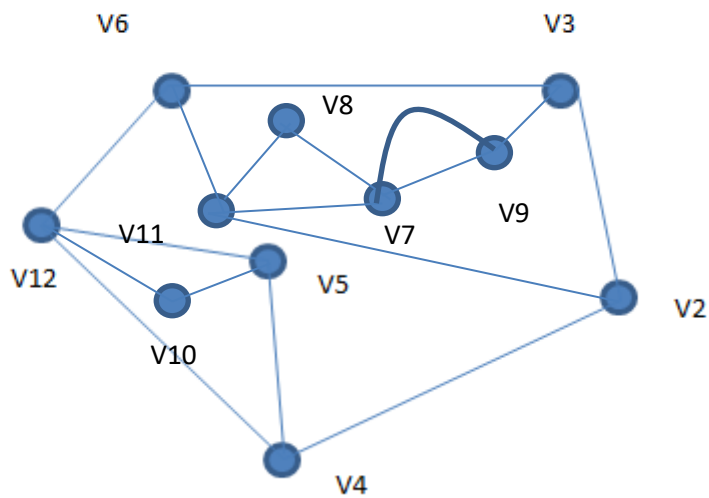


Далі серед вершин, що не належать циклу виділимо ту, що має найбільше спільних ребер з вершинами циклу.
Наприклад V5:

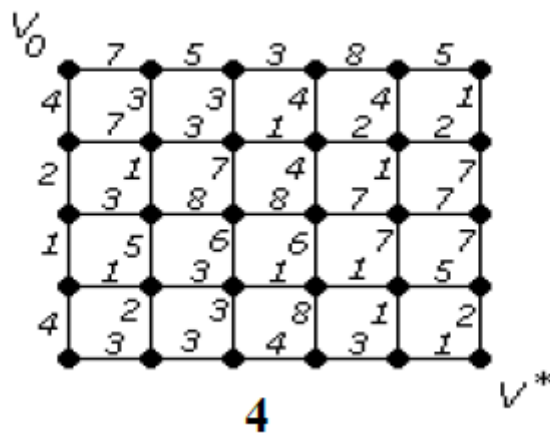




Повторюємо цей алгоритм кожного разу.



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 <bits/stdc++.h>
using namespace std;

int main()
{
    int n,i,j;
    n=30;
    int mygraph[n][n];
    for (i=0;i<n;i++)
    {
        for (j=0;j<n;j++) mygraph[i][j]=0;
    }

    mygraph[0][1]=mygraph[1][0]=7;
    mygraph[0][6]=mygraph[6][0]=4;
    mygraph[1][2]=mygraph[2][1]=5;
    mygraph[1][7]=mygraph[7][1]=3;
    mygraph[2][3]=mygraph[3][2]=3;
    mygraph[2][8]=mygraph[8][2]=3;
    mygraph[3][4]=mygraph[4][3]=8;
    mygraph[3][9]=mygraph[9][3]=4;
    mygraph[4][5]=mygraph[5][4]=5;
    mygraph[4][10]=mygraph[10][4]=4;
    mygraph[5][11]=mygraph[11][5]=1;
    mygraph[6][7]=mygraph[7][6]=7;
    mygraph[6][12]=mygraph[12][6]=2;
    mygraph[7][8]=mygraph[8][7]=3;
    mygraph[7][13]=mygraph[13][7]=1;
    mygraph[8][9]=mygraph[9][8]=1;
    mygraph[8][14]=mygraph[14][8]=7;
    mygraph[9][10]=mygraph[10][9]=2;
```

```

mygraph[12][13]=mygraph[13][12]=3;
mygraph[12][18]=mygraph[18][12]=1;
mygraph[13][14]=mygraph[14][13]=8;
mygraph[13][19]=mygraph[19][13]=5;
mygraph[14][15]=mygraph[15][14]=8;
mygraph[14][20]=mygraph[20][14]=6;
mygraph[15][16]=mygraph[16][15]=7;
mygraph[15][21]=mygraph[21][15]=6;
mygraph[16][17]=mygraph[17][16]=7;
mygraph[16][22]=mygraph[22][16]=7;
mygraph[17][23]=mygraph[23][17]=7;
mygraph[18][19]=mygraph[19][18]=1;
mygraph[18][24]=mygraph[24][18]=4;
mygraph[19][20]=mygraph[20][19]=3;
mygraph[19][25]=mygraph[25][19]=2;
mygraph[20][21]=mygraph[21][20]=1;
mygraph[20][26]=mygraph[26][20]=3;
mygraph[21][22]=mygraph[22][21]=1;
mygraph[21][27]=mygraph[27][21]=8;
mygraph[22][28]=mygraph[28][22]=1;
mygraph[22][23]=mygraph[23][22]=5;
mygraph[23][29]=mygraph[29][23]=2;
mygraph[24][25]=mygraph[25][24]=3;

```

```

mygraph[24][25]=mygraph[25][24]=3;
mygraph[25][26]=mygraph[26][25]=3;
mygraph[26][27]=mygraph[27][26]=4;
mygraph[27][28]=mygraph[28][27]=3;
mygraph[28][29]=mygraph[29][28]=1;

int vertindex[n]{-1};
int numbvectors=0;
for (i=0;i<n;i++)
{
    for (j=0;j<n;j++)
    {
        if (mygraph[i][j]!=0) numbvectors++;
    }
}
int vert[n];
bool vertdot[n];
for (i=0;i<n;i++)
{
    vert[i]=INT_MAX;
    vertdot[i]=0;
}
vert[0]=0;
int siz=1;
int nmin,imin,jmin;
vertdot[0]=1;
while (numbvectors!=0)
{
    nmin=INT_MAX;
    for (i=0;i<n;i++)
    {

```

```

{
    nmin=INT_MAX;
    for (i=0;i<n;i++)
    {
        if (vertdot[i]==1)
        {
            for (j=0;j<n;j++)
            {
                if (vert[i]+mygraph[i][j]<nmin&&mygraph[i][j]!=0)
                {
                    nmin=vert[i]+mygraph[i][j];
                    imin=i;
                    jmin=j;
                }
            }
        }
        if (vert[jmin]>nmin)
        {
            vert[jmin]=nmin;
            vertindex[jmin]=imin;
        }
        vertdot[jmin]=1;
        mygraph[imin][jmin]=mygraph[jmin][imin]=0;
        numbvectors-=2;//signiman pebpa
    }

    int thelast;
    int way[n];
    i=0;
    cout<<"Input finish vertex: ";

```

```

    {
        vert[jmin]=nmin;
        vertindex[jmin]=imin;
    }
    vertdot[jmin]=1;
    mygraph[imin][jmin]=mygraph[jmin][imin]=0;
    numbvectors-=2; // sinximan pebna
}

int thelast;
int way[n];
i=0;
cout<<"Input finish vertex: ";
cin>>thelast;
cout<<"\nWeight = "<<vert[thelast]<<endl;
cout<<"The way from start to finish:"<<endl;
while (thelast!=0)
{
    way[i]=thelast;
    thelast=vertindex[thelast];
    i++;
}
way[i]=0;

for (i;i>=0;i--)
{
    cout<<way[i];
    if (i!=0) cout<<"->";
}
}

```